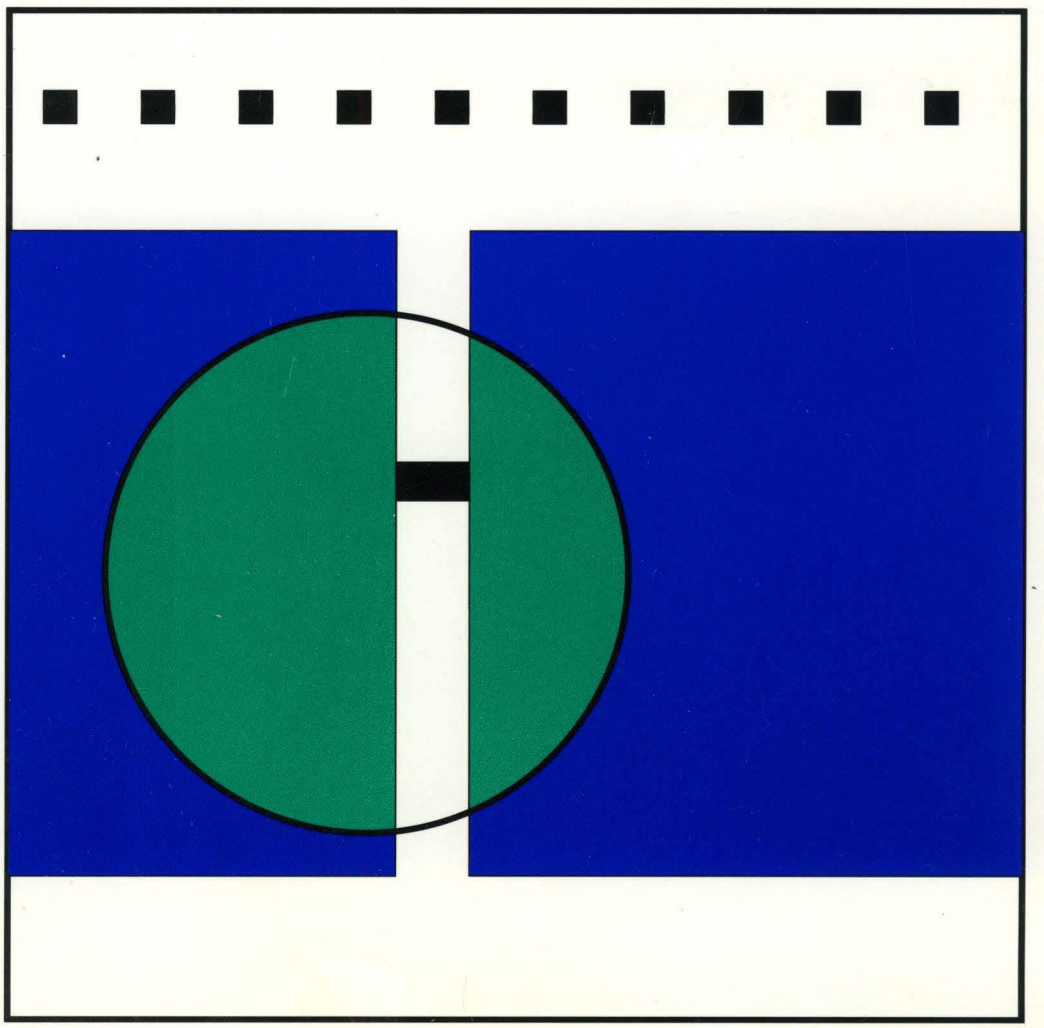
VTAM™

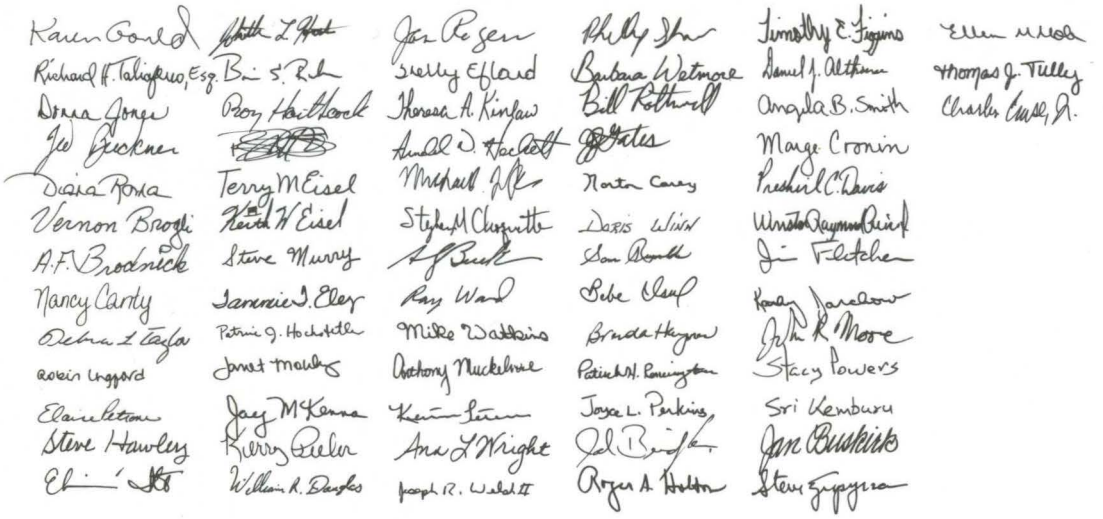
LY43-0065-01

Diagnosis

Version 4 Release 2 for MVS/ESA, VM/ESA, and VSE/ESA



The VTAM V4R2 Team



VTAM™

LY43-0065-01

Diagnosis

Version 4 Release 2 for MVS/ESA, VM/ESA, and VSE/ESA

Note!

Before using this document, read the general information under "Notices" on page xxxvii.

Second Edition (February 1995)

This licensed document applies to the Advanced Communications Function for Virtual Telecommunications Access Method (ACF/VTAM), an IBM licensed program:

VTAM Version 4 Release 2 for MVS/ESA (program number 5695-117)
VTAM Version 4 Release 2 for VM/ESA (program number 5654-010)
VTAM Version 4 Release 2 for VSE/ESA (program number 5686-065).

Publications are not stocked at the address given below. If you want more IBM publications, ask your IBM representative or write to the IBM branch office serving your locality.

A form for your comments is provided at the back of this document. If the form has been removed, you may address comments to:

IBM Corporation
Department E15
P.O. Box 12195
Research Triangle Park, North Carolina 27709
U.S.A.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Contents

Figures

Tables

Notices

Any reference to an IBM licensed program in this licensed document does not imply that IBM intends to make them available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of the intellectual property rights of IBM may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, NY 10594
USA

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement.

This document is not intended for production use and is furnished as is without any warranty of any kind, and all warranties are hereby disclaimed including the warranties of merchantability and fitness for a particular purpose.

Programming Interface Information

This book is intended to help the customer to diagnose *VTAM* and primarily documents Diagnosis, Modification, and Tuning Information.

Warning: Do not use this Diagnosis, Modification, and Tuning Information as a programming interface.

Trademarks

The following terms, denoted by an asterisk (*) at their first occurrence in this publication, are trademarks of the IBM corporation in this country or other countries or both:

Advanced Peer-to-Peer Networking	NetView
AnyNet	Personal System/2
APPN	RETAIN
AS/400	RISC System/6000
CICS	System/360
ES/9370	System/370
FFST	System/390
IBM	VM/ESA
MVS/ESA	VSE/ESA

About This Book

This publication is intended to help system programmers in a VTAM environment to diagnose problems with the VTAM program.

Who Should Use This Book

System programmers should use this book to analyze a VTAM problem, classify the problem as a specific type, and provide information about the problem to an IBM Support Center representative.

You should be familiar with VTAM's service aids and the procedures for reporting problems to an IBM Support Center representative. You should also be familiar with the new functions in VTAM V4R2 that are described in the *VTAM Release Guide for MVS/ESA*, the *VTAM Release Guide for VM/ESA*, and the *VTAM Release Guide for VSE/ESA*.

How to Use This Book

Use this book to isolate and identify problems with your VTAM network and to collect appropriate documentation to resolve network problems.

VM,VSE Use this book in conjunction with the *VTAM Overview for VM/ESA and VSE/ESA* flyer included with this book.

How This Book Is Organized

This book is organized into the following parts:

- "Part 1. Diagnostic Techniques" on page 1 describes how to identify a problem. This part contains the following information:
  - Chapter 1, "Diagnosing VTAM Problems: Where to Begin" on page 3 describes the basic problem identification process.
  - Chapter 2, "Collecting Documentation for Specific Types of Problems" on page 57 describes general procedures to collect documentation.
  - Chapter 3, "Collecting Documentation for TSO/VTAM Problems (MVS)" on page 113 describes procedures to collect documentation for TSO/VTAM problems.
  - Chapter 4, "Collecting Documentation for VSCS Problems (VM)" on page 131 describes procedures to collect documentation for VSCS problems.
- "Part 2. Diagnostic Procedures" on page 189 describes how to use diagnostic procedures. This part contains the following information:
  - Chapter 5, "Using DISPLAY and MODIFY Operator Commands" on page 193 describes how to use VTAM commands to collect dump and trace information.
  - Chapter 6, "Using Dumps" on page 213 describes dump procedures.

  - Chapter 7, "Using VTAM Dump Analysis Tools (MVS)" on page 237 describes how to analyze dump information.
  - Chapter 8, "Using Traces" on page 347 describes how to use traces to collect information.
  - Chapter 9, "Using the VTAM Internal Trace (VIT)" on page 397 describes how to use the VTAM Internal Trace (VIT).
  - Chapter 10, "Using the VIT Analysis Tool (MVS, VM)" on page 411 describes how to use the VIT analysis tool.
  - Chapter 11, "Using VSCS Service Aids (VM)" on page 449 describes messages, commands, dumps, traces, LBCSTATE values, and WEB function codes for VSCS.
  - Chapter 12, "Using Other Problem Solving Tools" on page 539 describes alerts, 3745 bus switching messages, the IBMTEST command, error recording, patch areas, and save area module linkage.
- "Part 3. Diagnostic References" on page 559 contains diagnostic reference material. This part contains the following information:
  - Appendix A, "VTAM Internal Trace (VIT) Record Descriptions" on page 567 describes the VIT records.
  - Appendix B, "Finding VTAM Buffer Pools (VM, VSE)" on page 837 provides step-by-step instructions for finding buffer pools.
  - Appendix C, "First Failure Support Technology (FFST) Probes (MVS)(VM)" on page 845 describes the FFST* probes that trigger dumps when an unusual condition occurs in VTAM.
  - Appendix D, "Channel Programs" on page 873 illustrates channel programs to help resolve network I/O-related problems.
  - Appendix E, "Network Flows" on page 925 illustrates flows of VTAM request or response units (RUs) in order to help resolve VTAM problems.
  - Appendix F, "APPN Flows" on page 1013 describes flows between APPN end nodes, network nodes, interchange nodes, and the subarea network.
  - Appendix G, "Control Point/Control Block (CPCB) Operation Codes" on page 1127 describes control point/control block (CPCB) operation codes that identify VTAM processes.
  - Appendix H, "Storage and Control Block ID Codes" on page 1137 summarizes storage and control block ID codes that identify VTAM control blocks.
  - Appendix I, "Path Information Unit (PIU) Discard Reason Codes" on page 1141 describes PIU discard reason codes in order to help isolate VTAM session and network problems.

Terms Used in the VTAM Library

Clarification of some terms used in the VTAM library follows:

- The term *Advanced Peer-to-Peer Networking* (APPN*)* represents VTAM's implementation of APPN.
- The terms *CPCP* and *CP-CP* are similar but have different meanings. *CPCP* is the name of a start option or operand. *CP-CP* refers to CP-CP sessions between control points.

- The term *end node* represents an APPN end node.
- The term *integrated communication adapter (ICA)* is used to represent all of the following, except where it is necessary to use the specific term:
  - Communication adapter
  - Telecommunications Subsystem Controller
  - IBM Token-Ring Subsystem Controller
  - Workstation Subsystem Controller
  - Multi-Protocol Communication Subsystem Controller.
- The term *network node* represents an APPN network node.
- The term *VTAM* refers to the VTAM program for V4R2 unless otherwise stated.

Symbols Used in the VTAM Library

The following symbols are used in the VTAM library to indicate information that applies only to a particular operating system or processor:

MVS Indicates information that applies only to MVS/ESA*.

VSE Indicates information that applies only to VSE/ESA*.

VM Indicates information that applies only to VM/ESA*.

These symbols either precede or follow unique information in the VTAM library. Following is an example of how these symbols are used:

VM VSCS is a VTAM application program and must be defined to VTAM.

What Is New in This Book

Information has been added to this book to reflect the new functions in VTAM V4R2 for VM/ESA and VSE/ESA. Changes have also been made as a result of user comments.

MVS This book was last published in May 1994 with order number LY43-0065-00.

VM VSE This book was last published in March 1992 with order number LY43-0059-1.

The glossary has been deleted from this publication. For definitions of the terms and abbreviations used in this book, refer to the *VTAM Glossary*.

The following sections contain summaries of the changes that have been made for each new function. Further information about the functions can be found on the pages referred to in these summaries. "Usability Enhancements to This Book" on page xliii describes usability changes that have been made to this book.

Network Dynamics

Advanced Peer-to-Peer Networking (APPN) Support

"Common Problems in APPN Networks" on page 27 has been added to help you solve some common APPN problems before calling the IBM Support Center.

Appendix F, "APPN Flows" on page 1013 has been added.

Operator Interface Enhancements

Display of Active Traces

A new command, DISPLAY TRACES, has been added. See "Display Traces" on page 208.

Performance Enhancements

Multipath Channels between Host Processors

Information on this new connectvity has been added to Appendix D, "Channel Programs" on page 873.

Non-Disruptive Deactivation of Cross-Domain Resource Managers

For flow diagrams illustrating this function, see Figure 168 on page 994, Figure 169 on page 995, and Figure 170 on page 996.

Problem Diagnosis

"Common Problems in Subarea Networks" on page 6 has been added to help you solve some common subarea problems before calling the IBM Support Center.

Buffer Contents Trace Enhancements

The full buffer contents trace can now record all data contained in each buffer, making it easier to diagnose problems in the network. See "Buffer Contents Trace" on page 369.

Capturing of Diagnosis Information with First Failure Support Technology (FFST) (VM)

First Failure Support Technology (FFST) helps you resolve VTAM software problems by capturing information about a potential problem when it occurs. See "First Failure Support Technology (FFST) for VTAM (MVS) (VM)" on page 217 and Appendix C, "First Failure Support Technology (FFST) Probes (MVS)(VM)" on page 845.

VTAM Internal Trace (VIT) Enhancements

Appendix A, "VTAM Internal Trace (VIT) Record Descriptions" on page 567 includes new and updated VIT entries.

The following default VIT options are automatically active and cannot be turned off:

- API
- MSG
- NRM

- PIU
- SSCP.

For more information see "Activating the Internal Trace" on page 398.

VM You can simultaneously record VIT data internally and externally. See "Internal and External Trace Recording" on page 401.

VSE See "Recording Traces in External File (MODE=EXT)" on page 403 for information on determining the size of the trace table.

VTAM Internal Trace (VIT) Analysis Tool (VM)

You can now process externally recorded VIT records with the VIT analysis tool, which enables you to:

- Analyze storage
- Count RUs
- Extract information from VIT records.

See Chapter 10, "Using the VIT Analysis Tool (MVS, VM)" on page 411.

System and Configuration Management

Dependent LU Server

"Dependent LU Server Flows" on page 1098 has been added.

Virtual-Route-Based Transmission Groups

A flow diagram has been added, illustrating activation of a CDRM with a virtual-route-based transmission group. See Figure 117 on page 945.

VM/SNA Console Support (VSCS) Enhancements

See Chapter 11, "Using VSCS Service Aids (VM)" on page 449 for enhancements to VSCS, including network qualified name (NQN) support, the operator trace (OPER), and the storage statistics trace (STAT).

Usability Enhancements to This Book

- To make information easier to find, the book has been divided into three parts, and some chapters have been reorganized.
- A sample path problem has been added. See "Example: Solving Path Problems" on page 199 for additional information.
- Indexes have been added for the flow diagrams, and the subarea flows have been reorganized so that flows are easier to find. See Table 64 on page 926 and Table 71 on page 1014.
- Command syntax is depicted in "railroad track" syntax diagrams instead of the brackets-and-braces style used in previous editions of this manual. The railroad track style is being adopted by all IBM products. If you are not familiar with railroad tracks, refer to "How to Read the Syntax Diagrams" on page 238.

  If you wish to comment on the use of railroad tracks or other documentation concerns, a Reader's Comment Form is located at the back of this manual. Your feedback will be appreciated.

Where to Find Information about VTAM

"Bibliography" on page 1159 describes the books in the VTAM V4R2 library, arranged according to related tasks. In addition, Table 80 on page 1165 tells where to go for information on a variety of related topics.

Part 1. Diagnostic Techniques

Chapter 1. Diagnosing VTAM Problems: Where to Begin

This chapter includes the following sections:

- To help you determine the source of your problem, see "Determining Whether the Problem Is VTAM or Non-VTAM" on page 4.
- To compare your problem to a list of common problems that have been identified by the IBM Support Center, see "Common Problems in Subarea Networks" on page 6 and "Common Problems in APPN Networks" on page 27.

  For additional information, see "Common Symptoms and Associated VTAM Problem Types" on page 39.
- If you are having problems with the trace or dump analysis tools, see "VTAM Internal Trace (VIT) Analysis Tool Problems (MVS) (VM)" on page 42 and "VTAM Dump Analysis Tool Problems (MVS)" on page 46.
- To gather additional documentation to help you to solve your problem, see "Recommended Documentation for VTAM Problems" on page 48.
- To prepare your documentation for submission to the IBM Support Center, see "Submitting Problem Documentation" on page 53.

Determining Whether the Problem Is VTAM or Non-VTAM

Problems can be classified into two types:

VTAM problems
These are problems that occur in the VTAM program.

Non-VTAM problems
These are problems that occur because of changes to your network or problems caused by other application programs or software in the network, such as a network control program (NCP) or a local area network (LAN).

If you did any of the following, the problem might be in your network setup, in your configuration, or in another IBM product.

- Did you modify an application program that has run without problems in the past?
- Did you modify a product exit routine that has run without problems in the past?
- Did you change the processing environment? For example, did you introduce a new host processor or communication controller?
- Did you modify the operating system, or did you install a new release of the operating system?
- Did you add a new terminal to your VTAM network that had incorrect features or incorrect Request for Engineering Activity (REA) and Engineering Change (EC) levels?
- Did you attach a link?
- Did you set switches at a terminal?
- Did you initialize link parameters for a programmable controller?
- Did you modify installation-provided VTAM tables? For example, did you modify logmode, class-of-service (COS), or unformatted system services (USS) tables?
- Did you modify VTAM messages?

If you cannot resolve the problem on your own:

- Compare your problem to the examples in "Common Problems in Subarea Networks" on page 6 and "Common Problems in APPN Networks" on page 27.
- Check Table 4 on page 39 for your problem symptom.
- Follow the instructions in "Recommended Documentation for VTAM Problems" on page 48. To gather additional information, refer to Chapter 2, "Collecting Documentation for Specific Types of Problems" on page 57.
- For non-VTAM problems, call your IBM branch office. For suspected VTAM problems, do either of the following steps:
  - If you have access to IBMLink*, search for known problems in this area. If no applicable matches are found, report the problem to IBM by using the electronic technical report (ETR) option on IBMLink.
  - Contact the IBM Support Center at 1-800-237-5511.

If a reported problem is a hardware, network definition, or user definition error, the IBM Support Center representative creates an ASKQ item for VTAM. The ASKQ item includes the solution for the problem and can be found in the problem determination database (PDDB).

Common Problems and Symptoms

This section contains the following information:

- "Common Problems in Subarea Networks"
- "Common Problems in APPN Networks" on page 27
- "Common Symptoms and Associated VTAM Problem Types" on page 39.

Even in cases when a VTAM problem has the same symptoms as a non-VTAM problem, by studying similar examples in this section, you may be better prepared when you contact the IBM Support Center.

If you have access to a software support database, you can search for your problem in that database and apply any recommended correction.

Common Problems in Subarea Networks

Table 1 includes a brief description of several common problems that occur in subarea networks. For additional information, go to the page indicated.

Table 1 (Page 1 of 2). Index of Common Problems in Subarea Networks

Problem	See
Abend 0C4 in ISTTSCPF when move character instruction processed (reason code 4, 10, or 11)	Page 7
Activating an NCP and resources fail	Page 25
APPL-APPL storage expansion failure with messages IST154I, IST561I, IST999E, IST566I, and IST930I	Page 19
BIND failure with message IST663I (sense code 083500xx or 08210000) and USSMSG07	Page 23
CICS* logon problem with message IST663I (sense code 08210000)	Page 17
CINIT failure with message IST663I (sense code 08010000)	Page 20
DSRLST pending condition and CD DSEARCH PENDING in message IST530I or message IST1278I	Page 17
LU hung in PNFYx state	Page 8
Message IKT029I with return code 061001 (TSO logon failure from a session manager application program) MVS	Page 9
Message IST259I and sessions end unexpectedly for a terminal, PU, line, or NCP	Page 10
Message IST259I indicating that an INOP RU was received for a link problem	Page 9
Message IST264I (required COS entry undefined) and message IST663I with sense code 08610000	Page 17
Message IST467I with error type 05, 07, 08, or 0B during activation of a PU	Page 11
Message IST489I when activating a major node	Page 14
Message IST530I (DSRLST pending condition)	Page 17
Message IST530I (GUNBIND pending)	Page 15
Message IST530I (NMVT PENDING)	Page 16
Message IST663I with sense code 08010000 (CINIT failure)	Page 20
Message IST663I with sense code 08210000 (CICS logon problem)	Page 17
Message IST663I (sense code 083500xx or 08210000) and USSMSG07 (BIND failure)	Page 23
Message IST663I (sense code 08610000) and message IST264I (required COS entry undefined)	Page 17
Message IST1272I when activating a major node	Page 14

Table 1 (Page 2 of 2). Index of Common Problems in Subarea Networks

Problem	See
Message IST1278I (DSRLST pending condition)	Page 17
Message IST1278I (GUNBIND pending)	Page 15
Message IST1278I (NMVT PENDING)	Page 16
Messages IST154I, IST561I, IST999E, IST566I, and IST930I (APPL-APPL storage expansion failure)	Page 19
Messages IST718I and IST719I when activating a CDRM	Page 14
PNFYx resource state and LU hung	Page 8
PSESEND session termination state and hung session	Page 15
Resources fail when activating an NCP (sense codes 10030000 and 08090000)	Page 25
Sense code 08010000 with message IST663I (CINIT failure)	Page 20
Sense code 081C0008 Received on ACTLINK VM	Page 21
Sense code 081C000C Received on ACTLINK VM	Page 21
Sense code 08210000 with message IST663I (CICS logon problem)	Page 17
Sense code 083500xx or 08210000 (message IST663I) and USSMSG07 (BIND failure)	Page 23
Sense code 08610000 (message IST663I) and message IST264I (required COS entry undefined)	Page 17
Sense code 0888000x with session failure	Page 25
Sense code 800A0000 or no message with session failure	Page 24
Sense code 80130104 and path problems	Page 199
Sense codes 10030000 and 08090000 (resources fail when activating an NCP)	Page 25
Session fails with sense code 0888000x	Page 25
Session hung with PSESEND session termination state	Page 15
Sessions end unexpectedly for a terminal, PU, line, or NCP with message IST259I	Page 10
Sessions end with no message or sense code 800A0000	Page 24
Storage problem	Page 105
TSO application program receives partial input for the TGET macroinstruction MVS	Page 20
TSO logon failure from a session manager application program with message IKT029I and return code 061001 MVS	Page 9
USS message USSMSG07 and message IST663I with sense code 083500xx or 08210000 (BIND failure)	Page 23
VTAM trace records were expected but are not in the GTF trace data set	Page 19

Descriptions of Common Problems in Subarea Networks

This section includes examples of common problems in subarea networks. See Table 1 on page 6 for an index of these problems.

Abend 0C4 Occurs in ISTTSCPF

Problem Statement

An abend 0C4 with a reason code of 4, 10, or 11 occurs in all levels of the module ISTTSCPF when a move character instruction is processed.

Common Symptoms

The USS message contains unexpected or extraneous characters, but not all USS messages are affected. Devices might not activate correctly or might fail to logon correctly. An abend 0Cx might occur in module ISTTSCPF.

Probable Cause

The USSTAB is incorrectly defined. In this module, register 4 points to storage that may not be paged-in (reason code 10 or 11) or to storage that should not be accessed (reason code 4).

The TSCB contains a data length field (TSCDATLN) equal to the length of the USSMSG text plus the length field itself. Subtracting the value in register 4 from the starting address of the USSMSG table shows that only the value of TSCDATLN (minus 2 bytes) was moved because the full amount of storage that was referenced was not paged-in.

User Response

Code only the length of the USSMSG entry in the USSMSG table. Do not include the size of the length field.

Line hangs in PALNK status

Problem Statement

A line is hung in PALNK status when ISTATUS=INACTIVE is coded on a line definition command.

Common Symptoms

Messages IST530I (or IST1278I, if MSGLVL=V4R1) ACTLINK PENDING and IST531I are received from a VTAM to ICA LINK request.

Probable Cause

On the LINE definition statement, the continuation character was put in column 71 instead of column 72.

User Response

At assembly time, determine how comments are to be identified for each type of definition statement.

LU Hung in PNFYx State

Problem Statement

An LU can hang in a PNFYx state if the application program does not issue the CLSDST macroinstruction when a LOSTERM user exit routine is scheduled.

Common Symptoms

An LU is hung in a PNFYx state. The LU is unable to log on to an application.

Probable Cause

An application program failed to issue the CLSDST macroinstruction when the application's LOSTERM user exit routine was scheduled with a reason code indicating that the CLSDST macroinstruction should be issued.

User Response

Check with the owner of the application program for known problems. A VTAM internal trace with the application interface (API) option active indicates whether the LOSTERM user exit was scheduled and the reason code that was passed. The API trace option can also be used to determine if the CLSDST was issued by the application in response to a LOSTERM user exit.

Notes:

1. See Page 85 for information on the VARY INACT,FORCE command and PNFYx status.
2. The information in this problem description is from information APAR II00757. See that APAR for additional information.

Message IKT029I (Return Code 061001) Received with TSO Logon Failure (MVS)

Problem Statement

Cannot log on to TSO from a session manager application.

Common Symptoms

The following message is displayed:

```
IKT029I  RC= 061001 SENSE= code TERMINAL termid ABOUT TO BE RELEASED BY VTAM
```

Probable Cause

Either the application or D/T8100 expects the first BIND from TSO to be from terminal control address space (TCAS). TCAS will only send the BIND for the TSO sub-application program (TSOxxxx).

User Response

If the secondary logical unit (SLU) does not support this type of session initiation, specify FASTPASS=NO on the SLU definition statement to force TCAS to send a BIND to the SLU before the TSO sub-application program sends its own BIND.

Message IST259I Received with INOP RU

Problem Statement

Message IST259I is generated by the inoperative (INOP) RU processor. The INOP RU is generated by the data link control (DLC) component for the subarea controlling the link, either intermediate network node (INN) or route extension (REX).

Common Symptoms

The following message is displayed:

```
IST259I  INOP RECEIVED FOR nodename CODE = code
```

If the link is INN, an ER.INOP will also flow, producing a series of explicit route (ER) or virtual route (VR) failure messages. This leads to an incorrect diagnosis when you do not associate the ER.INOP with the link or link station INOP.

If message IST259I contains the name of a channel-attached NCP or a local device, MVS message IOS000I or VM VSE IST446I might accompany the failure.

Some local SNA controllers require I/O buffer size to be an even number. For example, if an odd number is coded for a 3174, message IST259I with CODE=01 will be displayed at activation.

VM VSE Message IST259I with CODE=01 might be returned for a 9370 using a 6120 or 6120 control unit emulator if the I/O buffer size is less than 256.

Probable Cause

NCP link

This is a communication facility problem. Either the retry limit is exhausted, a negative acknowledgement is received for an SDLC transmission, a modem error occurred, or a link failure occurred.

Channel link
: Either a data transfer count mismatch occurred, an NCP abend has occurred, or the NCP was reloaded by another host.

ICA link
: This is the same as the NCP link problem, but no RECMS was received.

User Response

Trace the link.

List the system MVS VM LOGREC or VSE SYSREC to obtain the data from the record management statistics (RECMS) that accompany an INOP originating in an NCP node. The RECMS identifies the error that produced the INOP. Use environmental record editing and printing (EREP) to print the LOGREC records. Use the network problem determination application (NPDA) to interpret the RECMS record.

VTAM does not generate the RECMS for channel link and link station failures. The MVS LOGREC or VSE SYSREC entry for a local device will contain only statistical data.

Correct the error condition.

Note: The IBM Support Center representative can only suggest that you list MVS LOGREC or VSE SYSREC and assist you with interpreting the record.

If a channel-attached SNA device (NCP or cluster controller) is experiencing the INOPs at a regular or predictable interval (for example, every hour), the problem could be that VTAM'S ERP routine has been deleted. MVS Verify that CSECT ISTZBM0K in loadmodule IGE0004 (LPALIB) has not been deleted.

Message IST259I Received and Sessions End Unexpectedly

Problem Statement

One or more sessions have ended unexpectedly, and a terminal, PU, line, or NCP is in a wait state.

Common Symptoms

The following message is displayed:

```
IST259I  INOP RECEIVED FOR nodename CODE = code [text]
```

Probable Cause

- If the node is an NCP, the NCP detected an error and generated the INOP RU message.
- If the node is a channel-to-channel (CTC) link or a CTC link station, VTAM detected an error from an IO operation and generated the INOP RU message.
- If the node is a local attachment device, VTAM detected an IO error and generated the INOP RU message.

User Response

- For an explanation of the code in IST259I, refer to the description of the message in *VTAM Messages and Codes*.
- Check the system log for system (IOS) error messages that contain status information.
- If the NetView* program is installed, check NPDA for logged errors.
- Run EREP against LOGREC, and check for errors related to the device.

  Note: This information should identify the component causing the error. Contact the appropriate service organization for help with a specific component problem.

Message IST467I Received with CONTACTED ERROR TYPE 05, 07, 08, or 0B

Problem Statement

Message IST467I is received with contacted error type 05, 07, 08, or 0B during activation of a resource. The message indicates that the XID was rejected by the PU.

Common Symptoms

IST467I is the first in a group of messages. The exchange IDs (XIDs) are included in IST468I, IST469I, IST470I, and IST471I in the message group. Compare the XIDs to determine why the PU rejected the XID.

Sample XIDs from an IST467I message group:

```
*********************************************
*   The following is for XID format 2.  All   *
*   references to bytes and bits are in hex.  *
*********************************************
XID1 (Received from the NCP) =
      242AFFF0 00000000 00080000 00010000
      00035007 D5C3D7D3 D6C1C440 80000203
      002A05F3 00800000 0000
XID2 (Sent to the NCP by VTAM) =
      242AFFF0 00000000 200800F9 DE010000
      00010000 40404040 40404040 81000200
      002A05F3 00000000 0000
```

Note: If *type* is **0B** in message IST467I, additional error information may be contained in a CV X'22' appended to the end of the XID. See the **User Response** for examples.

Probable Cause

Note: The explanations that follow cover more than the single error that the sample XID1 and XID2 represents.

The first digit of the XID is the format. In the sample XIDs above, the format is format 2.

Byte X'12' of the XID1 received from the NCP is the error byte.

Bit 0	Reserved (unused)
Bit 1	Received XID is unacceptable.
Bit 2	Incompatible
Bit 3	Transmission group (TG) undefined

Bit 1 of byte X'12' is set for the following reasons:

1. The XID2 at displacement X'00' was not equal to 24 or 25.
2. The XID2 at displacement X'08' was not equal to 20.
3. The XID2 at displacement X'13' was not equal to 00.
4. The XID2 at displacement X'1E' was not equal to 02.
5. Depending on the release of NCP you have:
   a. *For NCP V4R3, V5R2, and later:* The XID2 from VTAM at displacement X'0B'—X'0C' is less than 1296 decimal (X'0510'). The value in this field is the result of MAXBFRU times UNITSZ specified on the HOST definition statement in the NCP. The value of UNITSZ should be the same as the value of the IOBUF buffer size in the VTAM start list. This error is set only when the NCP definitions have specified the HOST connection using GROUP LNCTL=CA.
   b. *For NCP prior to V4R3 and V5R2:* The XID2 from VTAM at displacement X'0B'—X'0C' is less than (XID plus X'20'—X'21') times (XID plus X'22'—X'23') minus (XID plus X'24') in the XID sent by NCP. This result corresponds to the value

specified on the MAXBFRU operand times the value specified on the UNITSZ operand minus the value specified on the BFRPAD operand.

6. There is no path to the subarea number defined at X'11' in the XID2 that uses this connection.
7. Bit 2 of byte X'12' is set because the received XID1 at X'25' is not equal to X'20', and an existing connection exists with the origin subarea.
8. Bit 3 of byte X'12' is set because either the TG number in the XID2 at displacement X'0D' or the subarea from the XID2 at X'11' is unknown to the NCP.

User Response

Reasons 1, 2, 3, and 4, should not occur, but should help to verify XID offsets.

For reason 5, refer to the *VTAM Resource Definition Reference* and the *NCP, SSP, and EP Resource Definition Reference* regarding specification of buffer sizes.

For reason 6, refer to the *VTAM Resource Definition Reference* and the *NCP, SSP, and EP Resource Definition Reference* regarding the definition of PATH statements. Also, transmission group (TG) mismatch could cause the problem. A TG mismatch could occur, for example, if an NCP is attached as a CA major node, and TG=ANY is coded in the CA major node in VTAM, and TG=ANY is coded on the NCP line definition for this attachment.

For reason 7, refer to the *VTAM Resource Definition Reference* regarding the use of the CHANCON parameter of the PCCU macro.

For reason 8, refer to the *NCP, SSP, and EP Resource Definition Reference* regarding the use of the CANETID parameter on the BUILD macro. Verify that it is coded correctly for each network in which it is assigned.

Notes:

1. In the sample XID1 given above, byte X'12' contains the value X'50'. Bit 2 and bit 3 indicate that the XID2 was unacceptable and that the transmission group was not defined. The problem in this case was that there were no PATH definition statements defined in the NCP for the host subarea.
2. The contacted error type 05 can also be posted if the NETIDs in the 2 XIDs do not match.

   The NETID will appear in a CV X'12' at the end of the XIDs. If they do not match, correct the NETID operand on the PU definition statement in one or both PU definitions.
3. A contacted error type 05 may occur if a channel-to-channel connection is defined between two VTAMs that have the same subarea. Subarea numbers must be unique.

   In this case, the 4 bytes starting at offset X'E' in XID1 and XID2 will be the same. XID offset X'E' contains the subarea numbers. XID1 contains the subarea number of the receiver, and XID2 contains the subarea number of the sender.

The following examples are for type 0B when a CV X'22' is appended to the end of the XID.

```
*****************
*   Example 1:  *
*****************

IST467I  CONTACTED ERROR TYPE  0B FOR ID = NCPPUX
IST468I  XID1=348AFFFFFFFD000010F7410000000080150213500032 7E03000000F0
IST469I  000107080000000000460909801500000000004F0E0AF4C8D6C4C34BC3F0F1
IST469I  F10E08F1D5F0F7F9F0F0F80E09F7D7E4D3F2F2F4F0F1103700161101130011
IST469I  F3F7F4F5F3F1C1F5F7F0F0F1F0F8F6F42011040E02F5F6F8F8F2F3F1F0F0F2
IST469I  F0F20804F0F6F0F2F0F007099302880815

IST470I  XID2=3596FFF00000000010B7410000000080000213000030000300200180
IST471I  000107080000000000E0AF1C8D6C4C34BC3F0F1F00E0AF4C8D6C4C34BC3F0
IST471I  F1F00E07F7D5C3D7D7E4E7460909800000000000000000103A002311040E02F5
IST471I  F6F9F5F1F1F7F0F1F1F0F10804F0F4F0F1F0F00A06C1C3C661E5E3C1D41611
IST471I  01130011F9F0F2F10000000000F0F1F2F0F6F1F2220700180000000000
                                            |
                                         CV X'22' showing error
                                         in byte X'18'

*****************
*   Example 2:  *
*****************

IST467I CONTACTED ERROR TYPE 0B

XID1=32 4A 05 6A 40 32 00 00 80 06 C0 00 00 00 00 00 01 01 0B 70 00 0F
     DC 00 00 00 00 07 00 0E 0D F4 E2 C5 E2 D7 C1 C4 F0 F0 4B E4 E3 E5
     10 17 F1 16 11 01 13 00 11 F9 F4 F0 F6 C6 F4 F5 F4 F4 F0 F0 C1 F4
     F0 F3 F2 22 03 00 0F 00
              |
     CV X'22' showing error
     in byte X'0F'

XID2=34 85 FF FF FF FC 00 00 10 C7 41 00 00 00 00 80 00 01 0B 70 0010
     BA 00 00 00 00 07 00 46 09 09 80 00 00 00 00 00 00 1F 0E 12 F4 E2
     C5 E2 D7 C1 C4 F0 F0 4B E2 C5 E2 D7 C1 C4 F0 F4 0E 05 F1 D5 F3 F1
     D7 0E 07 F7 C3 F3 F1 E4 E3 E5 10 37 00 16 11 01 13 00 11 F3 F7 F4
     F5 F4 F1 F0 F5 F7 F0 F0 F1 F0 F0 F9 F1 20 11 04 0E 02 F5 F6 F8 F8
     F2 F3 F1 F0 F0 F2 F0 F2 08 04 F0 F6 F0 F2 F0 F0 07 09 93 02 67 12
     59
```

```
*****************
*   Example 3:  *
*****************

IST590I CONNECTIN ESTABLISHED FOR PU IBT10P1 ON LINE J000A027
IST467I CONTACTED ERROR TYPE 0B FOR ID = IBM000P1
IST468I XID1=326C05D90001000008074C100000000800010B707890001000004
IST469I 000E0FF4D5C5E3F2E2C9E34BE6C1D5C7C3D70E09F7D3C9D5D2F0F0F0F11028
IST469I 001111040E02F5F6F2F1F2F5F4F0F0F2F0F0161103130011F5F1F7F0F0F0F0
IST469I F0F0F0F0F0F0F0F0F0F0220700090108090054
                          |
              CV X'22' with error byte X'09',
              bit 1 offset, and sense 08090054 included.
              The new length and sense occurs only in
              APPN nodes (VTAM Version 4 Release 1 and
              NCP Version 7 Release 1 or higher).

IST470I XID2=3471FFFFFFFC000010864000000000000000010B7000105A0000000007
IST471I 000E0FF4D5C5E3F2E2C9E34BC9C2E3C7E6F30E08F1C9C2E3F1F0C7E6103700
IST471I 161101130011F3F7F4F5F6F1F0F0F0F0F0F0F0F0F7F8F92011040E02F5F6F8F8
IST471I F2F3F1F0F0F1F0F20804F0F7F0F1F0F007099303191115
```

See the description of message IST467I in *VTAM Messages and Codes* for additional information.

Message IST489I or IST1272I Received When Activating a Major Node

Problem Statement

Message IST489I or message IST1272I is received when VTAM is activating a major node.

Common Symptoms

Message IST489I or message IST1272I is received.

Probable Cause

These messages indicate a duplicate name, parameter, or network address.

- The name might be the name of the node or subnode.
- The parameter might be the IDBLK and the IDNUM values specified for another subnode.
- The CPNAME specified for a subnode might be duplicated in the network.

User Response

The duplicate name can be discovered by entering the VTAM DISPLAY ID parameter for the *name* specified in message IST489I or message IST1272I.

If the node receiving message IST489I or message IST1272I contains IDBLK and IDNUM or CPNAME, check other definition statements that also contain IDBLK, IDNUM, or CPNAME for a duplicate definition name.

A duplicate network address may be more difficult to find because most network addresses are not explicitly defined. However, because the VTAM internal trace is run with OPT=(NRM,MSG), you can find the duplicate using the following process:

1. When the trace is run and message IST489I or message IST1272I has been issued, use the console DUMP command to dump MVS VTAM's region and CSA or VM VTAM's virtual machine and CSA or VSE VTAM's partition and SVA .
2. Locate the MSG entry for message IST489I or message IST1272I in the trace.

3. Before the message entry, locate the SRTA entry, which should have a non-zero return code, usually 04. The SRTE pointer field of this SRTA entry will point either directly or indirectly to the duplicate name.

   The SRT entry address field contains a storage address that can be located in the VTAM dump. This address points to the node name containing the duplicate definition name, or the SRT entry points to the node containing the duplicate name or address. This pointer is at offset X'10' into the SRT entry. For more information see "SRTx Entry for SRTADD, SRTCHG, SRTDEL, SRTFIND Macros" on page 794.

Message IST530I or IST1278I Received with GUNBIND PENDING or Session Hangs in PSESEND State

Problem Statement

A GUNBIND PENDING message is received at logoff time in a cross-domain environment (if IOPD is specified or defaulted in the VTAM start options), or the session hangs in PSESEND session termination state.

Common Symptoms

- Message IST530I or IST1278I:

  ```
  GUNBIND PENDING FROM applname TO LU
  ```

- Message IST530I or IST1278I

  ```
  GUNBIND PENDING FROM VTAM TO LU
  ```

- The session displays PSESEND as the session termination state.

Probable Cause

- The application did not issue a CLSDST macroinstruction.
- The device did not respond to the UNBIND request or returned a response that was incorrect or not valid.
- A virtual route between the primary logical unit (PLU) subarea and the secondary logical unit (SLU) subarea is held or blocked.
- The network ID defined in the NCP does not match the network ID coded in the VTAM start options.

User Response

- Enter (on the terminal owning the host) and note the status:

  ```
  D NET,ID=devicename,E
  ```

- If the session termination state is PSESEND, enter:

  ```
  D NET,SESSIONS,SID=sid (where the SID is that of the PSESEND
                          session in the IST635I message group)
  ```

  This display will show which session partner is withholding the session end signal to complete the session termination.

  - If a SESSEND is needed from the PLU, VTAM is waiting for a CLSDST macroinstruction to be issued.
  - If a SESSEND is needed from the SLU, there is usually a problem in a network element such as the host VTAM, NCP, or SLU.

- If the device is remote and hangs in PSESEND session termination state at logoff, start the following trace (in the device-owning host), and trace a logon and logoff:

```
F NET,TRACE,TYPE=BUF,ID=devicename
```

The X'15' vector in the SESSST and SESSEND RUs (if they are present) should be checked to see if the network ID matches the network ID coded in the VTAM start options.

- Enter (on the application-owning host) and note the status:

  ```
  D NET,ID=applname,E
  ```

  This indicates whether other sessions are affected. If the application name has many sessions, this display output can be very large.

  ```
  D NET,ID=devicename,E
  ```

  This indicates whether the session status matches the session status in the device host.

  ```
  D NET,ROUTE,DESTSUB=device_subarea_number,TEST=YES
  ```

  This indicates that a virtual route is held or blocked.

  ```
  D NET,TERM,SID=sid,TYPE=FORCE
  ```

  This may help expedite session termination.

- If it is suspected that no CLSDST macroinstruction is being issued, a buffer trace of the application and a VTAM internal trace with MODE=EXT,OPT=API specified may be needed to verify:
  - That the application was notified of session termination
  - Which exit was scheduled
  - What action/commands were issued (if any) by the application.

Message IST530I or IST1278I Received with NMVT PENDING

Problem Statement

Message IST530I or IST1278I is issued for a PU even though the NetView program (if installed) or the RISC System/6000* network management program for RISC System/6000 devices receives session awareness (SAW) data for a SNA device.

Common Symptoms

Message IST530I or IST1278I is issued each time the IOPD timer expires. For additional information, see the message description in *VTAM Messages and Codes.*

Probable Cause

The device is not real-time-monitor-capable. This means that the device did not process the response and return the requested information properly to the NetView program for most devices, or to Network Management/6000 for RISC System/6000 devices. A microcode change is needed to permanently resolve this problem.

User Response

You can prevent this problem by pointing the device to a KCLASS and using a SAW data filter to stop VTAM from attempting to collect the data.

Note: See "Reducing Session Awareness (SAW) Data" in the *VTAM Network Implementation Guide* and "Session Awareness (SAW) Data Filter (MVS)" in the *VTAM Resource Definition Reference* for details on how to code a SAW data filter.

Message IST530I or IST1278I Received with Pending DSRLST Condition

Problem Statement

A DSRLST PENDING message is received. Message IST530I or IST1278I is issued with CD DSEARCH PENDING FROM netid TO netid.

Common Symptoms

Message IST530I or IST1278I is issued for the application.

Probable Cause

- The ADJSSCP table was not coded; the ADJSSCP table is coded incorrectly; or the IOINT value is too low.
- The start option DYNASSCP and the ADJSSCP table are not correctly tuned.

User Response

To identify the ADJSSCP, enter (with or without a NETID operand):

```
D NET,ADJSSCPS
```

To determine the current value of IOINT, enter:

```
D NET,VTAMOPTS,OPTIONS=IOINT
```

To identify the ADJSSCP, enter (with or without a NETID operand):

```
D NET,ADJSSCPS
```

To determine the DYNASSCP value, enter (and note the DYNASSCP value specified):

```
D NET,VTAMOPTS,OPTIONS=*
or
D NET,VTAMOPTS,OPTIONS=DYNASSCP
```

Message IST663I (Sense Code 08210000) and cannot logon to CICS

Problem Statement

Sessions cannot logon to CICS.

Common Symptoms

The message **IST663I CINIT REQUEST FROM** *adjnode* **FAILED, SENSE=***08210000* is received.

Probable Cause

When running CICS with AUTO-INSTALLATION, the terminal definition in the terminal control table terminal entry (TCTTE) must match the VTAM LOGMODE definition statement for the device.

User Response

Either change the VTAM LOGMODE definition statement to match the CICS TCTTE, or code LOGMODE=0 in the TCTTE. Adding LOGMODE=0 to the TCTTE forces CICS to use VTAM's LOGMODE definition statement for this session.

Message IST663I (Sense Code 08610000) and IST264I Received for Undefined COS Entry

Problem Statement

A required COS entry is UNDEFINED.

Common Symptoms

The following messages are received:

```
IST663I  request REQUEST FAILED, SENSE=08610000
IST264I  REQUIRED COS luname UNDEFINED
MVS HASP208  LOSTTERM SCHEDULED SNA, VTAM, 14
MVS JSX026   J003, RTNCD 1012 REQSESS/TERMSESS OPEN OPNSEC FAILED
         SENSE 08570002
```

Message IST891I may be issued with the IST663I message group and provides information about the identity of the node(s) involved.

Probable Cause

An incorrect COS table was referenced. The NetView program also has a COS table, and the NetView program library was concatenated in front of the VTAM library, causing the wrong table selection.

User Response

To ensure that you are using the correct table, enter:

```
D NET,ID=resourcename
```

Check the library search order to ensure that there are no duplicate table names. Reassemble the table, and check the condition code(s). If the condition code received is what you expected, relink it to the table.

Message IST718I and IST719I Received when Activating a CDRM

Problem Statement

The messages IST718I and IST719I are received during the activation of a CDRM.

Common Symptoms

The following messages are displayed:

```
IST718I ADDRESS INVALID FOR NETID=cdrmnetid CDRM=cdrmname CODE=X'code'
IST719I SUBAREA subarea ELEMENT el
```

Probable Cause

The message is usually a symptom of a duplicate definition for a network address.

The duplicate may have been defined using the SUBAREA and ELEMENT parameters in another CDRM definition or in a GWPATH definition in a gateway NCP.

The duplicate may have been defined using the ADJNETSA and ADJNETEL parameters in another CDRM definition or in a GWPATH definition.

User Response

If the duplicate network cannot be found by inspecting other definitions, run a VTAM internal trace with OPT=(NRM,MSG). When the trace is completed and IST718I and IST719I have been issued, use the console DUMP command to dump VTAM's region and CSA.

- Locate the MSG entry for the IST718I message in the trace.
- Before the message entry there should be an SRTF entry with a non-zero return code, usually a 04. This SRTF entry points either directly or indirectly to the duplicate.
- See "SRTx Entry for SRTADD, SRTCHG, SRTDEL, SRTFIND Macros" on page 794.
- The SRT entry address field points to an SRT entry that can be located in the dump of VTAM. This SRT entry plus X'10' points to the definition that has the duplicate network address.

Messages IST154I, IST561I, IST999E, IST566I, and IST930I Received for APPL-APPL Storage Expansion Failure

Problem Statement

If APPL-APPL sessions are not paced at the session level, storage expansion failures can occur with messages IST154I, IST561I, IST999E, IST566I, and IST930I. The job entry subsystem (JES) has experienced this failure.

Common Symptoms

The following messages are symptoms of storage expansion failures.

```
IST154I  EXPANSION FAILED FOR LFBUF OR IOBUF BUFFER POOL
IST561I  STORAGE UNAVAILABLE
IST999E  VTAM MESSAGE LOST-INSUFFICIENT STORAGE
IST566I  STORAGE UNAVAILABLE xxxx SUBPOOL xxx
IST930I  LU-LU SESSION USING 15% OF IOBUF
```

Probable Cause

If an APPL-APPL session is not paced at the session level, there is no limit to the number of VTAM I/O buffers that the session can use.

User Response

Code VPACING operands on the APPL definition statements for both applications, and code a non-zero value for the SSNDPAC parameter on the LOGMODE operand for the secondary LU. To verify pacing for the session, start a buffer trace with ID=APPLID specified before you start the APPL-APPL session. The BIND RU will contain the pacing values for the session.

Missing VTAM Trace Records (MVS)

Problem Statement

The expected output data is missing from a VTAM trace that was run with GTF active.

Common Symptoms

There is no VTAM data, missing VTAM data, or unwanted data in the GTF trace data set.

Probable Cause

When TRACE=USR is specified, GTF collects all USR events issued in the MVS system.

User Response

To select the events you want to trace, specify USRP on the GTF macroinstruction and select the desired event identifiers (EIDs) as shown in the following examples:

VTAM buffer EIDs—FEF FF1 FF0 (EFEF EFF1 EFF0)

VTAM line trace EIDS (not formatted by GTFTRACE)—FE4 FF2 (EFE4 EFF2)

VTAM internal trace EID—FE1 (EFE1).

See "Activating Network Traces" on page 349 for more information.

Note: To prompt the system for VTAM records, specify USRP in the parameter field of the GTF procedure. You should code a GTF procedure that is used by VTAM only. If you do not, you will get GTF USR output that contains unwanted records.

Partial Input for TGET Received by TSO (MVS)

Problem Statement

A TSO application program receives partial input for the TGET macroinstruction.

Common Symptoms

The TSO application program does not receive the entire data stream buffer from a device. A partial buffer from a device will cause the application to enter a wait state. If the host application program then issues a second TGET, the second section of the buffer is returned to the host application program before processing for the first TGET is completed.

Probable Cause

The TSO application issued the set fullscreen mode (STFSMODE) macroinstruction without specifying the NOEDIT option. The error occurs most often after the application program sends a read partition query (RPQ) to the device. Many newer devices return the attribute byte X'1E' that is returned in the RPQ entry. TSO interprets the X'1E' as an end-of-input field mark. The NOEDIT option of the STFSMODE macroinstruction prevents TSO VTAM from validating the input data. This causes the entire buffer to be returned to the application program.

User Response

Verify the options on the STFSMODE macroinstruction. If STFSMODE is correct, see "Incorrect Output Problems" on page 121 for more information.

Sense Code 08010000 Received with CINIT Failure

Problem Statement

A CINIT request fails with the sense code 08010000 if an application rejects a terminal logon request by issuing the CLSDST macroinstruction.

Common Symptoms

The following message is displayed:

```
IST663I request REQUEST FAILED, SENSE=08010000
```

The logon from a terminal fails with the USS message USSMSG07.

Probable Cause

When an SLU logs on to an application, VTAM builds a CINIT RU and schedules the LOGON exit routine for the application PLU. If the application is not prepared to accept a session with this SLU, it rejects the logon by issuing a CLSDST macroinstruction. If the application does not supply sense code information on the CLSDST, VTAM builds a negative CINIT response with the sense code 08010000. In many cases, the application will also issue a message indicating the reason for the logon rejection.

User Response

Check the message log for a message indicating a failure for this application. Run a buffer trace on the application name to see if the CINIT passed to the application. The VTAM internal trace with the API option contains data about the LOGON exit and the CLSDST macroinstruction.

Sense Code 081C0008 Received on ACTLINK (VM)

Problem Statement

Activation of an I/O device fails with sense code 081C0008.

Common Symptoms

The following message is displayed:

```
IST380I ERROR FOR ID = nodename - REQUEST: ACTLINK, SENSE: 081C0008
```

Probable Cause

The I/O device being activated was not defined to the VTAM virtual machine.

User Response

1. To determine that the device exists, from the VTAM console enter:

   ```
   CP QUERY VIRTUAL vaddr
   ```

   `vaddr` is the virtual address for the device being activated.

2. From a class B virtual machine enter:

   ```
   CP_ATTACH raddr vtam_userid vaddr
   ```

   `raddr` is the real address of the device being activated.

   `vtam_userid` is the VTAM virtual machine's userid.

   `vaddr` is the virtual address that VTAM associates with the device, that is, the address specified by the CUADDR operand on the LOCAL, PCCU, PORT, or PU definition statement or the ADDRESS operand on the LINE definition statement.

Notes:

1. A permanent attachment may be made via the DEDICATE directory statement.
2. See Table 80 on page 1165 to determine what book describes the DEDICATE directory statement and the CP ATTACH and CP QUERY commands.

Sense Code 081C000C Received on ACTLINK (VM)

Problem Statement

Activation of an I/O device fails with sense code 081C000C.

Common Symptoms

The following message is displayed:

```
IST380I ERROR FOR ID = nodename - REQUEST: ACTLINK, SENSE: 081C000C
```

Probable Cause

VTAM found the device to be incorrectly defined.

VTAM uses a CP Diagnose x'24' to determine the device type and characteristics. Table 2 contains values that VTAM checks following a Diagnose x'24' for each CUADDR defined for each type of major node.

Table 2 (Page 1 of 2). Values Checked Following a Diagnose x'24'

VTAM Major Node	LNCTL	Type	DEVCLAS[1]	DEVTYPE[1]	DEVFTR[1]
VBUILD TYPE=CA	LNCTL=CTCA	CTCA	x'02'	x'80'	N/C
VBUILD TYPE=CA	LNCTL=NCP	37x5	x'02'	x'40'	N/C
VBUILD TYPE=CA	LNCTL=SCLC	ICA	x'80'	x'08'	N/C

Table 2 (Page 2 of 2). Values Checked Following a Diagnose x'24'

VTAM Major Node	LNCTL	Type	DEVCLAS[1]	DEVTYPE[1]	DEVFTR[1]
VBUILD TYPE=CA	LNCTL=BSC	ICA	x'80'	x'40,x'80'	N/C
VBUILD TYPE=XCA	N/A	3172	x'02'	x'80'	N/C
VBUILD TYPE=LAN	N/A	ICA	x'02'	x'80	x'40
LBUILD	N/A	3270	x'40'	x'01'—x'80'	N/C
VBUILD TYPE=LOCAL	N/A	3x74	x'02'	x'01,x'40'	N/C
VBUILD TYPE=LOCAL	N/A	3820	x'10'	x'4F'	N/C
VBUILD TYPE=LOCAL	N/A	3821	x'10'	x'4E'	N/C
PCCU	N/A	37x5	x'02'	x'40'	N/C
VBUILD TYPE=PACKET	N/A	ICA	N/C	N/C	N/C

Notes:

1. If a real device exists, then DEVCLAS, DEVTYPE and DEVFTR will be RDEVCLAS, REDVTYPE and RDEVFTR. If a real device is not presently attached, then DEVCLAS, DEVTYPE AND DEVFTR will be VDEVCLAS, VDEVTYPE and VDEVFTR.
2. RDEVFTR of x'40' on CTCA indicates that the device is a 3088.
3.

N/A	not applicable
N/C	VTAM does not check this field.
—	indicates a range of hexidecimal values
,	indicates multiple specific values

User Response

Run DIAG24 EXEC to determine the type and characteristics of a device. DIAG24 EXEC should be run from a CMS virtual machine to which the device in question has been temporarily attached by entering `ATTACH rdev_ userid`. DIAG24 EXEC will not run under GCS.

If the values displayed do not match what VTAM requires, you must modify HCPRIO to achieve the correct values. See Table 80 on page 1165 to determine what book contains more information on HCPRIO.

Figure 1 on page 23 shows sample output from DIAG24 EXEC.

DIAG24 EXEC:

```
/*************************************************************/
/* Function:                                                 */
/*    The following sample exec issues a CP Diagnose x'24'   */
/*    to determine the device type and characteristics. It   */
/*    then displays the output. Must be run under a CMS      */
/*    virtual machine.                                       */
/* Parameter:                                                */
/*    Hexadecimal virtual address of the device. The device  */
/*    must be attached to the CMS virtual machine running    */
/*    this EXEC.                                             */
/*************************************************************/
parse upper arg vaddr
if strip(vaddr)='' | ¬datatype(vaddr,'X') then
```

```
  do
    say 'Hexadecimal virtual address must be specified.'
    exit 99
  end
diag24=diagrc(24,vaddr)
cc=substr(diag24,11,1)
diag24=c2x(substr(diag24,17))
ry=left(diag24,8)
ryplusone=substr(diag24,9,8)
say 'Diag x''24'' condition code:' cc
select
  when cc=0 then
    do                      /* Real device exists           */
      say 'Ry: x"'ry'"  Ry+1: x"'ryplusone'"'
      say 'RDEVCLAS: x"'substr(ryplusone,1,2)'"',
          'RDEVTYPE: x"'substr(ryplusone,3,2)'"',
          'RDEVTFR: x"'substr(ryplusone,5,2)'"'
    end                     /* Real device exists           */
  when cc=2 then
    do                      /* Virtual device exists, no real*/
      say 'Ry: x"'ry'"'
      say 'VDEVCLAS: x"'substr(ry,1,2)'"',
          'VDEVTYPE: x"'substr(ry,3,2)'"',
          'VDEVFTR: x"'substr(ry,5,2)'"'
    end                     /* Virtual device exists, no real*/
  when cc=3 then
    do                      /* Device not attached          */
      say 'Device:' vaddr 'is not ATTACHed to this virtual',
          'machine.'
    end                     /* Device not attached          */
  otherwise
end
say 'See VM/ESA CP R2 Programming Services under appendix C,'
say '"Data Areas Used by DIAGNOSE Codes" in topic C.0'
exit
```

```
diag24 009

Diag x'24' condition code:0
Ry: x"80000000" Ry+1: x"40010350"
RDEVCLAS: x"40" RDEVTYPE: x"01" RDEVFTR: x"03"
See VM/ESA CP R2 Programming Services under appendix C,
"Data Areas Used by DIAGNOSE Codes" in topic C.0
Ready; T=0.01/0.01 15:30:53
```

Figure 1. Sample DIAG24 EXEC Output

In the sample shown in Figure 1, `RDevice class of x'40'` and `RDevice type of x'01'` indicates that this device would be valid for an LU under an LBUILD major node.

Sense Code 083500xx or 08210000 Received with BIND Failure

Problem Statement

BIND failure occurs with sense code 083500xx or 08210000.

Common Symptoms

The following messages are displayed in response to a terminal logon request:

```
IST663I  BIND REQUEST FAILED, SENSE=083500xx
  or
IST663I  BIND REQUEST FAILED, SENSE=08210000
 and
USSMSG07 luname UNABLE TO ESTABLISH SESSION-BIND FAILED
         WITH SENSE sense
```

Probable Cause

The sense codes indicate that the BIND contains parameters that are not valid. The sense code 08210000 gives no further explanation. Sense code 083500xx supplies an index (xx) into the BIND that identifies the bytes that the BIND receiver cannot interpret.

VTAM extracts BIND parameters from the LOGMODE entry associated with the logon, based on the LOGON command, the USSPARM PARM=LOGMODE from the USSTAB, or the default on the LU definition specified by DLOGMOD. The source of the BIND parameters can also be the application, which may override many of the parameters supplied by VTAM when the OPNDST macroinstruction is issued. When the requested LOGMODE cannot be found, VTAM may use a default LOGMODE (ISTCOSDF), which may contain session parameters that are unacceptable to the application.

User Response

Run a buffer trace on the application name for a terminal session logon to an application that is rejected by the BIND with a sense code of 083500xx or 08210000. This traces the CINIT request, which includes the VTAM supplied parameters. If the BIND that follows the CINIT request does not match these parameters, they were changed by the application. The documentation for the rejecting LU should list its required BIND parameters.

In order to prevent VTAM from using the default LOGMODE (ISTCOSDF), ensure that the requested LOGMODE is defined in the specified LOGMODE table. See the *VTAM Network Implementation Guide* and "Subarea Class-of-Service Table" in the *VTAM Resource Definition Reference* for more information on ISTCOSDF.

Sense Code 800A0000 or No Message, and Sessions End Unexpectedly

Problem Statement

A session ended unexpectedly and either no message is received or an exception request (EXR) with a sense code of 800A0000 flows to the destination LU.

Common Symptoms

Upon receiving the sense code 800A0000, the LU might return the code in a response. Some LUs will include the code in an UNBIND.

Probable Cause

If a path information unit (PIU) is too large to be passed from one PU type 4 or 5 to another, an exception request (EXR), containing sense code 800A0000 and up to three bytes of the RU, may be generated. See Table 80 on page 1165 to determine what book describes the building of the EXR.

User Response

For VTAM/NCP nodes in the session path, the following definition values should be checked for each configuration used:

- VTAM to channel-attached NCP: VTAM will take the smaller of the following two values:
  - MAXDATA value on the PCCU definition statement (or on the LINE definition statement for a channel-attached NCP)
  - Value sent in the XID of the maximum PIU size for the NCP. This number will be the product of the BFRS value from the BUILD definition statement and the TRANSFER

value from the channel adapter LINE definition statement for 3745 or 3720 with V5 NCP or from the BUILD definition statement for other NCPs.

- Channel-attached NCP to VTAM: The product of the MAXBFRU value from the HOST definition statement (or from the LINE definition statement for a channel-attached NCP) and the IOBUF size from the VTAM start options. This value will be the maximum size that can flow from the NCP to the host.
- VTAM-to-VTAM connection across a channel-to-channel interface:
  - If both VTAMs have the CTCA enhancement: The product of the MAXBFRU value from the CTCA LINE definition statement and the IOBUF size from the VTAM start options
  - If one or neither of the VTAMs have the CTCA enhancement: The product of the MAXBFRU value from the LINE macro and the IOBUF size from the VTAM start options
- NCP to link-attached NCP: The product of the TRANSFR value on the LINE definition statement and the BFRS value on the BUILD definition statement
- NCP to link-attached VTAM: The product of the MAXBFRU value from the CA LINE definition statement and the IOBUF size from the VTAM start options.

Notes:

1. The definition statements for all PU type 4 or 5 nodes on the session path must be checked because any PU type 4 or 5 can change the PIU into an 800A0000 exception request.
2. The information in this problem description is from information APAR II03990.

Sense Codes 10030000 and 08090000 Received when Activating an NCP

Problem Statement

Some resources fail to activate correctly when a new NCP is activated.

Common Symptoms

VTAM commands return sense codes 10030000 and 08090000.

Probable Cause

The resource resolution table (RRT) created when a new NCP was generated did not replace the previous RRT, and the old RRT is still being referred to by the system.

User Response

When you generate a new NCP, either rename the new NCP or use another method to ensure that the old RRT is replaced with the new RRT.

Session Failure with Sense Code 0888000x

Problem Statement

An attempt to establish a session fails with sense code 0888000x in an intermediate VTAM along the session setup path.

Common Symptoms

The session establishment is terminated.

Probable Cause

The intermediate VTAM that set the 0888000x sense codes is operating with NQNMODE=NAME or is a pre-V4 VTAM and therefore cannot define multiple resources with the same name, even if the network identifiers are different.

User Response

Change the intermediate domain to operate with NQNMODE=NQNAME to allow definition of multiple resources with the same name and different network identifiers, or reroute the session through another path.

Common Problems in APPN Networks

Table 3 includes a brief description of several common problems that occur in APPN networks. For additional information, go to the page indicated.

Table 3 (Page 1 of 2). Index of Common Problems in APPN Networks

Problem	See
Best path not taken for the session	Page 38
Message IST489I received during session takeover	Page 28
Message IST264I (required COS entry undefined) and message IST663I (sense code 08610000)	Page 29
Message IST663I (sense code 08610000) and message IST264I (required COS entry undefined)	Page 29
Messages IST1097I and IST1280I (sense code 08A00005) received with CP-CP session failure	Page 29
Messages IST1110I and IST1112I issued during CP-CP session activation failure	Page 30
Messages IST1110I and IST1113I issued during CP-CP session activation failure	Page 30
Messages IST1110I and IST1246I issued during CP-CP session activation failure	Page 31
Messages IST1110I, IST1246I, and IST1280I (sense code 80050000) issued during CP-CP session activation failure	Page 31
Messages IST1110I and IST1280I (sense code 08B50000) issued during CP-CP session activation failure	Page 32
Messages IST1110I and IST1280I (sense code 08910006) issued during CP-CP session activation failure	Page 32
Messages IST1110I, IST1280I (sense code 101E000A), and IST1356I issued during CP-CP session activation failure	Page 31
Message IST1272I received during session takeover	Page 28
Resource not found but resource exists in network	Page 33
Sense code 081C0008 Received on ACTLINK VM	Page 21
Sense code 081C000C Received on ACTLINK VM	Page 21
Sense code 08210002 issued during a session activation failure	Page 33
Sense code 0821000A issued during a session activation failure	Page 34
Sense code 083B0001 issued and session lost	Page 35
Sense code 08610000 (message IST663I) and message IST264I (required COS entry undefined)	Page 29
Sense code 087D0001 issued during a session activation failure	Page 34
Sense code 087D000A issued and session lost	Page 35
Sense code 08910006 and messages IST1110I and IST1280I issued during CP-CP session activation failure	Page 32
Sense code 08A00005 received unexpectedly with CP-CP session failure	Page 29
Sense code 08B50000 and messages IST1110I and IST1280I issued during CP-CP session activation failure	Page 32
Sense code 10145046 (AS/400*) issued during a session activation failure	Page 35
Sense code 101E000A and messages IST1110I, IST1280I, and IST1356I issued during CP-CP session activation failure	Page 31
Sense code 80050000 and messages IST1110I, IST1246I, and IST1280I issued during CP-CP session activation failure	Page 31
Sense code 80130000 issued during a session activation failure	Page 36

Table 3 (Page 2 of 2). Index of Common Problems in APPN Networks

Problem	See
Sense code 80130104 and path problems	Page 199
Sense code 80140001 issued during a session activation failure	Page 36
Sense code 80140002 issued during a session activation failure	Page 37
Sense code 80140005 issued during a session activation failure	Page 37
Session did not take the best path	Page 38
Session established with non-local instead of local application program	Page 38
Session lost with sense code 083B0001	Page 35
Session lost with sense code 087D000A	Page 35
Storage problem	Page 105

Descriptions of Common Problems in APPN Networks

This section includes examples of common problems in APPN networks. See Table 3 on page 27 for an index of these problems.

Message IST489I or IST1272I Received during Session Takeover

Problem Statement

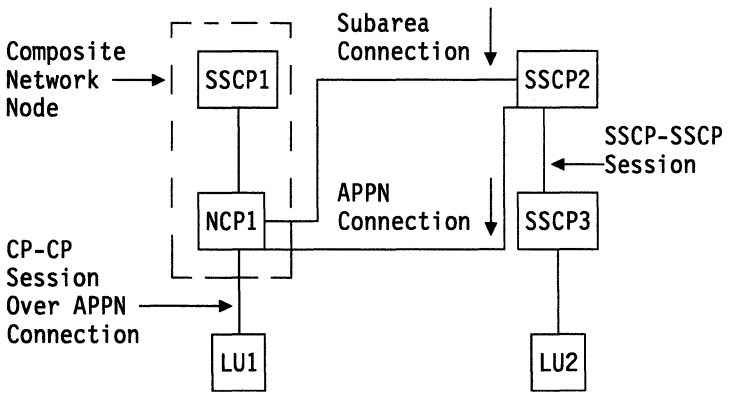
Message IST489I or IST1272I is received, indicating that VTAM cannot define a resource.

Common Symptoms

Message IST489I or IST1272I is received for a resource during takeover processing. The resource can be a dependent LU.

Probable Cause

A scenario similar to the following can cause this problem:



- Boundary function CP-CP sessions exist between SSCP1 and SSCP2. There is also a FID4 connection between SSCP2 and NCP1.
- There is an SSCP-SSCP session between SSCP2 and SSCP3.
- Dependent LU1, connected from NCP1, is owned by SSCP1.
- An LU-LU session is between LU1 and LU2. LU1 is known to SSCP2 as an APPN LU through the APPN connection between SSCP1 and SSCP2 through NCP1.

- The command, VARY INACT TYPE=G, was entered from SSCP1 which caused session takeover of the line from SSCP1 to SSCP2. SSCP2 owns LU2, the independent LU.

The failure, triggering message IST489I or message IST1272I, occurred because LU1 is known to the SSCP2 host as an independent APPN LU. SSCP2 cannot own the independent LU and the dependent LU at the same time.

User Response

The independent LU is freed when the last session ends. After this occurs, you must activate the dependent LU before it can be enabled.

Message IST663I (Sense Code 08610000) and IST264I Received for Undefined COS Entry

Problem Statement

A required COS entry is UNDEFINED.

Common Symptoms

The following messages are received:

```
IST663I  request REQUEST FAILED, SENSE=08610000
IST264I  REQUIRED COS luname UNDEFINED
MVS HASP208  LOSTTERM SCHEDULED SNA, VTAM, 14
MVS JSX026   J003, RTNCD 1012 REQSESS/TERMSESS OPEN OPNSEC FAILED
        SENSE 08570002
```

Message IST891I may be issued with the IST663I message group and provides information about the identity of the node(s) involved.

Probable Cause

An incorrect COS table is referenced. The NetView program also has a COS table, and the NetView program library was concatenated in front of the VTAM library, causing the wrong table selection.

User Response mode-to-class-of-service (COS)

To ensure that you are using the correct table, enter:

```
D NET,ID=resourcename
```

Check the library search order to ensure that there are no duplicate table names. Reassemble the table, and check the condition code(s). If the condition code received is what you expected, relink it to the table.

Messages IST1097I and IST1280I (Sense Code 08A00005) Received with CP-CP Session Failure

Problem Statement

CP-CP session failure occurs with sense code 08A00005.

Common Symptoms

The following message group is received:

```
IST1097I   CP-CP SESSION WITH partner_cpname TERMINATED
IST1280I   SESSION TYPE = CONLOSER - SENSE = 08A00005
IST314II   END
```

Probable Cause:

If CP-CP sessions have been deactivated with this sense code, it is likely that the topology database update (TDU) flowing between the two nodes has been lost due to a storage depletion condition on either the sending or receiving end of the TDU flow.

User response

If the CP-CP sessions do not come back up automatically, enter:

```
V ACT,ID=partner_cpname,IDTYPE=CP
```

If VTAM is experiencing temporary storage allocation problems, you might want to wait for the condition to clear before attempting to restart the session.

When the CP-CP session is restarted, TDUs will be exchanged so that the missing information in the lost flow will be recovered.

Messages IST1110I and IST1112I Received during CP-CP Session Activation Failure

Problem Statement

End node operator's attempt to activate CP-CP session pair by activating adjacent network node fails with issuance of messages IST1110I, IST1112I, and IST314I at the end node.

Common Symptoms

Activation of new CP-CP session pair with adjacent CP specified in message IST1110I is terminated. Messages IST1110I, IST1112I, and IST314I are displayed.

Probable Cause

While attempting to activate a contention winner CP-CP session with the network node specified in IST1110I, the end node determined that it already had an active CP-CP session pair with a different network node. If an end node already has a network node server, it will not accept a new CP-CP session with another network node.

User Response

Before attempting to activate a CP-CP session pair between an end node and a network node, enter the D NET,SESSIONS,LIST=ALL command at the end node to verify that no CP-CP sessions with a network node already exist.

Messages IST1110I and IST1113I Received during CP-CP Session Activation Failure

Problem Statement

Attempt by operator of end node to activate CP-CP session pair with adjacent network node by activating adjacent network node fails with issuance of messages IST1110I, IST1113I, and IST314I.

Common Symptoms

Activation of new CP-CP session pair with adjacent CP specified in message IST1110I is terminated. Messages IST1110I, IST1113I, and IST314I are displayed at the end node.

Probable Cause

The operator entered from an end node a V NET,ACT,ID=cpname command, where cpname is also an end node. CP-CP sessions are not permitted between end nodes.

User Response

Make sure that the start lists for the two nodes do not both specify end node as the type of node being started.

Messages IST1110I and IST1246I Received during CP-CP Session Activation Failure

Problem Statement

Attempt to activate CP-CP session pair between end node and network node fails after activation of CP-capable link.

Common Symptoms

Activation of new CP-CP session pair with network node specified in message IST1110I is terminated. Message IST1110I displayed along with message IST1246I at the end node.

Probable Cause

While in the process of bringing up the contention winner CP-CP session, the end node determined that the network node named in IST1110I is not explicitly named in the end node's network node server list and that there is no nameless entry in the network node server list.

User Response

Either perform an operator activation of the CP-CP session by entering V NET,ACT,ID=adjacent_cpname at the end node or modify the network node server list to include either an explicit entry for the desired network node or a nameless entry.

Messages IST1110I, IST1246I, and IST1280I Received during CP-CP Session Activation Failure

Problem Statement

Attempt to activate CP-CP session pair between end node and network node fails.

Common Symptoms

Activation of new CP-CP session pair with network node specified in message IST1110I is terminated. Message IST1110I displayed along with message IST1246I at the end node. Message IST1280I displays a sense code of 80050000.

Probable Cause

While in the process of bringing up the conloser CP-CP session, the end node determined that the network node named in IST1110I is not explicitly named in the end node's network node server list and that there is no nameless entry in the network node server list.

User Response

Either perform an operator-activation of the CP-CP session by entering V NET,ACT,ID=adjacent_cp_name at the end node or modify the network node server list to include either an explicit entry for the desired network node or a nameless entry.

Messages IST1110I, IST1280I (Sense Code 101E000A), and IST1356I Received during CP-CP Session Activation Failure

Problem Statement

Attempt to activate CP-CP session pair between end node and network node fails.

Common Symptoms

Activation of new CP-CP session pair with network node specified in message IST1110I is terminated. The operator at the end node sees messages IST1110I, IST1246I, IST1280I, and IST314I. IST1280I displays a sense code of 101E000A.

Probable Cause

The end node's network node server list entry for the network node failed to specify SLUINIT=OPT or, in the absence of an explicit entry for that node, the nameless entry failed to specify SLUINIT=OPT. The network node is probably an AS/400, NS/2, or Personal System/2*

computer, none of which provides the network node server capabilities provided by VTAM Version 4 network nodes. CP-CP sessions between VTAM Version 4 end nodes and such network nodes are allowed only if the end node's network node server list specifies SLUINIT=OPT.

User Response

Modify the network node server list to specify SLUINIT=OPT on either the explicit entry for the desired network node server or on the nameless entry. Activate the modified network node server list definition deck, and then reactivate the session.

Messages IST1110I and IST1280I (Sense Code 08910006) Received during CP-CP Session Activation Failure

Problem Statement

Attempt to activate CP-CP session pair between network nodes in two different network nodes fails.

Common Symptoms

Activation of new CP-CP session pair between this network node and adjacent network node specified in message IST1110I is terminated.

Probable Cause

A CP-CP session pair is not permitted between network nodes located in different networks unless you have specified BN=YES to enable the VTAM border node function. The messages indicate that CP-CP sessions were attempted between two network nodes in different networks.

User Response

Modify the VTAM start lists for the specific nodes so that both start lists specify the same network.

Messages IST1110I and IST1280I (Sense Code of 08B50000) Received during CP-CP Session Activation Failure

Problem Statement

Attempt to activate CP-CP session pair between end node and network node fails with sense code 08B50000, as indicated by message IST1280I.

Common Symptoms

Activation of new CP-CP session pair with adjacent CP specified in message IST1110I is terminated. Message IST1110I is displayed along with message IST1280I, which displays a sense code of 08B50000.

Probable Cause

The sense code indicates that the end node bringing up the contention-loser session does not require a CP-CP session pair with the network node specified in message IST1110I. The end node determined that it has an active CP-CP session with a different network node. If an end node already has a server, it will not accept a new CP-CP session with another network node.

User Response

Before attempting to activate a CP-CP session pair between an end node and a network node, enter the D NET,SESSIONS,LIST=ALL command at the end node to verify that no CP-CP sessions with a network node already exist.

Resource Not Found but Resource Exists in Network

Problem Statement

A resource exists in the network but is not found by a search.

Common Symptoms

The directory services management exit routine either rejects or limits the search scope.

Probable Cause

The resource was not registered to its network node server.

User Response

Register the resource to the network node server. For more information on the directory services management exit routine, see "Directory Services Management Exit Routine" in *VTAM Customization*.

Sense Code 08210002 Received with Session Activation Failure

Problem Statement

Session activation failed with the sense code 08210002.

Common Symptoms

An attempt to establish a session failed with the sense code 08210002 - invalid mode name.

Probable Cause

The sense code indicates that the logon mode name associated with the session request was not found in the table or in the default logon mode table (ISTINCLM).

User Response

Verify that the requested logon mode name is defined as follows:

- In a subarea-only environment, the mode name must be defined in the SSCP associated with the SLU.
- In an APPN-only environment, the mode name must be defined in the origin and destination nodes, as well as the origin and destination node servers if the origin or the destination is owned by an end node.
- In a combined APPN and subarea environment, the mode name must be defined at the APPN node that owns the origin or destination, at the node server if it is an end node, and at the interchange nodes that represent the subarea entry point. If the SLU is owned by a subarea node other than the interchange node representing the subarea entry point, the mode name must also be defined on the owning subarea.

  Note: Since mode table names are not carried on APPN line flows, a user-defined mode table is only used at the SSCP for the SLU. Other nodes defining the mode must define the SLU in the default logon mode table.

  See "What do you get if you take all the defaults?" in the *VTAM Network Implementation Guide* for details on mode to COS resolution in an APPN or in a combined subarea and APPN environment.

Sense Code 0821000A Received with Session Activation Failure

Problem Statement

Session activation failed with the sense code 0821000A.

Common Symptoms

An attempt to establish a session failed and the sense code 0821000A (mode table not found) was returned.

Probable Cause

The sense code indicates that the mode table associated with the LU was not found.

User Response

Verify that the specified table exists, and activate it.

Sense Code 087D0001 Received with Session Activation Failure

Problem Statement

Session activation failed with the sense code 087D0001.

Common Symptoms

An attempt to establish a session failed and the sense code 087D0001 (routing exhausted) was returned. Messages IST894 and IST895 indicate that one of the adjacent SSCPs tried is ISTAPNCP with a failure sense code of 087F0001 (resubmit requested for a request that was already resubmitted).

Probable Cause

Possibly one of the following:

- If messages IST894I and IST895I are issued, one of the adjacent SSCPs was ISTAPNCP with a failure sense code of 087F0001. This indicates that VTAM knows which node owns the LU but is not able to route a directed search to that node to verify the availability of the LU.
- There is no SSCP-SSCP session.
- The half-session control block (HSCB) count is too low in the NCP to handle the number of sessions. A possible solution to this problem is to code a larger value on the ADDSESS keyword of the BUILD definition statement and regen.
- Both sides are using the same SSCP name.

User Response

Verify that a valid search path exists. This includes CP-CP sessions, a subarea path, or both. One possible source of the problem is the absence of a CP-CP session between two nodes that share an active link that is CP-CP capable. If this situation occurs, take one of the following actions:

- Reactivate the CP-CP session.
- Deactivate the link, and reactivate it as a link that is not CP-CP capable. This notifies topology and routing services that the link is no longer available for use in directed search routing.

Sense Code 087D000A or 083B0001 Received and Session Lost during Takeover

Problem Statement

Session lost during takeover with sense code 087D000A or 083B0001.

Common Symptoms

An attempt to take over a switched connection that is defined with ANS=CONTINUE results in a session or sessions being lost. A message states that a BFSESSINFO request failed with the sense code 087D000A (routed through same SSCP twice) or with the sense code 083B0001 (duplicate PCID).

Probable Cause

The problem might be that a connection-network-capable control point (CP) on the connection network does not have a complete system definition.

User Response

If you have a connection network, check the resource definitions on each CP connected to the network. Any connection-network-capable CP must define both of the following:

- Its own connection to the connection network
- Connections to any CPs on the connection network that are not connection-network-capable.

Sense Code 10145046 Received with Session Activation Failure

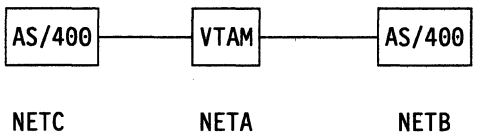
Problem Statement

Session activation failed with the AS/400 sense code 10145046.

Common Symptoms

An attempt was made to establish a session from one AS/400 to another AS/400 across a VTAM network. The session failed to complete, and route selection errors occurred on the initiating AS/400.

Probable Cause



The AS/400 (NETC) is sending VTAM (NETA) a CDINIT that specifies a session in a third network that is not supported in this release of VTAM. VTAM rejects the session, and the AS/400 returns the sense code 10145046 to VTAM.

User Response

Verify that APPN sessions across three networks are not present in your system. Sessions across three networks using APPN links are not supported by VTAM unless you have specified BN=YES to enable the VTAM border node function.

Sense Code 80130000 Received with Session Activation Failure

Problem Statement

Session activation failed with the sense code 80130000.

Common Symptoms

An attempt to establish a session failed, and the sense code 80130000 (class of service not available) was returned.

Probable Cause

This sense code indicates that the subarea class of service (COS) is not known. (In contrast, sense code 80140002 is issued by topology and routing services and indicates that the APPN COS is not known.)

User Response

Verify that the node issuing the sense code has a usable subarea class of service for the mode associated with the session request. This node is usually the primary logical unit (PLU) host, an intermediate gateway VTAM, or a gateway VTAM.

Message IST891I may be issued and provides the name of the failing node. See the description of message IST891I in *VTAM Messages and Codes* for additional information.

This problem can occur when a mode table is copied from one node to another, and the subarea classes of service specified by the table no longer map to valid COS names defined at that node.

Sense Code 80140001 Received with Session Activation Failure

Problem Statement

Session activation failed with the sense code 80140001.

Common Symptoms

An attempt to establish a session failed, and the sense code 80140001 (no route exists) was returned. This sense code indicates that no APPN route was found from the origin node to the destination node that meets the requirements of the requested class of service.

This can be due to any of several possible causes.

Probable Cause 1

There is not an active APPN route between the origin and destination.

User Response 1

Examine your network configuration to determine whether a valid path does exist. Use the DISPLAY TOPO command to verify that the topology database currently shows the links in the path as operational.

Probable Cause 2

Although a valid APPN path exists, the characteristics of the nodes and links in the operational paths do not meet the requirements of the specified class of service.

User Response 2

Check the following:

1. Verify that the mode name specified on the request maps to the intended class of service.
2. Examine the LINEROW and NODEROW operands in the class of service definition to determine the allowable ranges for the link and node characteristics.
3. Use the DISPLAY TOPO command to view the characteristics of the nodes and TGs in the likely paths. Look for problems such as:

a. Nodes in the path are congested or have route resistance values outside the limits set by the class of service.

b. The COS definition required secure links, but no path exists consisting exclusively of secure TGs.

c. High capacity (speed) was required by the COS definition, but no path exists in which all of the links are fast enough to meet the specified minimum capacity.

Probable Cause 3

The destination exists in a subarea network, or in another APPN network that is accessed through a subarea network, but paths acceptable to the specified class of service definition do not exist to all interchange nodes representing subarea entry points that can be used to reach the destination.

User Response 3:

As specified in "APPN and Subarea Class of Service Resolution" in *VTAM Network Implementation Guide*, if a destination can be reached by exiting one APPN network through two or more different interchange nodes, paths acceptable for the class of service to be used must be available to all of those possible exist interchange nodes. Verify this by examining the COS definition and the characteristics of the paths to the possible exit interchange nodes.

Sense Code 80140002 Received with Session Activation Failure

Problem Statement

Session activation failed with the sense code 80140002.

Common Symptoms

An attempt to establish a session failed, and the sense code 80140002 (invalid APPN COS name received) was returned.

Probable Cause

The sense code indicates that the APPN COS definition was not found. The definition might not exist at a node that is performing mode-to-COS resolution, or the mode-to-COS mapping specified in the mode table might not be mapping to the intended COS name.

User Response

Examine the mode definition to determine the APPN COS name. Verify that this definition exists in the VTAMLST members at the nodes that resolve the mode to an APPN class of service. Activate the member to be sure that the definition is active. If APPN COS substitution is enabled (by specifying the APPNCOS start option), verify that the COS it specifies has been activated.

Sense Code 80140005 Received with Session Activation Failure

Problem Statement

Session activation failed with the sense code 80140005.

Common Symptoms

An attempt to establish a session failed, and the sense code 80140005 (RSCV exceeds the maximum length) was returned.

Probable Cause

The sense code indicates that the number of hops between the origin and destination nodes was too large, so the attempt to build the Route Selection control vector failed.

User Response

Examine your network configuration to determine how many hops would be expected in the best route for the requested APPN class of service. If the number of hops within a single APPN

network is greater than six, then you may need to provide a more direct origin to the destination path.

Session Established with Non-Local instead of Local Application Program

Problem Statement

The session was intended to be established with a local application program; however, it was established with a non-local application program.

Probable Cause

The local application program is not yet active, and a local application program served by another node is registered to the non-local application program's network node server.

User Response

Be sure that the local application program is active before you attempt to logon to it.

Session Did Not Take the Best Path

Problem Statement

The session took a path through the subarea network even though a better path existed through the APPN network, or the session took a path through the APPN network even though a better path existed through the subarea network

Probable Cause

The SORDER operand or SSEARCH operand is coded with a value that indicates that the subarea network is to be searched before the APPN network is searched or that the APPN network is to be searched before the subarea network is searched.

User Response

Adjust the values on the SORDER and SSEARCH operands to suit your network.

Common Symptoms and Associated VTAM Problem Types

If your problem was not described in "Common Problems in Subarea Networks" on page 6 or "Common Problems in APPN Networks" on page 27, find the symptom you are experiencing in Table 4. The symptoms are listed alphabetically. Match your symptom to the appropriate VTAM problem type and go to the page indicated.

Table 4 (Page 1 of 3). Index of Problem Symptoms and Associated VTAM Problem Types

Symptom	Problem Type	See
Abend message	Abend	Page 58
Activating network nodes takes too long	Performance	Page 101
Application program cannot terminate	Wait	Page 67
Application programs and terminals cannot communicate	Wait or Loop	Page 67 Page 86
Application program reports an unexpected return or sense code	Incorrect Output or Message	Page 96 Page 93
Batch application program fails to complete	Wait	Page 67
Book is missing information or has wrong or ambiguous information	Documentation	Page 108
Books contradict each other	Documentation	Page 108
Command is not completed	Wait or Incorrect Output	Page 67 Page 96
Commands cannot be entered on system console	Loop	Page 86
Commands take too long to complete	Performance	Page 101
Cursor is in the wrong position. This is probably an application program or VTAM definition error, such as using an incorrect logmode definition.	Incorrect Output	Page 96
Deactivating network nodes takes too long	Performance	Page 101
VM DTI error message	Message	Page 93
VM DTI message is wrong or formatted improperly	Message or Incorrect Output	Page 93 Page 96
VM DTIS99I message appears	See chapter 131	Page 176
Dump file contains a VTAM or TSO/VTAM dump	Abend or Program Check	Page 58 Page 63
Error message	Message	Page 93
Hung session, LU, or terminal	Incorrect Output	Page 96
Hung system	Wait	Page 67
MVS IKT error message	Message	Page 93
MVS IKT message is wrong or formatted improperly	Message or Incorrect Output	Page 93 Page 96
VM Initialization problem (VSCS only)		Page 143
VM Internal error message (VSCS only)		Page 176
IST error message	Message	Page 93
IST message is wrong or formatted improperly	Message or Incorrect Output	Page 93 Page 96

Table 4 (Page 2 of 3). Index of Problem Symptoms and Associated VTAM Problem Types

Symptom	Problem Type	See
IST252I message appears	Program Check	Page 63
IST413I message appears	Abend	Page 58
Keyboard locks unexpectedly	Incorrect Output	Page 96
Logo problem	Incorrect Output	Page 96
LOGON takes too long to complete	Performance	Page 101
LOGREC entries indicate an abend	Abend	Page 58
LOGREC fills with repeated entries	Loop or hardware	Page 86
Message is wrong or formatted incorrectly	Message or Incorrect Output	Page 93 Page 96
Message from application program	Incorrect Output or Message	Page 96 Page 93
Message is sent to the wrong console	Incorrect Output	Page 96
Message repeats continuously	Loop	Page 86
Message text does not explain a condition	Message	Page 93
Message is missing text	Message or Incorrect Output	Page 93 Page 96
Output data is formatted incorrectly. This is probably an application program or VTAM definition error, such as using an incorrect logmode definition.	Incorrect Output	Page 96
Path problem	Performance	Page 199
Performance is degraded after a network outage	Performance	Page 101
Printers stop	Loop	Page 86
Program check message	Program Check	Page 63
PSWs point to a VTAM address	Loop	Page 86
Response time is slow	Performance	Page 101
Routing information is wrong	Incorrect Output	Page 96
Storage message IST154I or IST561I-IST566I	Storage	Page 105
VSE SYSREC entries indicate a program check	Program Check	Page 63
VSE SYSREC fills with repeated entries	Loop or Hardware	Page 86
VSE System enters disabled wait state	Program Check or Incorrect Output	Page 63 Page 96
System functions stop	Loop	Page 86
System light is on; Wait light is off	Loop	Page 86
Tapes stop	Loop	Page 86
Terminal user cannot log on, enter data, or log off	Incorrect Output	Page 96
Terminal user gets unexpected response. This is probably an application program or VTAM definition error, such as using an incorrect logmode definition.	Incorrect Output	Page 96
Terminal user reports incorrect or missing data. This is probably an application program or VTAM definition error, such as using an incorrect logmode definition.	Incorrect Output	Page 96

Table 4 (Page 3 of 3). Index of Problem Symptoms and Associated VTAM Problem Types

Symptom	Problem Type	See
VM Termination problem (VSCS only)		Page 150
Traffic ceases through a network component (BSC link, SDLC link, communication controller, control unit)	Wait	Page 67
VTAM does not work as described in a book	Documentation	Page 108
VTAM is not communicating with system console	Wait or Loop	Page 67 Page 86
VTAM process issues an error message	Message	Page 93

VTAM Internal Trace (VIT) Analysis Tool Problems (MVS) (VM)

This section describes how to diagnose problems that might occur while running the VIT analysis tool and includes the following sections:

- "Checklist for Isolating the Problem"
- "Common Symptoms and Actions" on page 43
- "Documenting an APAR for VIT Analysis Tool Problems" on page 44.

If you are having problems during installation of the tool, refer to "Installing the VTAM Dump Analysis Enhancements and the VIT Analysis Tool" in the *VTAM Network Implementation Guide* for additional information.

Checklist for Isolating the Problem

1. Does the problem exist with an ISPF panel or with the VIT analysis tool?

   The ISPF panel process simply creates the parameter data set, which is the input to the VIT analysis tool.

2. MVS Are any errors noted in the JCL output?

   IBM publishes sample JCL, as described under "Step 2. Set up to Run the Tool" on page 412. However, you can change it to suit your environment. The JCL output indicates whether the tools' entry module, ISTRAFT1, is found. The JCL output contains an error message if the SUMMARY data set cannot be written to (for example, if the wrong DCB information was specified for the SUMMARY data set). Also check the JCL to verify that the right trace tapes or DASD data sets are specified and that multiple tapes are specified in the correct order.

3. Check the SUMMARY data set to determine whether any errors were noticed, such as the wrong DCB information being supplied for the LOG data set.

   The SUMMARY data set tells the input parameters and defaults used. The SUMMARY data set reports time stamps and types of records and VIT entry occurrences found on the trace. The SUMMARY data set should always be created, unless there was a problem with the SUMMARY data set itself, in which case the JCL or REXX output shows what happened.

4. Check the input parameters in the parameter data set with the syntax diagrams.

   The parameter data set is created by either the ISPF panel process or by coding the parameters directly using an editor. For more information on checking syntax diagrams, see Chapter 10, "Using the VIT Analysis Tool (MVS, VM)" on page 411.

5. Check the following output data sets:

   - LOG

     The LOG data set shows whether counters overflowed, whether the trace wrapped, and so on. The LOG data set is always used unless a fatal error prevents the tool from initializing completely. The LOG data set might contain only the title line and description (if a description exists).

- VITEXT

  The VITEXT data set is used only if the VIT extraction function is chosen. It is not used for storage analysis or RU-counting.

- DETAILS

  The DETAILS data set is used only if the storage analysis or RU-counting function is chosen. It is not used for VIT extraction.

- OUTSTAN

  The OUTSTAN data set is used only if storage-analysis-counting function is chosen and only if the outstanding option (to list unmatched allocate entries) is chosen. It is not used for RU counting or VIT extraction.

Common Symptoms and Actions

Use Table 5 to diagnose and correct problems.

Table 5 (Page 1 of 2). VIT Analysis Tool Problems: Common Symptoms and Actions

Symptom	Action

Runs too long	If the tool is taking a very long time to run (several hours):
	• If the MATCH option was specified for storage analysis:
	  – Remove the MATCH option.
	  – Specify only the few pools you are interested in when using the MATCH option.
	• Check to see if the tool is waiting on a tape to be mounted or to access a data set in use by some other job.

No output	If no output appears (for example, no matching VIT entries are found, no RUs are counted, no storage VIT entries are found):
	• Verify that the trace has VIT records (see the SUMMARY data set).
	• Verify that the entries required for the job are on the VIT. (All occurrences of VIT entries are listed at the bottom of the SUMMARY data set.)
	  – Storage analysis requires the SMS VIT entries.
	  – RU counting requires the PIU VIT entries.
	  – VIT extraction origin and destination options work only on PIU VIT entries.
	• Check to see if the VIT is a different level from the VIT analysis tool. For example, the DISP entry is now called DSP. Therefore, if you are extracting all occurrences of VIT option PSS with entry name e'DI*', no matches are found in VTAM V4R2.
	• Do not specify a start or stop time. Note that an event reported on the console can be several seconds off from the GTF time stamp.
	• Use the INTERVAL option to ensure that some output is seen before the job abends or is cancelled.

Same output from a previous date	If the job runs but the output data sets contain data from a previous job, check the DISP parameter. When DISP is NEW and the data set already exists, the batch job runs anyway and then deletes the new data set. A message in the JCL log indicates whether this has happened.

Table 5 (Page 2 of 2). VIT Analysis Tool Problems: Common Symptoms and Actions

Symptom	Action
ABEND 80A	If you are running the storage analysis function with MATCH and LENGTH options, try one or more of the following: • Increase storage on the job. • Reduce storage pools to one storage type (GBLK, REQS, or VTAL). • Match only a few pools rather than all GBLK or all VTAL or all REQS pools. • Run the LENGTH option without the MATCH option, and save the output for future reference. The LENGTH option is independent of the MATCH option, so the same LENGTH output is shown, regardless of whether the MATCH option is specified. • Remove MATCH and LENGTH options. • Specify a start and stop time to limit the amount of data being processed.
Message CANNOT READ FILE WITH DD NAME TRACE received	When processing multiple tapes using the VIT analysis tools, you receive the message CANNOT READ FILE WITH DD NAME TRACE and the return code is 10. If you are attempting to process multiple standard label (SL) tapes using the bypass label process (BLP), verify that the LABEL parameter on the TRACE DD statement is coded correctly. See Table 80 on page 1165 to determine what book describes job control language (JCL).

In addition to the actions suggested in Table 5 on page 43, try the following actions to help you diagnose the problem:

- Use the DEBUG option (add it as a keyword in the parameter data set), which produces large quantities of data showing what the tool is doing. Run this on a small portion of the trace to prevent the output from being too large to be useful.
- Run another tool, such as the MVS IPCS GTFTRACE subcommand, VM TRACERED, or ACF/TAP to see whether they work on this trace data set and to compare output such as time stamps.
- Run a short job to see what is on the tape. A simple way to run a short job is to run VIT extraction with an expression that is never true. Use the NOFORMAT option to avoid the overhead of loading the format routine. You can use the following parameter data set as a short job:

```
Desc Tell what's in this trace data set by running without extracting
Desc any entries.
NOWRAP NOFORMAT
VITEXT e'zzzz'
```

  This data set extracts all VIT entries with the name ZZZZ. (Presumably, no entries start with ZZZZ.) The SUMMARY data set shows whether the data set wrapped, which types of records are traced, which VIT entries and options are traced, and the first and last time stamps.

Documenting an APAR for VIT Analysis Tool Problems

If an APAR is submitted, the following information is required:

- Input data
  - JCL or REXX EXEC or CLIST
  - PARM data set
  - TRACE data set
  - VIT analysis tool load module (ISTRAFT1)

  - Format load module (AMDUSRFD) MVS

- Output data
  - JCL log
  - Data sets produced by the VIT analysis tool
    - SUMMARY
    - LOG
    - DETAILS
    - OUTSTAN
    - VITEXT

VTAM Dump Analysis Tool Problems (MVS)

This section describes how to diagnose problems that might occur while running the VTAM dump analysis tools and includes the following sections:

- "Checklist for Isolating the Problem"
- "Documenting an APAR for Dump Analysis Tool Problems."

If you are having problems during installation of the tool, refer to "Installing the VTAM Dump Analysis Enhancements and the VIT Analysis Tool" in the *VTAM Network Implementation Guide* for additional information.

Checklist for Isolating the Problem

1. Determine whether the error occurred as a result of an ISPF panel or a module.

   ISPF handles its own error conditions and displays them directly on the panel. If the error message appears in your IPCS output, then it is probably issued from a formatted dump module.

2. If you submit your job using JCL, verify that no JCL errors are issued. If a bad return code is issued, determine whether it is a result of the JCL job or a formatted dump module.

3. If you receive the message `Storage access failed for xxxxxxxx`, browse the dump to determine whether the actual location exists in the dump. Storage requests are usually in terms of the length of the control block. For example, if control block BB is X'20' bytes long, the storage service will be trying to retrieve X'20' bytes of data.

4. View the output to determine whether any error messages were issued during execution. Messages may indicate the cause of the termination.

   If the VTAM formatted dump routine cannot access a field (either in the control block or in the chain of pointers to the control block), an abend will occur and a note of the condition is made on the dump output.

5. Check whether the required ISPF and IPCS maintenance has been applied as documented in the program directory. Check whether maintenance has been applied to IPCS, ISPF, or VTAM. If any of these are down-level, unpredictable results might occur.

6. If your ISPF prompt lists or PF keys are not working properly, refer to "Key Data Sets Containing Information for VTAM" in the *VTAM Network Implementation Guide* to ensure that everything is installed and concatenated properly.

Documenting an APAR for Dump Analysis Tool Problems

If an APAR is submitted for the problem, the following information is required:

- Dump used when error occurred
- IPCSPRNT output dataset

  IPCSPRNT is the output dataset allocated by you to store all data generated during an IPCS session. See Table 80 on page 1165 to determine what book describes IPCSPRNT.

- JCL if submitted through batch

- Maintenance levels of the following:
  - FFST
  - IPCS
  - ISPF
  - VTAM
  - TSO/E REXX.

  Knowing the level of IPCS, ISPF, and VTAM can help determine whether you are running back-level on these products.

Recommended Documentation for VTAM Problems

Symptoms are often related to a particular device, command, or update to the system. If you suspect this is so, you should tell the IBM Support Center of this relationship. The following sections describe some possible relationships and the kind of documentation you should have for each one.

APAR or PTF Number

If the problem appears after you apply an authorized programming analysis report (APAR) fix, supply the APAR number. If the fix is a PTF (program temporary fix), supply the PTF number. The following is the format for APAR numbers and PTF numbers.

Operating System	APAR	PTF
MVS/ESA	OWnnnnn OYnnnnn	UWnnnnn UYnnnnn
VM	VMnnnnn	UVnnnnn
VSE	DYnnnnn	UDnnnnn

Device Type

If the problem is associated with the use of a particular type of terminal or other hardware unit, supply that device type (such as 3278 Model 2). If the problem is associated with a particular type of communication link, supply appropriate link characteristics, such as SDLC, BSC, SNA, or non-SNA. Also, identify any recent microcode activity on the control units involved.

Operator Command

If the problem is associated with a particular VTAM operator command, supply the full command name (such as VARY). Also, note any command operand (such as INACT) or a network node type (such as CDRM) that has been associated with the problem.

Terminal Action

If the problem is associated with a particular terminal action, such as IBMECHO, USS LOGON, or pressing the CLEAR key, describe the action (or sequence of actions).

VTAM Application Program

If the problem is associated with a VTAM application program that is an IBM licensed program (such as CICS or TSO), supply the name of the licensed program.

Hardware Error Condition

Sometimes it is immediately apparent that a problem is related to a specific hardware error condition. The hardware error might have been detected and reported in several ways:

- By an operating system message
- By a VTAM or application program message
- By the system operator
- By a VTAM buffer filling up with information from one device
- MVS VM Through LOGREC
- VSE Through SYSREC or RMS

- By a terminal user (an indicator of the error status appears in the operator information area, at the bottom of the terminal screen).

If a hardware error occurred, note the failure condition that accompanied it, such as UNIT CHECK or TIMEOUT.

If you think your problem is related to a hardware failure, use the following tools to collect information about the hardware failure:

- SDLC link level 2 (LL2) test. For more information on the LL2 test, see "Modify SDLC Link Level 2 Test" on page 211.
- NCP intensive mode error recording. For more information on intensive mode recording, see "Modify NCP Intensive Mode Recording" on page 210. See Table 80 on page 1165 to determine what NCP book describes intensive mode recording.
- The NetView program, if you use it in your system
- LOGREC (or similar operating system facilities).

Note: For help with hardware problems, use the NetView program if you have it installed, or use the system console messages to identify the affected part of the network. If you need further assistance, contact your IBM branch office.

Coding Change

A problem can occur after you make coding changes to the following:

- VTAM network definitions
- Macro usage
- Start options
- User-coded exit routines
- MVS VSE Job control statements
- User applications.

Supply information about the coding change. For example, if you change the PACING operand on an NCP LU definition statement, supply that information.

Use Table 6 to determine the type of documentation you need to either solve your problem or supply to the IBM Support Center.

Note: Documentation for the NetView program is included in Table 6.

Table 6 (Page 1 of 3). Recommended Documentation for VTAM Problems

Documentation	Description
MVS VM Alias names	If your configuration is using SNA network interconnection and you are using alias names, keep a list of the alias names defined to each name translation program.
Application program log (if appropriate)	Some user-written operator application programs produce an application program log.
VM DCSS map	The installation process produces a map of the VTAM Shared Segment (DCSS). This map has the filename of your VTAMDCSS and a filetype of MAP. It is located on the VTAM BUILD0 PRODUCTION disk (default X'29A').
Exit routines	Keep a list of VTAM exit routines.
Link-edit map	MVS If a VTAM load module is involved in a problem, an XREF map of the load module is needed to show the location of other VTAM modules within that load module. To get an XREF map, use the service aid LIST (AMBLIST) with the control statement LISTLOAD and the parameter OUTPUT=XREF. This produces a listing showing the module (CSECT) names and their location within the load module. See Table 80 on page 1165 to determine what book describes how to use the LIST service aid. VM Save the VTAM LKEDIT and VSCS LKEDIT files that are produced by the installation procedure for VTAM and VSCS. These files are on the VTAM BUILD0 PRODUCTION disk (default X'29A'). VSE Most VTAM modules are link-edited into multiple CSECT phases. In some cases, it might be necessary to determine where within the phase a failing module is loaded. To do this you must have the link-edit map for the phase. Save the output from the following: • Initial installation steps for VTAM • Maintain System History Program (MSHP) linkage editor. The MSHP linkage editor is used when maintenance is applied to the system.
MVS Link Pack Area (LPA) map	Contains names and starting addresses of modules in SYS1.LPALIB. To get an LPA map, use the IBM service aid LIST (AMBLIST) with the control statement LISTLPA. See Table 80 on page 1165 to determine what book describes the LIST service aid. When it is used with a link-edit map and a dump, an LPA map enables you to identify a module that is found at a specific address within the link pack area.
MVS VM LOGREC	This is a MVS data set or VM file that contains records of various types of system failures, both hardware and software. For hardware failures, LOGREC entries contain sense and status information about the device causing the failure. For software failures, LOGREC entries contain information such as the program status word (PSW), the abend code, the failing module name (when possible), a symptom string, and the general registers at the time of failure. LOGREC entries are written each time VTAM produces a supervisor call (SVC) dump.
VSE MAP command output	Contains the starting and ending addresses of each partition and SVA.
NetView hardcopy log (if using the NetView program)	Contains messages routed to the NetView program that are associated with an operator terminal.

Table 6 (Page 2 of 3). Recommended Documentation for VTAM Problems

Documentation	Description
NetView file (if using the NetView program)	Contains session awareness data for all active sessions and session trace data for sessions with a resource for which a session monitor trace has been started. Session awareness data includes: • Session type • Names of session partners • Session activation status • IDs of subarea physical units contained in the explicit route assigned to the session • Transmission group numbers • Addresses and network IDs of SSCPs that own links in the transmission groups. Session trace data includes: • Session activation parameters • VTAM PIU data • NCP data.
Network configuration	List any application programs, new devices, or new levels of the operating system you have added to your network. Save the System Modification Program (SMP) configuration data set (CDS) for VTAM and TSO/VTAM components. See Table 80 on page 1165 to determine what book describes SMP.
Program Update Tape (PUT) and Program Temporary Fix (PTF)	Supply a list of any PUTs and PTFs that have been applied to your system. Also, supply a list of changes that have been applied to the hardware, such as requests for engineering activity (REAs) and engineering changes (ECs). If you have identified a module as the source of the problem, supply the PTF eye-catcher if the module has one. (The PTF eye-catcher is the latest PTF number that has been applied to a module. It follows the module ID in a dump.)
Routing data	Keep a table of destination subareas, explicit route numbers, virtual route numbers, paths, and transmission groups as well as a table associating session types, class-of-service (COS) names, and COS tables.
Symptom string	Some VTAM routines provide a symptom string after a failure. After an abend, you will receive message IST931I, which contains the symptom string text. See *VTAM Messages and Codes* for a description of message IST931I. MVS VM The symptom string is put in the system diagnostic work area (SDWA), which is printed by the Environmental Recording, Editing, and Printing (EREP) program as part of the LOGREC entries. See Table 80 on page 1165 to determine what book contains more information on LOGREC. If a first failure support technology (FFST) probe produced the symptom string, EPW messages will appear in the console listing to describe the symptom string. See Table 80 on page 1165 to determine what book contains more information on FFST messages. VSE The symptom string is put in the first records at the beginning of the SVC dump, usually on the first page.
VSE SYSREC	The VSE recorder file (SYSREC) contains error records for devices owned by VTAM. SYSREC records are printed with the EREP program.

Table 6 (Page 3 of 3). Recommended Documentation for VTAM Problems

Documentation	Description
System-console hardcopy Log	Shows all messages sent to or commands received from the operator. May help indicate when the system began to have problems. (VTAM problems may not be apparent at the time they occur.) If your installation has written its own version of a VTAM message, supply the original VTAM message when you report the problem.
Tables	Keep a list of the VTAM tables your installation has defined, such as USS and logmode.
Version/Release number Component ID	**V4R2** **Component** **ID** VTAM MVS 5695-11701-201 VTAM VM 5654-01001-420 VTAM VSE 5686-06501-FE6 At VTAM startup when VTAM initialization is completed, messages IST020I and IST1349I are issued with this information. Message IST020I displays the version and release number, and message IST1349I displays the component ID. In addition, information about the release level of each component is contained in an access-method-support vector list pointed to by the access method control block (ACB). See "ACB-Create an Access Method Control Block" in *VTAM Programming* for more information about the ACB.
VTAM definition library	This is a set of definition statements for resources in the VTAM network, such as the application programs and network nodes. The VTAM definition library also contains the start options used to initialize VTAM, unless they were entered by the system operator. Include configuration lists and user installation exits with the definition library. Detailed information about the VTAM definition library is in the "Key Data Sets Containing Information for VTAM" in the *VTAM Network Implementation Guide.*

Submitting Problem Documentation

If after talking to the Level 2 Support Center representative about your problem, it is decided an APAR should be submitted to the VTAM change team, it may be more convenient for you or the VTAM change team to have the documentation submitted in machine-readable form (that is, on tape). Machine-readable documentation can be handled most efficiently by the change team if you follow these guidelines when creating the tape.

Tapes that are submitted to the VTAM change team may be standard label (SL) or non-label (NL). Cartridge (3480) improved data capability (feature on 3480, standard 3490) (IDRC) may be used. Each tape should contain an external label to identify the tape and its contents in some way. The APAR number should appear on the label. If you use multiple tapes or multiple files on one tape include a separate explanation itemizing the contents of each tape.

With each tape, include the output from the job used to create the tape. To verify that the tape was created correctly and that the job completed normally, the change team must have the output from the job that created the tape (not simply the job control statements that were used).

Submitting Documentation (MVS)

Note: The information in this section is from APAR OY17061. See that APAR for additional information.

To submit dumps, traces, and other information to the MVS change team, do the following:

For dumps

You should not dump data in any way prior to or during the transfer of the dump to tape. MVS/ESA dumps can be transferred to tape using IPCS or IEBGENER. See Table 80 on page 1165 to determine what book describes how to use the IPCS and IEBGENER utilities.

You should not change the data control block (DCB) parameters of the dump data set. The DCB parameters should be defined for MVS/ESA as follows:

```
LRECL=4160, BLKSIZE=4160, RECFM=F
```

For GTF traces

Move the GTF trace data from the trace data set (which is usually SYS1.TRACE) to tape using IEBGENER only. The DCB parameters for a GTF trace should be one of the following:

```
LRECL=4092, BLKSIZE=4096, RECFM=VBA
LRECL=4092, BLKSIZE=32760, RECFM=VBA
```

For both traces and dumps, do not reblock the data (that is, use a different BLKSIZE) when moving it to tape. Use only the DCB parameters shown in the preceding example.

Restriction	Use of any other utility (IBM or non-IBM) to transfer dump or trace data to tape may result in a processing delay and could result in the APAR being returned to you (closed "RET") due to the inability of the change team to process the tape.

For Other Types of Information

Other types of information (for example, VTAM definitions, NCP stage one input, and console logs) may be submitted on paper or tape. If you submit the data on tape, it should be written to tape using IEBGENER only. The DCB parameters used when writing this type of data to tape should be the same as the input data set (that is, the same DCB parameters as the source of the data).

Submitting Documentation (VM)

To submit dumps, traces, and other information to the VM change team, do the following:

For Dumps and Traces:

- Make sure you have accessed the DVF and TRACERED disks.
- Process the dump using DUMPLOAD.

  Process the trace using TRACERED.
- Receive the trace to a disk.
- Use VMFPLC2 to create the tape. See Table 80 on page 1165 to determine what book describes VMFPLC2 for your operating system.
- Include on the tape all the files produced by the processing of the dump (for example, DUMP, REPORT) or the trace and other files that have been requested by Level 2.

  If you have a problem processing either the dump or trace, you should submit unformatted system dump files to the VTAM change team. Both the dump and the trace are initially in this VM system dump type format (`VM CP QUERY RDR` shows class `DMP`, `QUERY TRFILES` shows trace files). The only way to put these files on tape is to use the CP SPTAPE command. See Table 80 on page 1165 to determine what book describes how to use SPTAPE on your operating system. If the dump or trace can be processed satisfactorily, do not use this format because it makes the tape more difficult to process for the change team.

For Other Types of Information: Following are examples of other files you may need to submit:

- VTAM and system console logs
- Nucleus, load, and LKEDIT maps
- VTAMLST files (for example, major node definitions, start lists)
- DTIUSERx assemble files
- VTAM or VSCS LOADLIBs.

After dumping all the files to the tape, write 100 tape marks at the end as follows:

```
VMFPLC2 WTM 100
```

Adding the tape marks ensures that the tape does not spin off the end.

If you need to submit both system dump files and CMS files, you can put SPTAPE and VMFPLC2 output on the same tape. You can use the class B CP command `DETACH raddr FROM userid LEAVE` to leave the tape positioned for SPTAPE after using VMFPLC2, or you can use `VMFPLC2 FSF x` to forward space "x" tape marks to

bypass the SPTAPE files. Include a cover letter with the tape indicating what commands were used to process the tape and what files are on the tape.

Submitting Documentation (VSE)

Note: The information in this section is from information APAR II07725. See that APAR for additional information.

Dump Output

Attention Routine (AR) Dump: To send an attention routine dump to tape, issue

```
DUMP Fx,cuu
```

where cuu is the address of the tape drive.

SYSDUMP: Using INFO/ANALYSIS:

- Select DUMP OFFLOAD to put a dump file residing in the dump library onto tape.
- Select DUMP ONLOAD to copy a dump file from tape into the dump library.

Stand-alone Dump: The output tape from the stand-alone dump can be submitted as is.

Power Queue Output

Use Power Offload (POFFLOAD) to put Power queue entries onto tape.

BACKUP or SAVE	Puts queue entries onto tape.
LOAD	Retrieves queue contents from tape.
SELECT	Retrieves individual queue entries.

TPRINT Output: Use Power Offload (POFFLOAD) to put TPRINT output onto tape.

DOSVSDMP Output: Use Power Offload (POFFLOAD) to put DOSVSDMP output onto tape.

Chapter 2. Collecting Documentation for Specific Types of Problems

After you have classified your problem as a specific type using information in Chapter 1, "Diagnosing VTAM Problems: Where to Begin" on page 3, this chapter shows you how to collect the additional information you need before contacting the IBM Support Center.

This chapter includes the following sections:

- "Common Problem Determination Procedures" describes procedures for specific problem types.
- "Failing Module" on page 109 tells what to do when you have isolated the problem to a specific module of the VTAM program. You might be sent to this section from within the procedure for the problem type you have chosen.
- "Symptom String Structure" on page 111 describes the meaning of the fields found in a symptom string.
- "Reporting the Problem to IBM" on page 112 describes how to report the problem to your local branch office or the IBM Support Center.

Common Problem Determination Procedures

This section includes a description of the following procedures:

- "Abnormal End (Abend) (MVS) (VM)" on page 58
- "Program Check (VSE)" on page 63
- "Wait" on page 67
  - "VTAM Locks" on page 80
  - "Using the VARY INACT,FORCE Command" on page 83
- "Loop" on page 86
- "Message Problem" on page 93
- "Incorrect Output" on page 96
- "Performance Problem" on page 101
- "Storage Problem Procedure (MVS)" on page 105
- "Documentation Problem" on page 108.

Abnormal End (Abend) (MVS) (VM)

If the problem is an abend, use the procedure in Figure 2 to collect the following documentation:

- Dump output
- LOGREC
- Symptom string
- Abend or system completion code
- Contents of the general registers (at the time of the abend)
- Module ID and PTF eye-catcher
- PSW (at the time of the abend)
- VIT at the time of the abend.

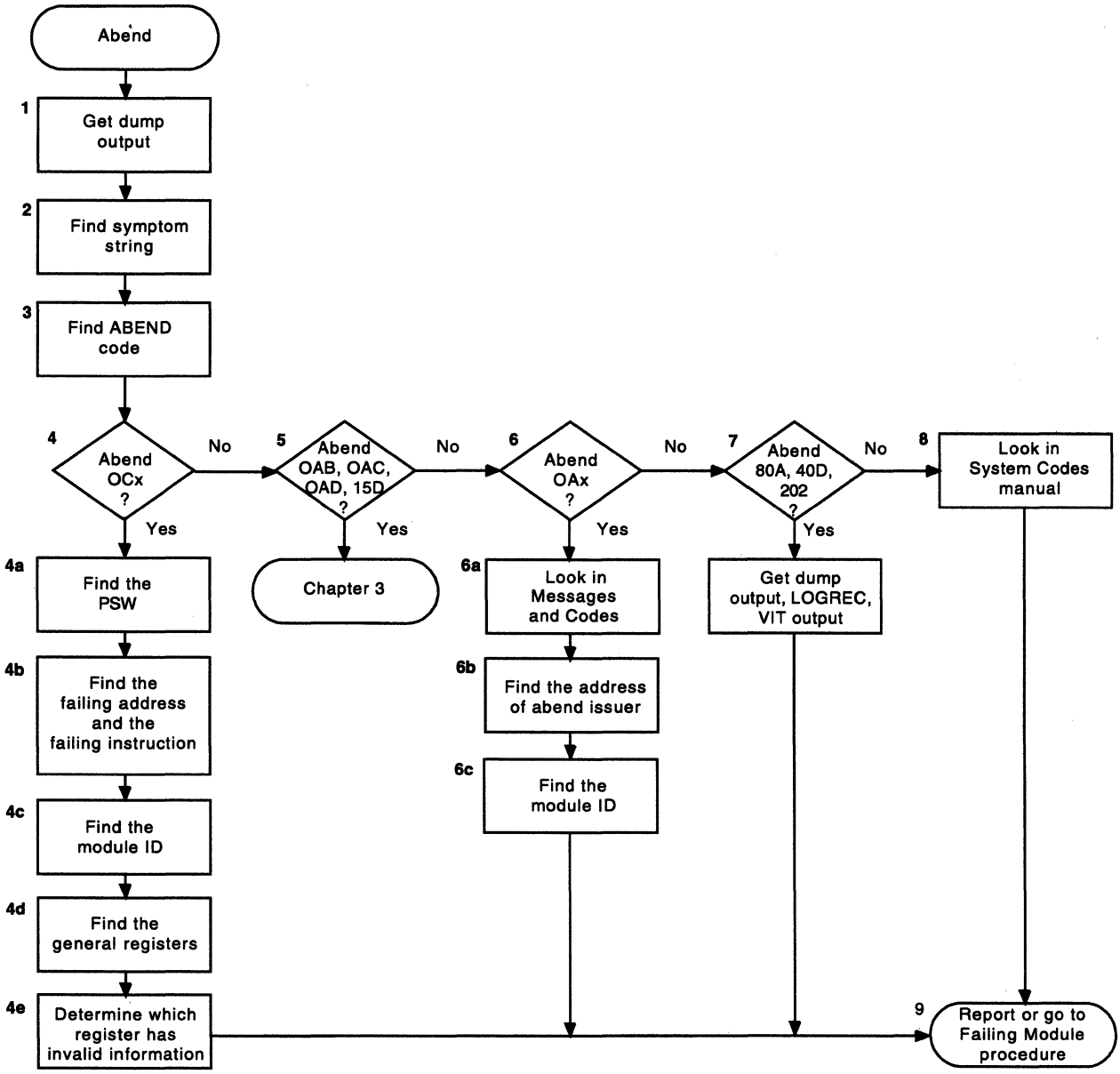


Figure 2. Overview of the Abend Procedure MVS VM

Following is a description of each step shown in Figure 2 on page 58.

1. **Get dump output.**

   VTAM usually produces a dump for an abend. If no dump was taken, the dump files or spools might be full. Check for a message that an error occurred while VTAM was trying to produce the dump. If VTAM is not able to complete the dump, you have to re-create the abend or wait for it to occur again.

   Note: To extract abend information from a VTAM dump, invoke the CLIST "ISTVABND" on page 280 or continue with the following steps.

2. **Find the symptom string.**

   A symptom string is a structured database search argument. The symptom string gives information about what was happening at the time of the abend. Message IST931I, which contains the symptom string text, is issued when a symptom string is produced. A record is written to the LOGREC data set whenever VTAM takes a supervisor call (SVC) dump. For about 85% of all abends, recovery routines produce a symptom string subset, which you can get by printing LOGREC. The symptom string subset, if it occurs, is located in control block SDWA in LOGREC.

   MVS The SDWA address should be listed in the beginning of the dump, in the dump abstract information. It is also printed out in LOGREC, labeled "Hex Dump of Record" at the end of each software entry.

   The symptom string begins at X'194' in the SDWA. Field SDWAURAL gives the length of the symptom string, which can be up to 256 bytes.

   VM There are three ways to find the symptom string:

   a. If VIT is active, locate the ABND SNAP trace record. This shows the address of the SDWA.

   b. If VIT is not active, scan the registers shown at the beginning of a dump for the address of the SDWA.

      The SDWA can be difficult to recognize. However, the abend code is at + X'4', so knowing the abend code before you start looking for the SDWA can help.

   c. If neither the trace nor the dump is available, scan the software error portion of LOGREC for the symptom string. The symptom string can be found in the comments section to the right of the SDWA.

   See "Symptom String Structure" on page 111 for a description of the fields in a symptom string.

3. **Find the abend, system completion, or user completion code.**

   MVS You can find the abend (or completion) code in the output of several different service aids. The system control block RTM2 work area (RTM2WA), the SYS1.LOGREC software record, and the task control block (field TCBCMPC) contain the completion code. The RTM2WA is pointed to by the TCB of the failing task (field TCBRTWA), and is listed after the abending TCB.

   VM You can find the abend code, ABNCODE, in the abend work area, ABNWA. Low core X'298' points to the ABNWA. You can also find registers 0-F and the PSW in the ABNWA. Registers 0-F are found at offset X'00' of the ABNWA, the PSW is found at offset X'40', and the ABNCODE is found at offset X'48'.

For more information on the abend codes issued by VTAM, see Chapter 11, "Abend Codes (MVS) (VM)" in *VTAM Messages and Codes*.

4. **Determine whether the abend code is OCx.**

   If the completion code is of the form 0C*x* (where *x* = the program interruption code from the PSW), continue with this step. If the abend code is not 0Cx, go to step 5 on page 61.

   a. Find the program status word (PSW) at the time of the abend.

      The PSW is found in:

      - MVS The LOGREC output, the SDWA, or the RTM2WA
      - VM The ABNWA, the SDWA, or the LOGREC output.

      The location of the PSW in the dump output varies depending on the type of dump taken. For assistance in locating the PSW in dump output, see the diagnostic books for your operating system.

   b. Find the failing address or the failing instruction.

      The PSW contains either the address of the next instruction to be executed at the time of the abend or the instruction that failed at the time of the abend, depending on the interruption code.

      If the interruption code is X'10' or X'11', then the PSW address points to the failing instruction. Otherwise, back up the PSW by the instruction length, and *that* is the failing instruction. Scan the dump output to find the address given in the PSW.

      Note: VM If VTAM is using an extended-mode PSW, you must determine if the PSW address is 24 or 31 bits in length. If the PSW is 24 bits in length, ignore the high-order byte.

      If you cannot find the address, the dump might not contain the relevant portion of main storage.

   c. Find the module ID for the module that contains the failing address.

      VTAM identifies modules with the module name, Julian date, and PTF or APAR eye-catcher at or near the beginning of each module. This module identifier is in the form:

      ```
      ISTxxxxx yy.ddd nnnnnnn
      ```

      where *xxxxx* is the last five characters of the module name, *yy.ddd* is the Julian date the module was assembled, and *nnnnnnn* is the latest PTF or APAR fix (if any) that has been applied to this module.

      VM VSCS identifies modules with the module name, Julian date, and PTF or APAR eye-catcher at or near the beginning of each module. This module identifier is in the form:

      ```
      DTIxxxxx yy.ddd nnnnnnn
      ```

      where *xxxxx* is the last five characters of the module name, *yy.ddd* is the Julian date the module was assembled, and *nnnnnnn* is the latest PTF or APAR fix (if any) that has been applied to this module.

      Sometimes VTAM puts the module name of the failing module in LOGREC. If it is not there, you can find it in a dump. To find the module ID in a

dump, start at the failing address and scan in descending address order along the right side of the listing. The module ID is printed in EBCDIC.

MVS You can also scan the LPA map for the name of the load module and then go to the AMB list in the load module to find the CSECT that contains the failing address.

VM If the failure is in a shared VTAM module, you can scan the VTAM DCSS MAP file for the name of the module that contains the failing address.

d. Find the general registers.

- MVS The LOGREC output, the SDWA, or the RTM2WA
- VM The ABNWA, the SDWA, or the LOGREC output.

Use the diagnostic books for your operating system to help find the registers.

e. Determine which register has information that is not valid.

The failing instruction often uses a register with an address that is not valid in one of the general registers, or points to an location that is not valid (for example, low-address storage). Use *Principles of Operation* for your operating system, the program interruption code from the MVS SDWA or from VM ABNWA and the general registers used in the failing instruction, to determine (if possible) which register contains or points to incorrect data.

Note: When determining the validity of the register's contents, be careful to consider the address mode used by your operating system. Depending on the address mode being used, values used in 31-bit addressing might be interpreted differently than those used in 24-bit addressing.

Next go to step 9 on page 62.

5. MVS **Determine whether the abend code is 0AB, 0AC, 0AD, or 15D.**

These abend codes indicate a TSO/VTAM abend. For diagnosis information, go to "TSO/VTAM Abends" on page 117.

6. **Determine whether the abend code is OAx.**

If the abend code is in the form 0A*x*, continue with this step. If not, go to step 7 on page 62.

a. Find the abend code explanation in Chapter 11, "Abend Codes (MVS) (VM)" in *VTAM Messages and Codes*.

An abend code of 0A*x* indicates a problem within the VTAM network. The problem could have originated in VTAM, the NCP, an application program, or the hardware of some other network component. Look up the code in Chapter 11, "Abend Codes (MVS) (VM)" *VTAM Messages and Codes*. Most 0A*x* abends place a return or reason code in register 15 at the time of failure. You can find the return code in register 15 by using the set of general registers from:

- MVS The LOGREC output, the SDWA, or the RTM2WA
- VM The ABNWA, the SDWA, or the LOGREC output.

b. Find the address of the module that issued the abend, using the PSW, which points to the next instruction after SVC 13.

c. Find the module ID.

From the address determined in the previous step, scan in descending address order through the dump to find the module ID (see step 4c on page 60).

Go to step 9.

7. **Determine whether the abend code is 80A, MVS 40D, or 202**

If the abend code is one of these, continue with this step. Otherwise, continue with step 8.

These abend codes indicate storage problems. Collect the following documentation:

- A dump of the VTAM address space
- A dump of the VTAM common storage area (CSA)
- LOGREC output
- VIT output at the time of the abend.

After obtaining this documentation, go to "Reporting the Problem to IBM" on page 112.

8. **If the abend code is none of the above, refer to your operating system documentation.**

To determine the publication that describes the abend codes for your operating system, see Table 80 on page 1165.

Each code has an explanation of the documentation required and the problem determination steps to follow. For example, many abends occur during execution of SVC instructions. Parameter lists and register contents passed to SVC routines are in the diagnostic books for your operating system. These books might suggest that you obtain additional information such as a module name, a return code, a register containing information that is not valid, or the name of a system control block containing parameters that are not valid. After making a complete check of these sources, you are ready to report the problem.

9. **Report or go to the failing module procedure.**

If you determined the module ID, go to "Failing Module" on page 109. Otherwise, see "Reporting the Problem to IBM" on page 112.

Program Check (VSE)

If the problem is a program check, use the procedure in Figure 3 to collect the following documentation:

- Dump of VTAM partition, supervisor, and data spaces.
- SYSREC
- Symptom string
- Message information
- Contents of the general registers (at the time of the error)
- Module ID and PTF eye-catcher
- PSW (at the time of the error).

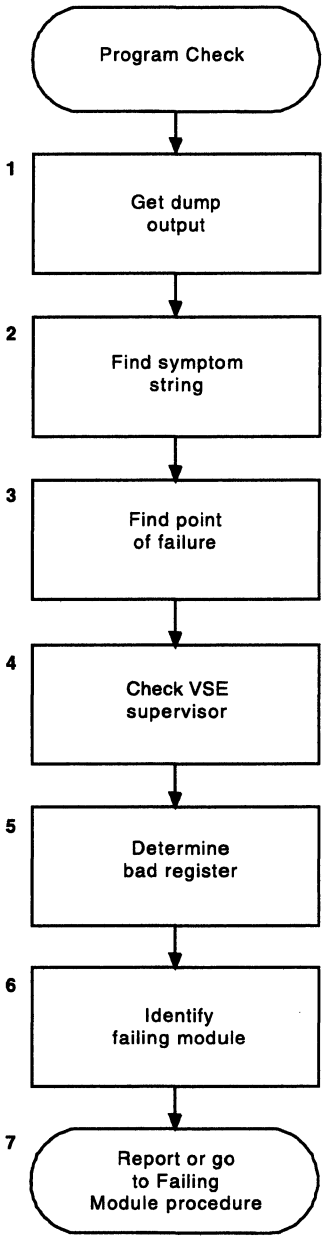


Figure 3. Overview of the Program Check Procedure VSE

Following is a description of each step shown in Figure 3.

1. **Get dump output.**

   When a VTAM abend occurs, two back-to-back dumps (IDUMPs) and a dump for each open data space are taken. The first dump contains the VTAM partition and VSE supervisor storage, including used portions of the system GETVIS area. The second dump contains the shared virtual area (SVA). Both dumps are necessary because the SVA contains initialization and user-task modules that are not automatically dumped by the supervisor. The dumps contain the same symptom string data.

   The VTAM job should specify:

   ```
   // OPTION DUMP,SYSDUMP
   ```

   OPTION DUMP provides the system GETVIS area and other supervisor areas as part of the VTAM abend partition dump. Do not specify PARTDUMP; OPTION PARTDUMP does not provide all of the VSE supervisor areas used in performing VTAM functions.

   In the hard wait cases, use the formatting option to print the VSE supervisor areas and the VTAM partition. If VTAM resides in a private partition, print the SVA. If VTAM does not reside in a private partition, it is not necessary to print the SVA since VTAM modules are not executed in the SVA.

   See Table 80 on page 1165 to determine what book describes how to obtain and format VSE dumps.

2. **Find the symptom string.**

   A symptom string is a structured database search argument. The symptom string gives information about what is happening at the time of the program check. A record is written to the SYSREC file whenever VTAM takes an SVC dump. For about 85% of all program checks, recovery routines produce a symptom string subset, which you can get by printing SYSREC.

   Recovery routines produce the symptom string if storage is available for it. The symptom string subset, if it occurs, is located in the first records of the SVC dump output, usually on the first page. Figure 17 on page 111 shows an example of a symptom string subset in dump output.

3. **Locate the point of failure.**

   - If the program check is reported by VSE message 0S03I, the exception type and location are contained in the message text.
   - For hard wait states (X'FFF' and X'FFB'), the failing instruction address can be determined from the program check old PSW. See Table 80 on page 1165 to determine what book describes the contents of the DOSVSDMP dump.
   - For task termination messages other than VSE message 0S03I, find the save area containing the error PSW and registers (in the beginning of the task termination dump).
   - For message IST252I with return code 36, or VSE message 0V08I, the canceled task is VTAMRP, but the save area used is not the normal subtask save area. Instead, the VSE program check handler stores the PSW and registers at label SVPCSAVE.
   - For message IST252I with return code 20, the VSE dump of VTAM is taken under the main task. Locate the VTAMRP save area either by finding its

task control block in the VSE supervisor dump or by using location X'408' to find the ATCVT and field ATCFLDA to locate module ISTINCF0. ISTINCF0 contains the save area.

Identify the save area by the name VTAMRP in the first 8 bytes. VTAMRP begins the save area of the routine that generated the dump. The PSW and registers of the failing task start at offset X'90' from the beginning of the save area.

Note: The registers of the failing task begin with register 9 and end with register 8, as follows:

R9, R10, ... R15, R0, R1, ... R8

- For wait state X'FED', use the address in register 5 as the failure address.

4. **For WAITFFF and WAITFFB, determine if the failure is in the VSE supervisor.**
   - Check the VSE supervisor listing, and include the label preceding the point of error in your information for the Support Center. If the logic pertains to a VTAM function, the problem might be in VTAM.
   - If the instructions have been overlaid by data, try to identify that data from its contents, and note where the overlay begins and its length. If the overlay contains terminal data or a recognizable VTAM control block, the problem is probably in VTAM. Save any pertinent information about the overlay to discuss with the IBM Support Center. If the overlay is determined to be a VTAM control block, the name should be given to the Support Center.
   - If the last interruption is a VTAM SVC 49 or SVC 53, add that to the information in the form SVC*xx*.

     If the last interruption is another VSE SVC issued by VTAM, then VTAM might be issuing the SVC incorrectly. Add that information in the form SVC*xx*.

     Proceed to step 5, using the address in the SVC old PSW as the failure address.
   - If the last interruption is an I/O interruption from a VTAM-owned channel-attached device, the problem might involve one of the VTAM appendage routines.
   - For the FFB condition, try to determine what is being referred to that caused the page fault. If the exception address at location X'90' points to an area in the VTAM partition, the problem might be in VTAM. If the address is within a VTAM module (appears to be executable code), identify the module as in step 6 on page 66. If not, try to identify it as a VTAM control block, using the list of control block IDs in Appendix H, "Storage and Control Block ID Codes" on page 1137. If you determine the name of the control block, add the name to your list of documentation. If not, the area should be either in a VTAM module (see step 6 on page 66) or in a VTAM buffer pool (see Appendix B, "Finding VTAM Buffer Pools (VM, VSE)").
5. **Determine which register contains or points to incorrect data.**

The failing instruction normally has an address that is not valid in one of the general registers or points to a location that is not valid (for example, low storage). Use *Principles of Operation* for your processor, the program interruption code from the PSW, and general registers in the failing instruction to determine (if possible) which register contains or points to incorrect data. Add this register number string in the form REG*nn* to the information for the Support Center.

6. **If the failing instruction address found in step 3 on page 64 is in the VTAM partition, identify the failing VTAM module.**

   Most VTAM modules are identified by a field containing the name IST*xxxxx* and the current level of the module near the beginning of the module (for example, ISTTSCBY 93.299 UD48890). To locate this identifier, scan the dump in descending address order from the point of failure. The identifier is printed in EBCDIC along the right-hand side of the dump. Add the module name to your documentation list. Go to "Failing Module" on page 109 to continue obtaining information for the Support Center.

7. **Report or go to the failing module procedure.**

   If you determined the module ID, go to "Failing Module" on page 109. Otherwise, go to "Reporting the Problem to IBM" on page 112 for information on contacting IBM.

Wait

If the problem is a wait, use the procedure in Figure 4 on page 68 to collect the following documentation:

- I/O trace output
- Buffer contents trace output
- Session trace data (if using the NetView program)
- Session awareness data (if using the NetView program)
- MVS Dump of the VTAM primary address space including CSA
- VM Dump of the VTAM virtual machine including DCSS. (DCSS is called DSS in VM commands.)
- VSE Dump of the VTAM partition and VSE supervisor.
- List of:
  - Waiting process anchor blocks (PABs)
  - Waiting request elements (WREs) and associated event IDs (EIDs)
  - Waiting request parameter headers (RPHs).
- For problems associated with an application program:
  - Output from the VIT (all options except LOCK)
  - RPLs or FMCBs queued to the ACDEB.
- For problems associated with the network:
  - Trace output
    - Line trace
    - Generalized PIU trace (GPT)
    - Transmission group (TG) trace
    - Scanner interface trace (SIT) for IBM 3720, 3725, and 3745 controllers.
  - VM VSE Dynamic trace of the integrated communication adapter (ICA) (if available)
  - Dump output
    - NCP dump
    - MOSS dump
    - CSP dump
  - Reports from NetView, IMR, or EREP (if available).

Note: Use the documentation you have available to isolate or resolve the problem. If you have to re-create the problem, make sure the traces listed above are active.

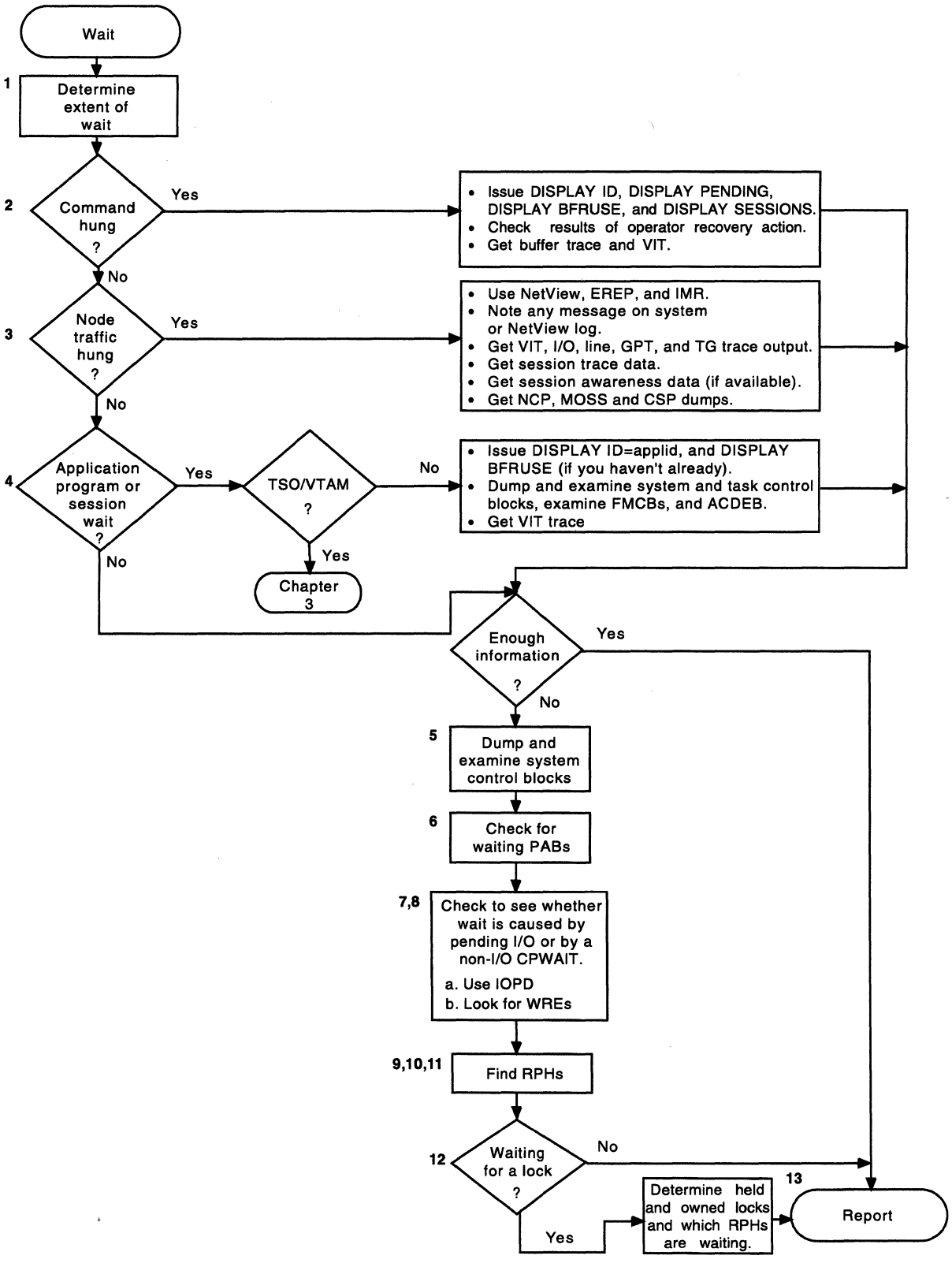


Figure 4. Overview of the Wait Procedure

Following is a description of each step shown in Figure 4 on page 68.

1. **Determine the extent of the wait condition.**

   Determine how extensive the wait state is in the operation of the VTAM network. Determine whether all VTAM processing stopped or only processing with respect to a single device, application, or something in between. Also determine what, if any, recovery action was taken at the time the wait was encountered by the operator or user. Some information about the activity that immediately preceded the wait might be available on the system log or in application program transaction logs.

2. **Did a log on, log off, or command fail to complete?**

   If so, continue with this step; otherwise, go to step 3 on page 70.

   - If the wait condition was actually the failure of a VTAM procedure to complete, use the DISPLAY ID command to identify the status of VTAM resources at the time of the problem. Note any status codes that are abnormal. (Resource status codes are explained in "Resource Status Codes and Modifiers" in *VTAM Messages and Codes*.)
   - Use the VTAM DISPLAY PENDING, DISPLAY SESSIONS, or MODIFY IOPD commands to identify I/O requests for which VTAM is awaiting a response from a network node. Sometimes a network node appears in a pending state awaiting the completion of activity at a higher- or lower-level node (for example, PSUB1, PTRM2). The pending status on the other node is needed in such a case. See "Resource Status Codes and Modifiers" in *VTAM Messages and Codes* for information on resource status codes.
   - Use the VTAM DISPLAY BFRUSE command to get information about VTAM buffer pools. Save the output for use later in this procedure.
   - A VTAM operator might have attempted a recovery action (such as issuing a VARY INACT,FORCE command). "Using the VARY INACT,FORCE Command" on page 83 shows how to determine whether this command completed. Check the node status to determine if the recovery action reset the state of the node for which the original command was issued.
   - If VTAM is waiting for an I/O response, look at the output of the VTAM buffer contents trace (assuming it is active when the problem occurs). If the trace shows that VTAM did send a request and is expecting a response, the problem is probably in another network node.
   - You can get additional information on the status of a command from the VTAM internal trace (VIT). With the SSCP and PIU options, you can match requests and responses and determine any requests that are outstanding (that is, for which responses have not been received). The SMS option supplies information about resource usage, and the PSS option provides information about VTAM's scheduling of the dispatching process. (See Chapter 9, "Using the VTAM Internal Trace (VIT)" for a description of the internal trace entries.)

   At this point you might have enough documentation to report the problem to the Support Center. If so, go to "Reporting the Problem to IBM" on page 112. Otherwise, go to step 5 on page 72.

3. **Is network traffic stopped through a specific node?**

   If so, continue with this step. Otherwise, go to step 4.

   - Add the specific node type to your problem documentation. For example, the node could be a 3705, 3720, 3725, 3745, 3790, or a 3274. NetView and EREP facilities show whether errors have been recorded for the node in question. Session trace data (collected by the NetView program) shows whether the node is not responding to VTAM, or whether VTAM is discarding the responses. Consider using NCP intensive mode recording (IMR) for recurrent problems of this type.
   - Note any messages on the system or NetView command facility log reporting ER-INOP outages or other failures. Use the VIT trace, or use the I/O trace with the EVERY operand, to trace the network flow up to the point of failure. NetView, MVS VM LOGREC, and VSE SYSREC show the reason for the INOP.
   - For NCP-related problems, use the line trace or generalized PIU trace if the affected node is in an adjacent subarea. Use the transmission group trace to record intermediate node flows up to the point where the problem occurred.
   - If the problem might be in NCP software or communication controller hardware, obtain a dump of NCP storage. If the wait affects only part of the network, use the dynamic NCP dump facility. It allows the rest of the network to continue operating while the dump is taken. If the failure requires reactivating the NCP, use the MODIFY DUMP command. See "Network Control Program (NCP) Dump" on page 227 for more information on NCP dumps.

     If the NCP is hung or if the hung resource is attached to an NCP, see Table 80 on page 1165 to determine what NCP diagnostic book describes troubleshooting the NCP.
   - VM VSE If the problem is in a device attached to the ICA, use the dynamic trace of the ICA to obtain information about line-related errors. See Table 80 on page 1165 to determine what book describes the dynamic trace.
   - If the problem is in a channel-attached device or a channel-to-channel attachment, examine one of the following traces, if available, to determine the sequence of events preceding the wait. (If no trace output is available, you have to re-create the problem to get it.)
     - VIT trace with the CIO option
     - MVS CCWTRACE
     - VM CP virtual machine trace with I/O, SIO, CCW, and CSW.
     - VSE SDAID program with the SIO and I/O trace types and the output of CCWD. To see all the data that is transferred on the channel, set the value of CCWD equal to IOBUF size.

     To determine what book describes I/O control blocks for your operating system, see Table 80 on page 1165.

   If enough information is available, go to "Reporting the Problem to IBM" on page 112. Otherwise, go to step 5 on page 72.

4. **Is it a session or application program wait?**

If the wait condition appears to be related to a particular VTAM application program, continue with this step. Otherwise, go to step 5 on page 72.

- Enter the DISPLAY ID command for the application program, using the EVERY or SCOPE=ALL operand. If there are any nodes with status ACT/U, re-enter the DISPLAY command. If you are again informed that the status of a node is ACT/U, issue VARY INACT,FORCE for that node. If you still have a wait condition, continue with the next step.
- If only one application program is waiting while others continue to communicate with VTAM, that application program probably contains an error. To determine what caused the problem, obtain a dump of the application program and the operating system supervisor at the time of the problem.
  - Make sure that the error is not an operating system error. (Use the diagnostic books for your operating system.)
  - If possible, use the dump to determine the reason the application program is waiting. If the application program is not waiting for VTAM, use the documentation for the application program to determine the reason for the wait. If the problem is in TSO/VTAM, go to Chapter 3, "Collecting Documentation for TSO/VTAM Problems (MVS)." If the problem is in VSCS, go to Chapter 4, "Collecting Documentation for VSCS Problems (VM)."
- If VTAM still seems to be the cause of the problem, you need output from the VIT to obtain a record of activity on the failing session. Because large amounts of data will wraparound in the internal trace table, you might want to specify MODE=EXT.

  See Chapter 9, "Using the VTAM Internal Trace (VIT)" for more information on using the internal trace. You can also use the I/O or buffer contents traces to get information about all sessions with that application; specify ID=*application program name*.
- Using a dump of the problem, find the address of the VTAM ACDEB for the application program.

  You can find an ACDEB associated with an application by using the MVS VTAMMAP SES formatted dump tool. If VTAMMAP cannot be run, then find the ACDEB chain pointer in the ATCACDA field of the ATCVT.

  a. Use the ACDEB's address to find it in the dump.

     On the FMCB RECEIVE ANY queue, ACDRAFQH points to the first FMCB.

     On the RPL RECEIVE ANY queue, ACDRARQ points to the first RPL.

     Notes:

     1) If there are FMCBs (ACDRAFQH is not equal to 0), but no RPLs (ACDRARQ = 0), a problem has prevented the application program from issuing RECEIVEs.

     2) If there are RPLs (ACDRARQ is not equal to 0), but no FMCBs (ACDRAFQH = 0), there might be a problem involving the continue any/continue specific (CA/CS) state of the session. (See "Continue-Any Mode Versus Continue-Specific Mode" in *VTAM Pro-*

gramming for more information on the continue any/continue specific mode.)

b. Check for blocked PABs in the process scheduling table (PST). ACDTSKID points to the PST.

   Look at the following PABs in the PST. To determine the offset locations for these PABs, see *VTAM Data Areas for MVS/ESA*, *VTAM Data Areas for VM/ESA*, or *VTAM Data Areas for VSE/ESA*.

   PSTRQPAB Request PAB
   PSTRSPAB Response PAB
   PSTUEPAB User exit PAB

   See steps 6 and 9 for additional recommended actions.

c. Get the LUCB address (field ACDLUCBA in the ACDEB).

d. Get the address of a chain of FMCB extensions (field LUCFMCBA in the LUCB). Each FMCB extension represents one LU-LU session.

e. Each FMCB extension contains a pointer (field TSPFMCBA) to the address of an associated FMCB. Find the FMCBs associated with hung sessions.

   In those FMCBs, look for:

   - The CA/CS indicator (in TSPPSFL1 and TSPPSFL2)
   - The data queues (in TSPACCUM, TSPEWAIT, TSPNWAIT, TSPEDATA, TSPNDATA, TSPTSOP, and TSPTSIP)
   - Session state flags (in TSPSESSR, TSPDTSR, TSPCRVSR, and TSPRQRSR).

f. Determine whether there are any indications of unusual conditions. See *VTAM Data Areas for MVS/ESA*, *VTAM Data Areas for VM/ESA*, or *VTAM Data Areas for VSE/ESA*.

g. Make a cross-reference listing of network addresses and node names to correlate the VIT PIU and I/O trace entries with VTAM session control blocks, such as the LUCB and FMCB.

See Table 80 on page 1165 to determine what NCP book contains information on hung sessions.

If enough information is available, go to "Reporting the Problem to IBM" on page 112. Otherwise, go to step 5.

5. **Dump and examine the system data areas.**

   MVS If you have not already done so, obtain a dump of the VTAM address space, CSA, LSQA, and SQA.

   Find and analyze the task control blocks. Use the VTAMMAP PABSCAN dump tool to format the output. See "PABSCAN" on page 295 for information on using PABSCAN. See Table 80 on page 1165 to determine what book contains more information on using dumps and finding and analyzing task control blocks.

   VM If you have not already done so, issue #CP VMDUMP 0- DSS FORMAT GCS, with the appropriate options, to obtain a dump of the VTAM virtual

machine. Send the dump output to an authorized service machine to obtain all of the output.

VSE If you have not already done so, obtain a dump of the VTAM partition, SVA, and VSE supervisor. Find and analyze the task control blocks. See Table 80 on page 1165 to determine what book contains information on locating and analyzing task control blocks.

6. **Check for waiting PABs**.

   Note: You can use the MVS VTAMMAP VTCVTPAB formatted dump tool as an alternative to step 6.

   Look at the following PABs in the ATCVT. To determine the offset locations for these PABS, see *VTAM Data Areas for MVS/ESA*, *VTAM Data Areas for VM/ESA*, or *VTAM Data Areas for VSE/ESA*.

ATCCSPAB	Configuration services PAB
ATCVDPAB	VARY definition DYPAB
ATCPXPAB	Buffer pool expansion DYPAB
ATCPUPAB	Physical unit services DYPAB
ATCPUIOP	Physical unit services I/O DYPAB
ATCLUSRT	Logical unit services router DYPAB
ATCNSPAB	TSC no sessions DYPAB
ATCNOSPB	Network operator services DYPAB
ATCSSPAB	Session serialization PAB
ATCSOPAB	Session outage notification PAB
ATCCNSPB	MVS VM CNS logon PAB
ATCTPMPB	Message DYPAB
ATCTRMPB	Termination subtask DYPAB.

   Check the contents of the PABWEQP (or the PABVERYA for very extended PABs) and PABRPHA fields. The field PABWEQP in each PAB contains the address of a chain of work elements that have not yet been processed by VTAM. The field PABVERYA is defined at the same location as PABWEQA and contains a pointer to an array of WKE queues.

   The array pointed to by the PABVERYA field contains the following information:

   - A 4-word header containing some control information about the very extended PAB.
   - An array of work element queues in descending priority. For example, queue 1 is the first queue in the array, and it has the highest priority; queue 2 is the next queue in the array, and it has the next highest priority, etc. Each queue has the following structure:
     - (Field PABVFRST) A pointer to the first WKE (head, or oldest) on this level's queue
     - (Field PABVLAST) A pointer to the last WKE (tail, or youngest) on this level's queue
     - (Field PABVSRVL) Service level
     - (Field PABVSRVC) Service count

   The field PABRPHA in each PAB contains the address of an RPH that is either running or waiting.

   Note: In some PABs, PABRPHA might contain the address of an RPH, even though the RPH is not running or waiting.

Note the contents of these fields in each of the PABs, and have this information available when you contact IBM.

Figure 5 shows how to find each PAB. Figure 6 shows the relative location of fields in a normal, extended, and slightly extended PAB. Figure 7 on page 75 shows the layout for a very extended PAB. The DYPAB begins 10 bytes before the PAB.

Note: The PAB pointers shown in Figure 5 are not contiguous in the ATCVT, but are shown that way for demonstration purposes only.

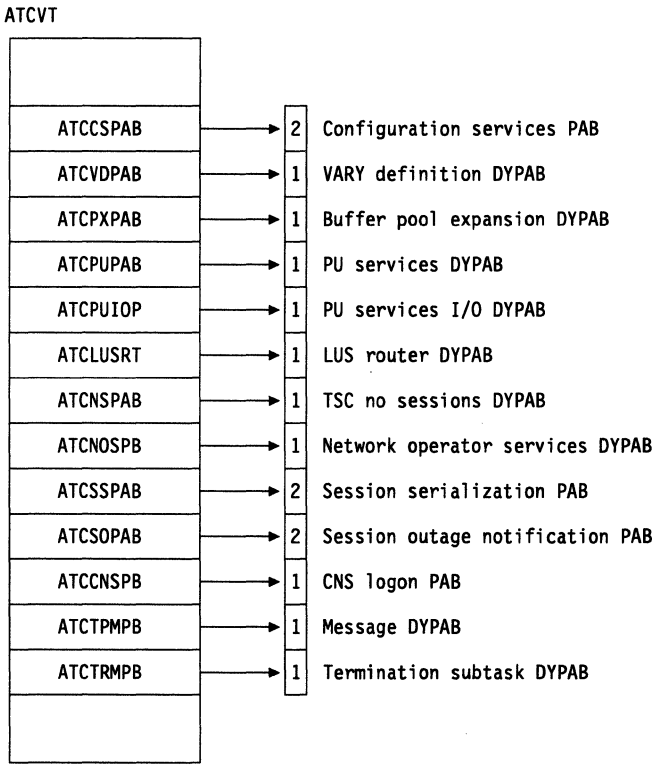


Figure 5. PAB Locations

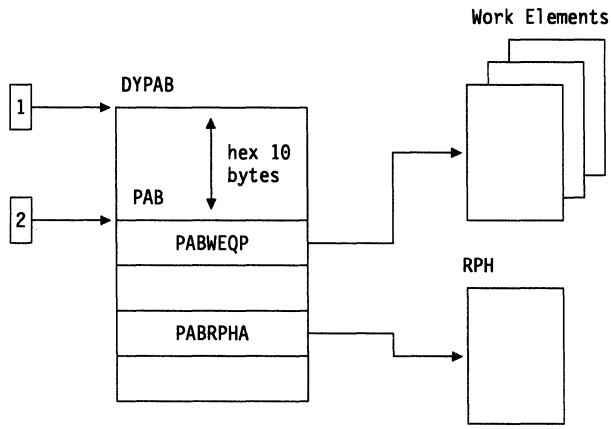


Figure 6. Normal PABs, Extended PABs, and Slightly Extended PABs

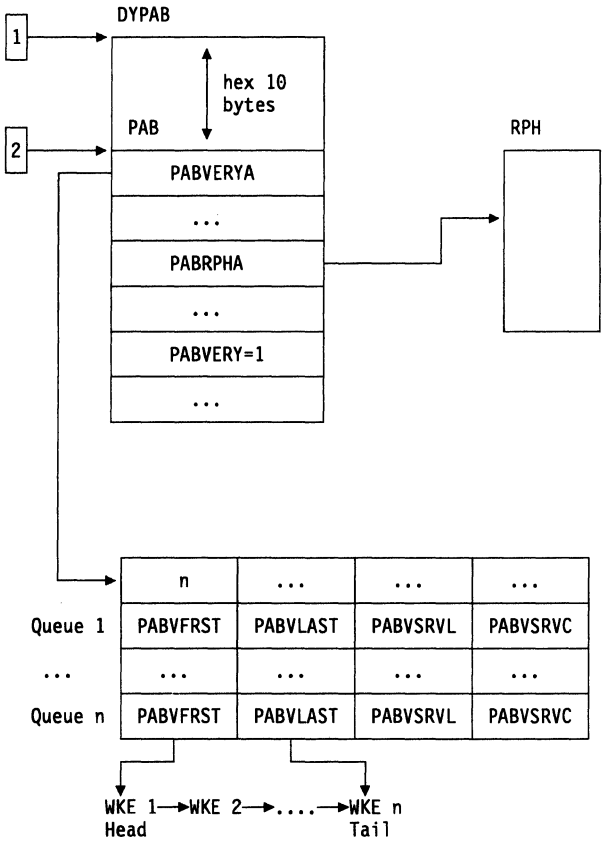


Figure 7. Very Extended PAB

7. **Is the wait caused by pending I/O?**

   Use the Input/Output Problem Determination (IOPD) facility to detect and report to the operator I/O operations that have been pending longer than a user-defined time limit. (For more information on using IOPD, see "Modify Input/Output Problem Determination" on page 209 and "MODIFY IOPD Command" in *VTAM Operation*.)

   When a VTAM process is waiting for a response, the process is represented by a waiting request element (WRE) queued to one or more LQABs within a single I/O LQAB group.

   The WRE points to an event ID (EID), which indicates the reason for the wait. (Wait state event IDs are described in Chapter 18, "Wait State Event Codes and IDs" in *VTAM Messages and Codes*.)

   Look for the WREs and corresponding EIDs in a dump by using Figure 8 on page 78 and Figure 9 on page 78 and the following steps.

   Note: MVS You can use the VTAMMAP VTWRE formatted dump tool to count or help analyze WREs. See "VTWRE" on page 344 for information on using VTWRE.

   a. Find the address of the ATCVT at low-storage address MVS VSE X'408' or VM X'200'.

MVS If this low-address location is not available in a dump, use the pointer in the MVS control block CVT (CVTATCVT) to find the VTAM control block AVT. Location X'00' in the AVT points to the ATCVT.

The ATCVT is identified by release level at offset X'00' in the ATCVT. For VTAM V4R2,

VM	VA42 (X'E5C1F4F240404040')
VSE	VS42 (X'E5E2F4F240404040')
MVS	VE42 (X'E5C5F4F240404040').

b. Get the address of the I/O LQAB-group hash table from field ATCIOLQB. This hash table contains a number-of-entries field (LQHENTNM) followed by an array of table entries numbered starting with zero.

c. Use the hash table to find the I/O LQAB groups for active subareas.

Each entry in the hash table is four bytes long and contains either 0, indicating an empty chain, or the address of the first LQAB group in a chain of I/O LQAB groups.

Within each I/O LQAB group, the LQGLINK field (offset X'10') contains the address of the next LQAB group in the chain. An LQGLINK value of 0 indicates the end of the chain.

- To find the I/O LQAB group for a specific subarea:
  - Calculate the hash table entry number, *N*, by dividing the subarea number by LQHENTNM and taking the remainder.
  - Search the chain for hash table entry *N* to find the LQAB group whose LQGSUBA field (offset X'0C') equals the subarea number.

  Note: I/O LQAB groups are allocated only when needed. Therefore, you do not find an LQAB group for a subarea that has had no I/O traffic.

- To find all I/O LQAB groups, search the chain for each entry in the hash table.

d. Find all the WREs chained off of a given I/O LQAB group.

- Each I/O LQAB group contains several different LQABs. Use the global LQAB (LQGGLOBL) to analyze wait states, because its chain contains all of the group's WREs. (Chains off of the other LQABs in the group usually do not contain all of the group's WREs.) You can locate LQGGLOBL at the beginning of the LQAB group (offset 0).
- The LQAB starts with the LQABFRST field, which contains either 0, indicating an empty chain, or the address of the first (oldest) WRE for this subarea.
- Within each WRE, the WREGFWD field (offset 4) contains the address of the next WRE in the chain. The end of the chain is indicated by a WREGFWD value equal to the LQAB address minus 4.

e. Find the waiting event. Each WRE contains a WREIDCD field (offset X'32') that identifies the waiting event. Use explanations of wait state event IDs in Chapter 18, "Wait State Event Codes and IDs" in *VTAM Messages and Codes* to find the reason for the wait. The address and length of the waiting event ID are in the fields WREIDP (offset X'24') and WREIDL (offset X'30'), respectively.

For additional information, check the WREDTA field (offset X'2C'). In most cases, this field contains a CPCB operation code. If so, look in Appendix G, "Control Point/Control Block (CPCB) Operation Codes" on page 1127 to determine what function the operation code represents.

8. Is the wait caused by a non-I/O CPWAIT?

   When a VTAM process has suspended itself using a CPWAIT and is waiting for a matching CPPOST or CPPURGE, the process is represented by a WRE queued to one or more LQABs within a single non-I/O LQAB group.

   Analyze non-I/O CPWAITs using the steps described for pending I/O in step 7 on page 75, with the following exceptions:

   - The IOPD facility does not detect and report these non-I/O events.
   - No arrays or hash tables are used. Instead, each of the six LQAB groups is pointed to directly by its own address field in the ATCVT. These address fields are as follows:

     ATCLUSMQ – logical unit services
     ATCMCQAB – miscellaneous command
     ATCPULQB – physical unit services
     ATCNOSQ – network operator services
     ATCSSLQB – SSCP session services 1
     ATCSSMQB – SSCP session services 2

   - WREs for non-I/O events do not contain a CPCB operation code value in the WREDTA field.

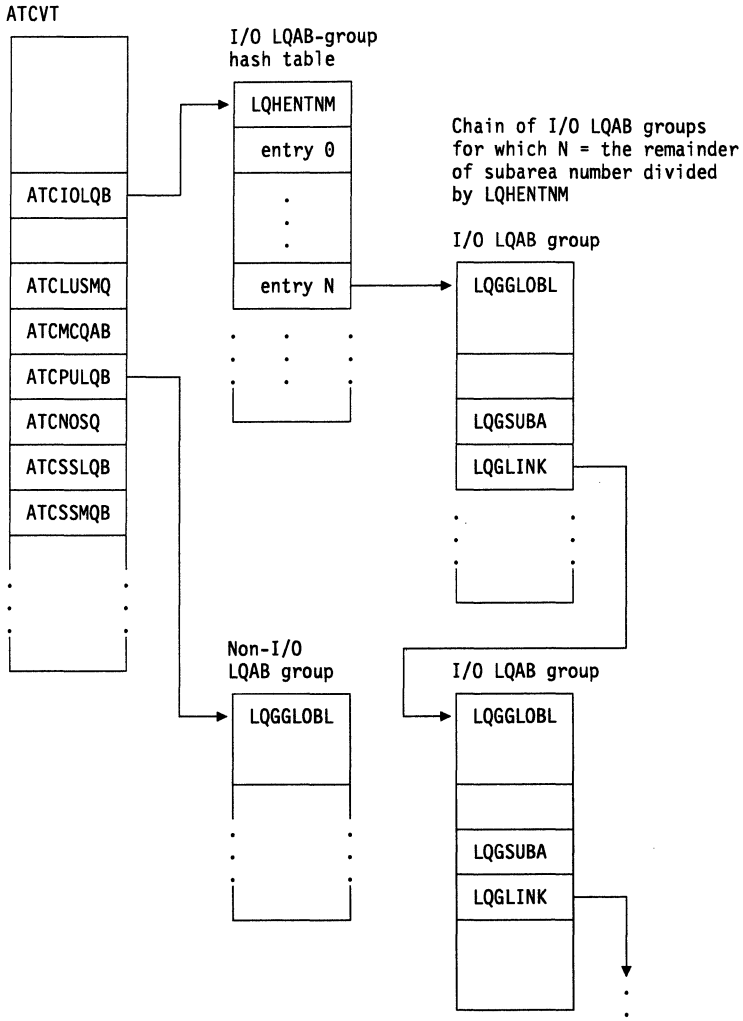


Figure 8. Finding LQAB Groups

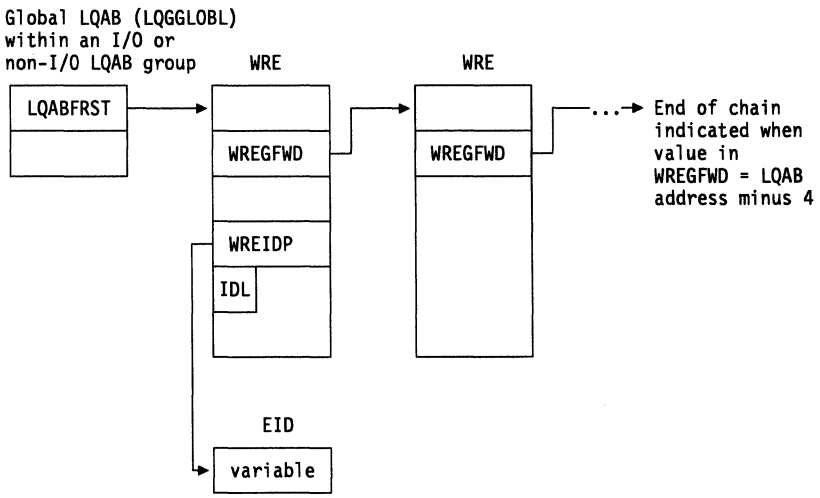


Figure 9. Finding Waiting Request Elements for an LQAB Group

9. **Find waiting RPHs.**

   The following steps give instructions for examining two kinds of wait conditions: (1) a process is waiting for a buffer, and (2) a process is waiting for some other resource. Both kinds of waiting processes are represented by *request parameter header* (RPH) control blocks, but the RPH is found in different locations for each type of wait condition.

   - Step 10 explains how to find RPHs queued from a buffer pool control block. These RPHs show that the buffer pool cannot supply the required buffers, and as a result, the process is waiting. Note which buffer pool cannot supply the required buffers.
   - Step 11 explains how to find RPHs that indicate a waiting process.

10. **Find RPHs queued from buffer pool control blocks.**

    A buffer pool that has no available buffers can cause a wait state. There are many reasons for running out of buffers (for example, incorrect allocation in the VTAM start options, a VTAM programming problem, or an application programming problem). Use the DISPLAY BFRUSE output obtained in step 2 on page 69, if you were able to get it, to analyze buffer pool usage. Or use the MVS VTAMMAP VTBUF and STORAGE formatted dump tools. See "VTBUF" on page 327 and "STORAGE" on page 320.

    VSE Examine location PCBFRPHA in each buffer pool control block.

    Also, follow the chain at offset X'04' into the RPH to obtain the addresses of other RPHs waiting for the same pool.

11. **Find other waiting RPHs.**

    Waiting RPHs indicate a VTAM process that has not been completed. To locate the waiting RPHs, search the large pageable buffer pool (LPBUF) by hand or use the MVS VTAMMAP VTRPH formatted dump tool. For more information see "VTRPH" on page 339. MVS VM Look at the formatted dump output.

    MVS Use the VTAMMAP VTBASIC formatted dump tool to analyze the request parameter headers (RPH) in the component recovery area (CRA). This function formats CRAs which contain RPHs. For more information see "VTBASIC" on page 326.

    Use the VM PRTDUMP command, specifying the VTBASIC option. This formats the CRAs, which contain the RPHs.

12. **Find RPHs waiting for locks.**

    a. For each waiting RPH, look at the CRALxPTR fields. If any pointer (PTR) fields are non-zero, check the corresponding bit in CRALKACT. For example:

       - If CRAL1PTR is non-zero, look at the last bit in CRALKACT.
       - If CRAL2PTR is non-zero, look at the next-to-last bit in CRALKACT.
       - If CRAL3PTR is non-zero, look at the third-from-last bit in CRALKACT.

       If the corresponding bit in CRALKACT is off (zero), the RPH is waiting for this lock. If the bit is on (non-zero), the RPH is holding the lock and might be waiting for another lock. On your list of waiting RPHs, add the name of the lock being held or waited for. (See Table 7 on page 80.)

b. If you cannot find any locks waiting or being held using step 12a, scan the LPBUF buffer pool again, and list all allocated buffers that contain a non-zero value in field CRALKACT. These buffers indicate which RPHs own locks, if any, and which locks are held. A CRA can hold several locks. For example, a value of X'06' indicates two locks being held: the RDTLOCK (X'04') and the VOCLOCK (X'02'). (See Table 7.)

   For each allocated buffer with a non-zero CRALKACT field, look at the CRALxPTR fields. (The buffer might contain a resume address.) A non-zero pointer field contains a lockword address. Find the lockword. The first word of the lockword shows a queue of RPHs waiting for that lock. Add these RPHs to your documentation list.

13. **Report the problem.**

    Go to "Reporting the Problem to IBM" on page 112.

VTAM Locks

Table 7 includes a description of each VTAM lock, and Figure 10 on page 83 provides information on VTAM lock pointers.

Table 7 (Page 1 of 3). VTAM Locks

Name	Lvl	Hex Value	Control Block	Field Name	Quantity	Function
8SLOCK	3	04	MPNCB	MPN8SLK	One per multi-path channel (MPC) line represented by an MPNCB	Serializes MPC outbound scheduling in a VTAM operating under MVS with multiple CPUs and System/390* hardware. Ensures single remover for TPREMEL macros.
ADJLOCK	5	10	ADJSA	ATCADJLK	One	Protects users of CIDCTL when adding or deleting an adjacent node.
AHNCBLOK	5	10	AHNCB	AHNLOCK	One per active APPN host-to-host channel PU	Serializes AHNCB PU PAB with AHNCB PC PAB.
BSBLOCK MVS	4	08	BSB	BSBLOCK	One per session using SNA over TCP/IP	Protects updates and references of the BSB.
CIDLOCK	8	80	CIT	CITLOCK	One per session	Serializes changes to or deletion of FMCB.
CONNLOCK MVS	6	20	SAACB	SAA_CONN	One	Serializes processing that establishes paths to TCP/IP.
CONVLOCK	9	100	CONVT	CONVTLOK	One per APPC conversation	Serializes deletions of RAB.
CRYTOKLK	8	80	ATCVT	ATCRYKLW	One	Serializes use of the session key token chain.

Table 7 (Page 2 of 3). VTAM Locks

Name	Lvl	Hex Value	Control Block	Field Name	Quantity	Function
DEBLKCT	6	20	DEBX3	DEBLKPTR	One per ACB index table entry	Serializes queuing of an application's API requests with the closing of an ACB.
DEBLOCK	5	10	ACDEB	ACDLOCK	One per OPEN application program	1. Protects FMCB queue off ACDEB. 2. Held by TSC and by OPEN or CLOSE.
DWALOCK	8	80	DWA	DWALOCK	One	Used by certain disabled TSC modules to serialize use of the disabled work area (DWA).
HNTELOCK	7	40	HNTE	HNTELOCK	One per minor node (per host element address)	Serializes updates and references to control blocks based off the HNTE (RDTE, NCB, LUCB, FMCB).
HNTERBLK	8	80	HNT	HNTERBLK	One per minor node	Serializes APPC conversion data in the RAB.
HNTLOCK	6	20	HNT	ATCHNTLK	One	Protects updates and references to HNT during most CIDCTL functions.
HSQCHAIN	5	08	ATCVT	ATCHSQLK	One	Serializes usage of the HSQH queues. One lock is used to protect all of the queues.
IAPTREE MVS	2	02	SAACB	SAAIAPLK	One	Serializes modifications and references to the IAP tree.
INNLOCK	9	100	ATCVT	ATCINNLK	One	Ensures that PIUs that are going to a node that is in slowdown mode are sent in FIFO order.
LMELLOCK	6	20	LME	LMELLOCK	One for every partner LU entry for every APPC application	Used to serialize access to partner LU information in the APPC logical unit mode (LM) Table.
LMHTLOCK	5	10	LMHDR	LMHTLOCK	One per APPC application	Used to serialize access to the APPC logical unit mode (LM) Table.
LSNLOCK	1	01	LSNCB	LSNLOCKW	One per PU connection to an IBM 3172 Interconnect Controller	Serializes the LSNCB PU PAB with the LSNCB PC and IR PABs.
NCBQ	9	100	ATCVT	ATCLNLOK	One	Serializes access to the queue of LSA NCBs anchored at ATCLNNCB.
PDBUFLK	9	100	ATCVT	ATCBUFLK	One	Allows the user to move in problem diagnosis trace data before the data is processed.

Table 7 (Page 3 of 3). VTAM Locks

Name	Lvl	Hex Value	Control Block	Field Name	Quantity	Function
PSTLOCK	8	80	ATCVT	ATCPSTLK	One	Serializes queueing and dequeueing of FMCB to PSTFMCB queue. Serializes release of PST storage.
QDCBLOCK	5	10	APNVT	APNQDCBL	One	Serializes access to the queue of QDCBs attached to the APNVT.
QUEUE	9	100	PAB	PABLOCK	One per extended PAB	Serializes queueing and dequeueing of work elements to an extended PAB.
RDTLOCK	2	04	ATCVT	ATCRDTLK	One	Protects users of CIDCTL (PAFIND). Obtained by PUNS when a network-addressable unit is to be added or deleted, or a use count decremented.
SDESLOCK MVS	8	80	SOCCB	SOT_DLCK	One per socket task (SOTCB)	Serializes usage of the socket descriptor list.
SKTASGN MVS	6	20	SAACB	SAA_ASGN	One	Serializes assignment of sessions to the socket tasks.
SKTLOCK MVS	8	80	SOCCB	SOT_LOCK	One per socket task (SOTCB)	Protects SOCCB chain off the SOTCB.
SLENTLOK	5	10	SLENT	SLE_LOCK	One per session list entry	Protects updates and references to the session list entry state indicators and to the sequential list of the TP work queue.
TASKLOCK MVS	7	40	SAACB	SAA_TASK	One	Protects SOTCB chain off the SAACB.
TCNLOCK MVS	3	04	TCNCB	TCNLOCK	One	Protects the BSB PCID and BSBSA tree for SNA/IP.
TREELOCK MVS	6	20	SAACB	SAA_TREE	One	Protects the SOCCB tree.
VDLOCK	9	100	ATCVT	ATCVDLOK	One	Serializes directed load processor.
VOCLOCK	2	02	ATCVT	ATCVOCLK	One	1. Serializes OPEN/CLOSE with VARY. 2. Serializes VARY Activate, VARY Deactivate, and VARY ERP.
VRLOCK	3	04	VRBLK	VRBLOK	One per virtual route	Serializes usage of the VRBLK.

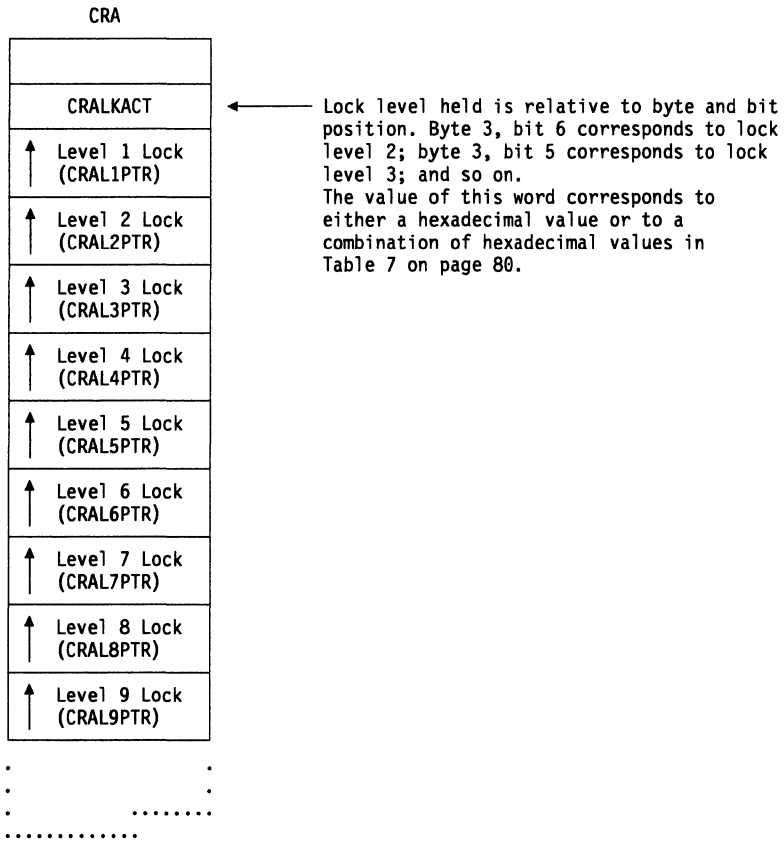


Figure 10. Pointers to VTAM Locks

Using the VARY INACT,FORCE Command

If the operator attempted a VARY INACT,FORCE command, check whether the command can be completed or whether there is a VTAM problem. Determine this using the following steps according to the resource specified on the VARY INACT command.

Note: Except for channel-attached SNA devices, if this command is issued to a resource with outstanding I/O, the command will not complete and VTAM must be recycled.

1. **Channel-attached physical unit or logical unit, SNA or non-SNA**:
   a. Display the resource status. If it is PHLIN, PHLAC, PDLUC, or PSUB1, the channel is hung or a required interrupt is missing.
   b. If the status is PNFYx, go to step 11 on page 85.
2. **Link-attached SNA logical unit, switched logical unit**:
   a. Display the resource status. If it is PNFYx, go to step 11 on page 85.
   b. If it is anything else, there is a VTAM problem.
3. **Link-attached BSC 3270 logical unit**:
   a. Display the resource status. You should see PDACL or PFDLU.
   b. Issue VARY INACT,FORCE for the NCP or CA major node that defines the device.
   c. If the deactivation succeeded and all lower-level nodes are inactive, message IST105I indicates this.

d. If the deactivation failed, display the status of all the resources in the NCP or CA major node.
e. If the status is PNFYx, go to step 11 on page 85.
f. If it is anything else, there could be a VTAM or NCP problem. Go to "Reporting the Problem to IBM" on page 112.

4. **Link-attached SNA physical unit, switched physical unit**:

   a. Display the resource status. You should see PDISC or PFDSC.
   b. Issue VARY INACT,FORCE for either the physical unit to which the device is attached, or for the NCP or CA major node that defines the device.
   c. If the deactivation succeeded and all lower-level nodes are inactive, message IST105I indicates this.
   d. If the deactivation failed, display the status of all the resources attached to the NCP.
   e. If the status is PSUBx, go to step 10.
   f. If the status is PNFYx, go to step 11 on page 85.
   g. If it is anything else, there is a VTAM problem.

5. **Link-attached BSC 3270 physical unit**:

   a. Display the resource status. You should see PDACP or PFDCP.
   b. Issue VARY INACT,FORCE for the NCP or CA major node that defines the device.
   c. If the deactivation succeeded and all lower-level nodes are inactive, message IST105I indicates this.
   d. If deactivation failed, display the status of all the resources in the NCP or CA major node.
   e. If the status is PSUBx, go to step 10.
   f. If the status is PNFYx, go to step 11 on page 85.
   g. If it is anything else, there is a VTAM problem.

6. **Local SNA or non-SNA major node, switched major node**:

   a. Display the resource status. You should see PSUBx.
   b. Issue VARY INACT,FORCE for any minor nodes that are not inactive. This should allow deactivation to be completed.

7. **Link**:

   a. Display the resource status. You should see PDLNK.
   b. Issue VARY INACT,FORCE for the NCP to which the link is attached. This should allow deactivation to be completed.

8. **Channel-attached NCP**:

   a. Display the resource status. You should see PDISC.
   b. Press the RESET LOAD button on the communication controller. This should allow deactivation to be completed.

9. **Link-attached NCP**:

   a. Display the resource status. You should see PSUBx.
   b. Display the status of the lower-level nodes.
   c. If the status is PNFYx, go to step 11 on page 85.
   d. If the status is anything else, there is a VTAM problem.

10. **PSUBx status**:

    a. Display the status of the lower-level nodes to find any pending states.

b. Deactivate any active or pending nodes. This should allow deactivation to be completed.

11. **PNFYx status:**

    a. **For application programs with an NSEXIT exit routine:**

       If the VARY INACT,FORCE command is unable to complete, there is a VTAM problem. Otherwise, deactivation should complete.

    b. **For application programs with only a LOSTERM exit routine:**

       1) If the application program has issued a CLSDST macroinstruction, deactivation should complete.

       2) If the application program has not issued a CLSDST macroinstruction for the logical unit, issue a second VARY INACT,FORCE for the logical unit in question. If that does not correct the problem, you might need to cancel the application program to allow the deactivation to complete. (Cancelling the application program terminates all of the LU-LU sessions with the application program.)

    c. **For application programs with neither exit:**

       Deactivation does not complete until the application program issues CLSDST, the application program closes its ACB, or the operator cancels the application program.

Return to step 2 on page 69.

Loop

If the problem is a loop, use the procedure in Figure 11 on page 87 to collect the following documentation.

Note: MVS If you are using TSO/VTAM, use this procedure. You do not need to go to Chapter 3, "Collecting Documentation for TSO/VTAM Problems (MVS)."

- System console log
- Messages associated with the loop (if any)
- Failing module ID
- Dump output:
  - MVS Dump of the VTAM address space that is looping
  - VM Dump of the virtual machine that is looping
  - VSE Dump of the VTAM partition, SVA, and supervisor
- MVS VM Error file output (LOGREC)
- VSE SYSREC
- For a problem associated with a specific device:
  - VIT trace output (all except LOCK)
  - I/O trace output
  - Session trace data (if using the NetView program)
  - Session awareness data (if using the NetView program)
  - NetView report (if using the NetView program)
  - MVS GTF CCWTRACE output (if available)
  - VM VM branch trace
  - VM VSCS internal trace
  - VM CP internal trace
  - VM CP virtual machine trace
  - VM GCS internal trace
  - VSE SDAID CCWD trace output.

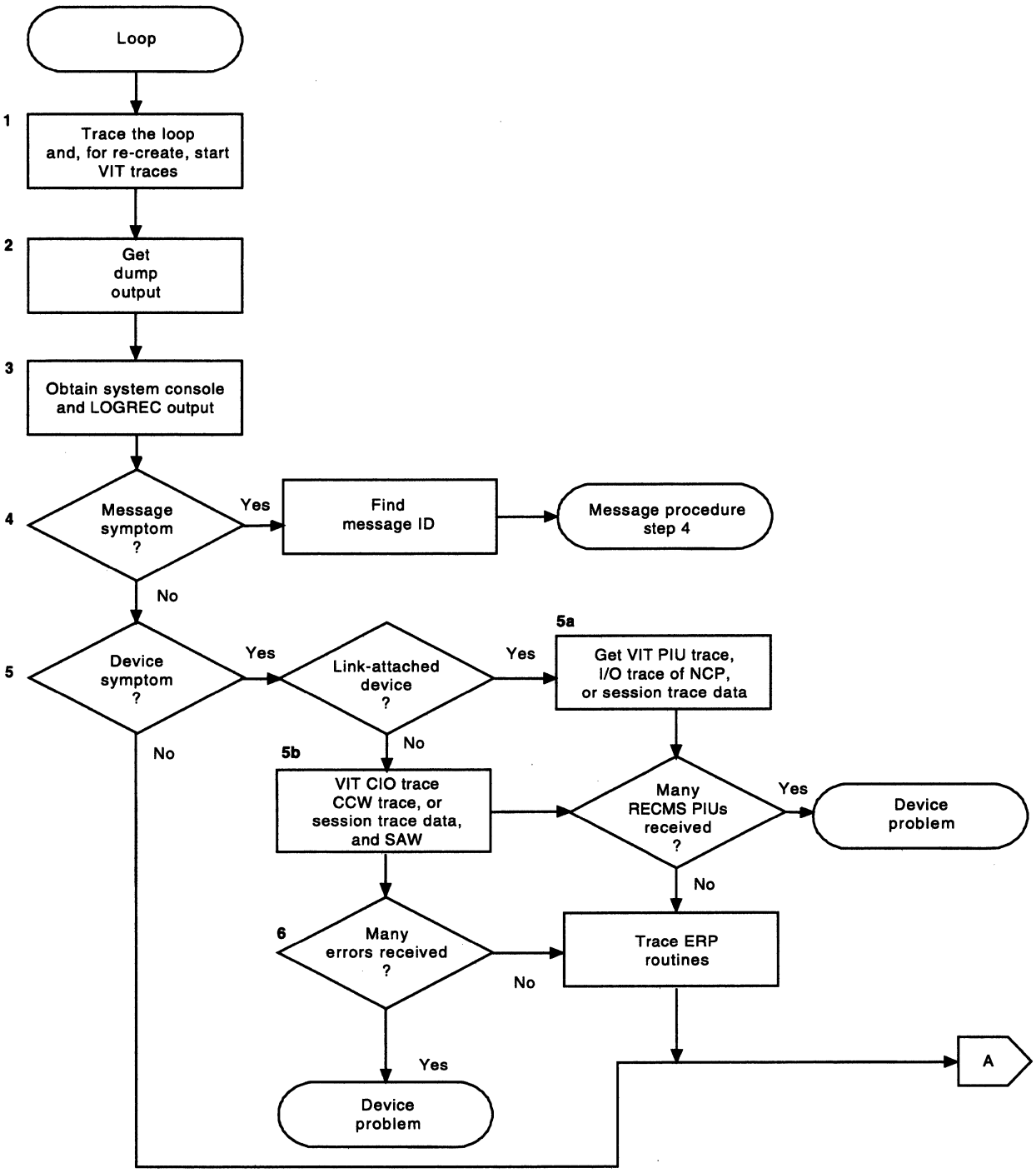


Figure 11 (Part 1 of 2). Overview of the Loop Procedure

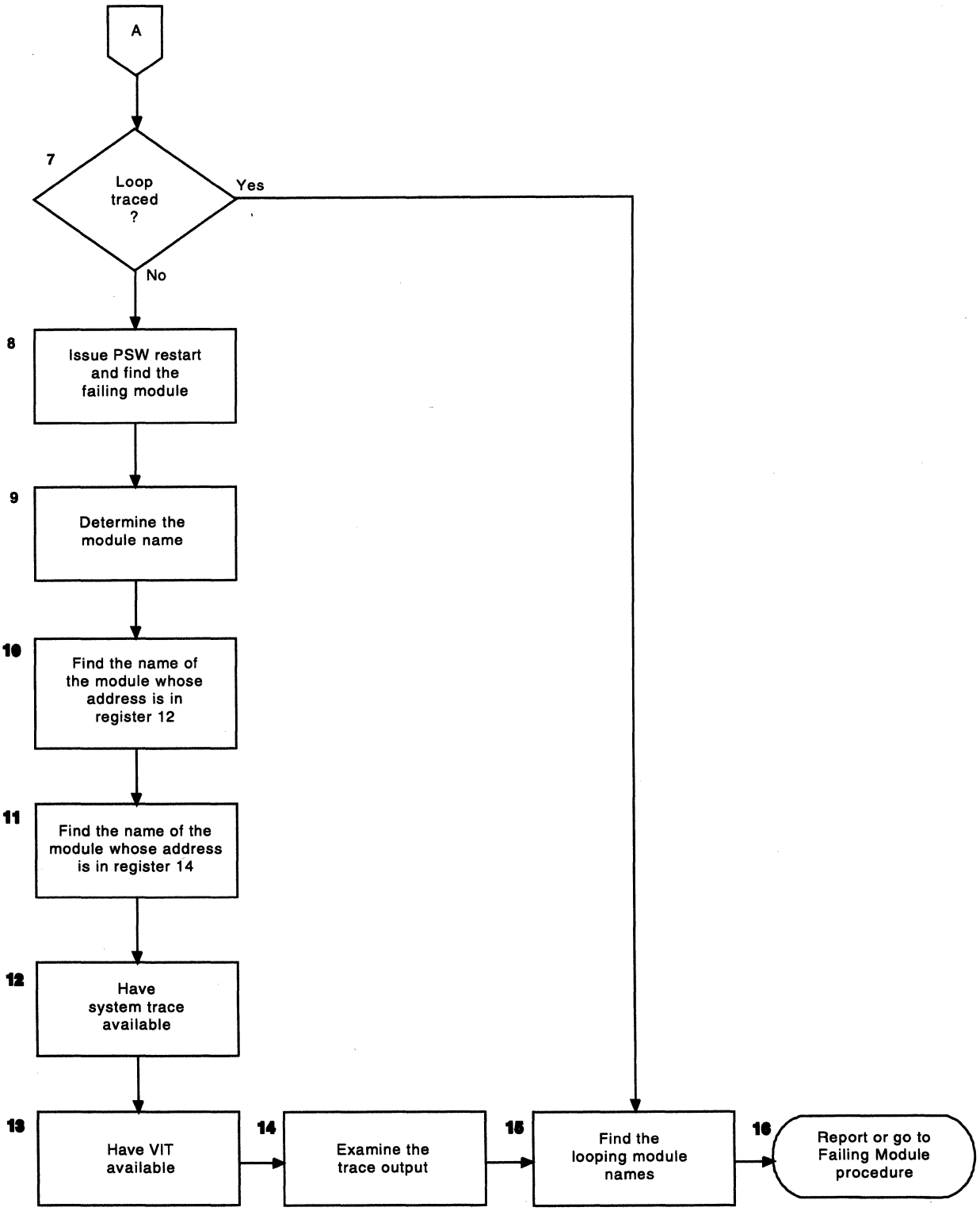


Figure 11 (Part 2 of 2). Overview of the Loop Procedure

Following is a description of each step shown in Figure 11 on page 87.

1. **Trace the loop.**

   Loop problems might involve many modules or a single module. If possible, trace the looping instructions. Using the *Operator's Reference* for your host processor, instruction-step through the looping addresses. Save these addresses for use in diagnosing the problem.

   VM VSE In an enabled loop (PSW is enabled for interruptions), PSWs with low addresses might indicate a normal system process. Because most VTAM routines are enabled, look for PSW addresses that are in the range of the VM VTAM virtual machine, VSE partition, or SVA.

   MVS Take a dump and determine which module is looping by checking the PSW addresses in the CLKC entries for a repeating pattern.

   If the VIT was running when the loop started, look for any exception conditions that might have led to the loop. If the internal trace was not running, you might have to re-create the problem to get the trace at the time of the loop. Set the internal trace to MODE=EXT to record the trace entries in an external file.

   VM Display the PSW and REGS several times to trace the loop, using the command #CP D P G.

   VSE When VSE is native on the CPU, stop the processor and instruction-step the CPU to get the PSWs of the loop. When VSE is a guest of VM, get the PSWs with the command #CP TRACE I ALL. Display the PSW and REGS several times to trace the loop, using the command #CP D P G.

2. VM **Look for a VSCS loop.**

   If you are using VSCS, go to "Wait and Loop Problems" on page 159. If you cannot identify the problem using those procedures, return to this procedure.

3. **Get dump output.** Get a dump of VTAM:

   - MVS Issue the DUMP command, or press the Program Restart key.
   - VM Issue #CP VMDUMP 0- DSS FORMAT GCS, with the appropriate options.
   - VSE Issue the DUMP command.

   MVS VSE If the loop is disabled, the system console is not available for input, so take a stand-alone dump. (See "Stand-Alone Dump" on page 214 or "Stand-Alone Dump Utility" on page 216.)

4. **Get the system console log and LOGREC or SYSREC output.**

   The system console log might contain information, such as error messages, that can help you diagnose the problem. Also, print the LOGREC or SYSREC file.

   MVS Use the LOGDATA option to print the in-core LOGREC buffers. See Table 80 on page 1165 to determine what book has information on LOGDATA.

5. **Is a message involved?**

   Determine whether there are any messages associated with the loop, such as a particular message always preceding the problem, or the same message being issued repeatedly. If so, add the message numbers to your problem documentation and go to the message procedure, step 4 on page 94.

6. **Is it a device error?**

For any kind of device error, you should first check the NetView report (if you have the NetView program) and then the LOGREC or SYSREC output.

Does the LOGREC or SYSREC output show repetitive entries for the same kind of error on a particular device? If so, VTAM is receiving several different errors from that device.

a. If the LOGREC or SYSREC error records are for a link or link station attached to a communication controller, get VIT PIU records and an I/O trace of the NCP. If you have the NetView program, get session trace data or session awareness data for the NCP. If the error records are for a link or device attached to a communication adapter, get VIT PIU records or a dynamic trace of the communication adapter.

   If the trace shows continual arrival of RECMS PIUs, then the repetitive entries in LOGREC or SYSREC are caused by a device error.

   Note: For information on counting PIUs see "Counting Request/Response Units (RUs)" on page 426.

b. For channel-attached devices, use one or more of the following traces for the device to determine whether VTAM is receiving many errors:

   - VTAM internal trace with CIO option
   - Session trace data (if using the NetView program)
   - Session awareness data (if using the NetView program)
   - MVS CCWTRACE (if available)
   - VM SIO and I/O traces for CP and GCS
   - VSE SDAID program with the SIO and I/O trace types and the output of CCWD.

If VTAM is receiving many errors, the problem is probably in the device. Run a CIO VIT trace to trace execution of the VTAM ERP routines. Then continue with step 7.

7. **Is the loop traced?**

   If you were able to instruction-step through the loop, go to step 15 on page 92; otherwise, continue with step 8.

8. **Find the failing module.**

   Use the PSW to find the failing module.

   - MVS The PSW is found in LOGREC output, the SDWA, or the RTM2WA.

     When you use PSW RESTART to terminate a looping task, a LOGREC entry is created with a completion code of X'071' for the task. An RTM2WA is also created for the task. Use the LOGREC record and the RTM work area to locate the failing module. See the diagnostic books listed in the "Bibliography" on page 1159 for your operating system for help in locating the PSW in dump output.

     Depending on the PSW bit 32, the last 3 bytes (24-bit mode) or 4 bytes (31-bit mode) of the PSW contain the address being executed at the time of the dump. Scan the dump output to find the address given in the PSW. See Table 80 on page 1165 to determine what book contains more information on PSWs.

     Note: Addresses might not always be in numeric order because the dump does not always generate output in sequential order.

If you cannot find the address, the dump might not contain the relevant portion of main storage. For example, the address might be in LPA storage. Have this portion of storage dumped, or use output from LPAMAP to identify the module, and proceed as above.

- VM You should have found the PSW when you issued DISPLAY PSW in step 1 on page 89.
- VSE You should have the PSWs from step 1 on page 89.

  Note: Addresses might not always be in numeric order because the dump does not always generate output in sequential order.

Note: MVS The VTAMMAP VTFNDMOD formatted dump tool can be used to gather the module information described in steps 9, 10, and 11.

9. **Find the module name containing the failing address.**

   VTAM identifies modules with an EBCDIC module name and the Julian date (and, if appropriate, the latest PTF applied) at or near the beginning of most modules. This module identifier is usually in the form:

   IST*xxxxx yy.ddd [nnnnnnn]*

   where *xxxxx* is the last five characters of the module name, *yy.ddd* is the Julian date the module was assembled, and *nnnnnnn* is the latest PTF (if any) that has been applied to this module.

   To find the module ID, start at the failing address and scan upward (in descending address order) along the right side of the dump listing. The module ID is printed in EBCDIC. Add the module name to your documentation list.

10. **Find the module pointed to by register 12.**

    General register 12 (X'0C') is normally the base register for VTAM modules. In a VTAM loop, register 12 should point to the same module found in step 11. If not, add this module name to your documentation list.

11. **Find the module pointed to by register 14.**

    General register 14 (X'0E') might point to a module that called the routine that is looping. Add this module name to your documentation list.

    Add the module names from steps 9, 10, and 11 to your documentation list. You can report the problem next, but you might need to continue with step 12.

12. **Get the system trace output.**

    The system trace might show many external and I/O interrupts. The PSW addresses in system trace entries will be part of the loop.

13. **Get the VIT output.**

    The VIT is useful in determining the reason for a loop, such as a process being continually re-dispatched for the same request. Get the VIT output. If you require VIT options in addition to the default options (API, MSG, PIU, SSCP, and NRM), start a VIT in addition to the default and specify MODE=EXT. If VTAM does not accept the command it might be necessary re-create the problem. For more information on using the VIT, see Chapter 9, "Using the VTAM Internal Trace (VIT)."

14. **Examine the trace entries.**

By examining all of the trace entries, you might be able to determine whether there is a loop. The most obvious loops would be a module or modules getting continual control of the VTAM system, or a control block chaining to itself. Check the output of the PSS option to see which VTAM routines are getting control. If you see a pattern of repetition in the trace entries, it does not necessarily mean VTAM is looping. Some VTAM processes are timer-driven and repeat periodically.

Notes:

a. Get the trace information and examine the clock comparative entries for repeating PSW addresses. For short loops the repeating PSWs show the extent of the loop

b. The absence of any apparent loop does not necessarily mean VTAM is **not** looping. The loop might not contain a VTAM trace point.

If a module or modules are looping, get their addresses from the trace entries. Step 15 explains how to find the module name.

If you find a control block chained to itself, or if a queue of control blocks is in a cycle, try to identify the control block. Most control blocks have a 1-byte ID at offset X'00'. See the control block ID codes in Appendix H, "Storage and Control Block ID Codes" on page 1137 to identify the control block name.

15. **Find the module names.**

    Note: MVS You can also use the VTAMMAP VTFNDMOD formatted dump tool to find the module ID. See "VTFNDMOD" on page 332.

    Use the addresses found in step 14 on page 91 to find the module names involved in the loop.

    To find the module ID, start at the failing address and scan upward (in descending address order) along the right side of the dump listing. The module ID is printed in EBCDIC. Add this module ID to your documentation list. Continue with step 16.

16. **Report or go to the failing module procedure.**

    If you determined the module names, go to "Failing Module" on page 109. Otherwise, you are ready to contact IBM. Go to "Reporting the Problem to IBM" on page 112.

Message Problem

If the problem is a message, use the procedure in Figure 12 to collect the following documentation:

- Issuing module
- Message number
- System console log
- Dump output if required
- VIT output.

Note: If your installation changed the text of the message, the message ID might not be included, or might not match the ID of the message as it appears in *VTAM Messages and Codes*. Therefore, it is recommended that you re-create the problem using the VTAM-supplied message text. Otherwise, determine what VTAM-supplied message text corresponds to the message text your installation is using.

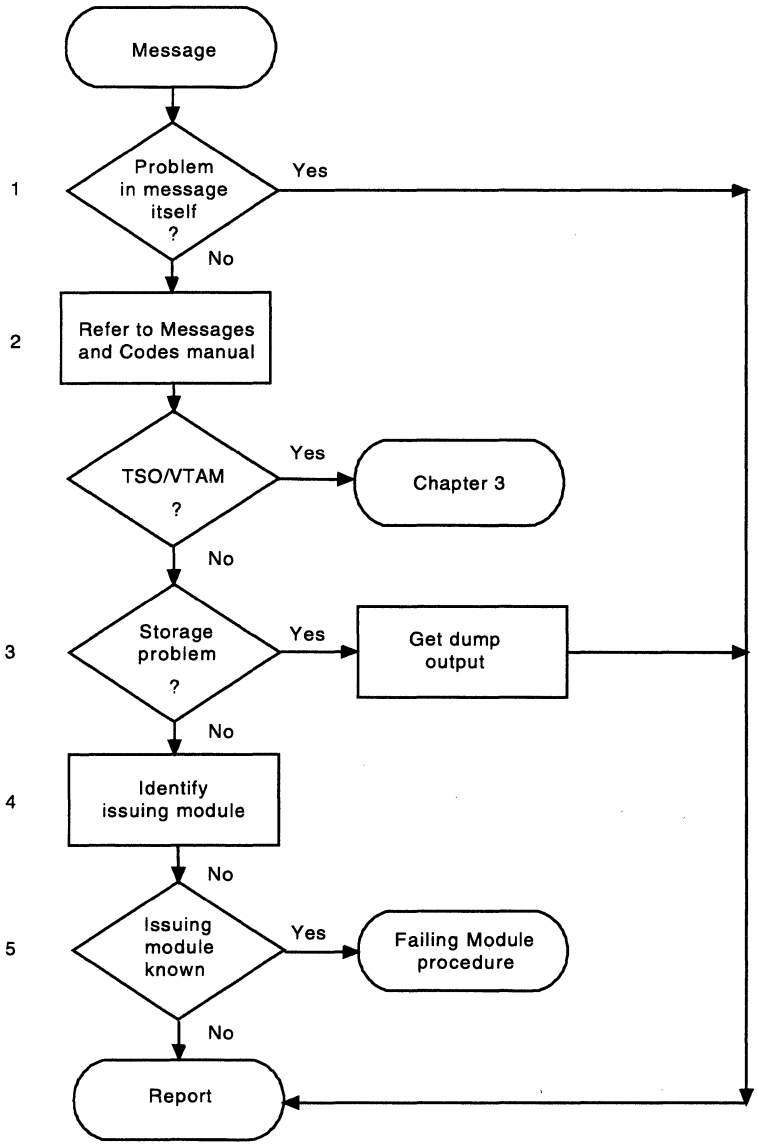


Figure 12. Overview of the Message Procedure

Following is a description of each step shown in Figure 12.

1. **Report if the problem is in the message itself.**

   If the content of the message is incorrect or the meaning of the message is not clear, go to "Reporting the Problem to IBM" on page 112.

2. **Follow the recommended action.**

   For all messages, see *VTAM Messages and Codes* for recommended operator and programmer actions. (See the list of VTAM books in the "Bibliography" on page 1159 for the appropriate form number.) In addition:

   - Identify the issuing component.
   - If the message indicates a storage problem, go to step 3.
   - MVS If the message indicates a TSO/VTAM problem, see Chapter 3, "Collecting Documentation for TSO/VTAM Problems (MVS)" on page 113.
   - VM If message DTIS99I indicates a VSCS problem, see "Message Problems: Message DTIS99I Indicates a VSCS Internal Error" on page 176.

   The following list shows message prefixes and the components that issue those messages.

Prefix	Issuing Component
ELM	MVS Logon Manager
DTI	VM VSCS
IKT	MVS TSO/VTAM
IST	VTAM
ISU	MVS Sockets over SNA

   Notes:

   a. If the message starts with any other characters, it is issued from another network component or the operating system.

   b. Messages that begin with the prefix **ISTF** are issued by the MVS VTAM dump analysis tools and the MVS VM VTAM internal trace (VIT) analysis tool.

      Help information for **ISTF** messages is available as a part of each tool by pressing F1. Therefore, these messages are not documented in *VTAM Messages and Codes*.

      See Chapter 7, "Using VTAM Dump Analysis Tools (MVS)" on page 237 and Chapter 10, "Using the VIT Analysis Tool (MVS, VM)" on page 411 for additional information on the dump and trace analysis tools.

3. **Is there a storage problem?**

   If there is a storage problem, see "Storage Problem Procedure (MVS)" on page 105 for additional information.

   If there is not a storage problem, continue with step 4.

4. **Identify the issuing module.**

   Try to identify the module issuing the message. If the MSGMOD start option is active or the MODIFY MSGMOD command is issued before the problem occurs, the message text contains the last five characters of the issuing module name. Add the message prefix to the module name, and add this name to

your problem documentation. (To modify the module identifier in messages, see "Modify Message Module Identification" on page 210.)

The VTAM internal trace MSG entries contain the message number, the save area address, and the module ID (the 4th, 5th, 6th, 7th, and 8th characters of the module name). Use these to identify the issuing module. If the trace entry contains no module identifier, use the caller's address from the trace entry.

5. **Report or go to the failing module procedure.**

   If you know the name of the issuing module, go to "Failing Module" on page 109. If you are unable to determine the issuing module or resolve the problem, go to "Reporting the Problem to IBM" on page 112.

Incorrect Output

If the problem is **incorrect output**, use the procedure in Figure 13 on page 97 to collect the following documentation:

- Specific output that is incorrect
- Device type (if appropriate)
- Buffer contents trace output
- VIT trace output (all except LOCK)
- Session trace data (if using the NetView program)
- Session awareness data (if using the NetView program)
- Network controller line trace output (3710 only)
- GTF CCWTRACE output (if available) MVS
- For TSO problem, TGET/TPUT trace MVS
- CP internal trace VM
- CP virtual machine trace VM
- GCS internal trace VM
- For VSCS problem, VSCS internal trace VM
- SDAID program output VSE
- ICA line trace output VSE VM
- Network problem:
  - Line trace
  - Generalized PIU trace (GPT)
  - Transmission group (TG) trace
  - Scanner interface trace (SIT) for 3720, 3725, and 3745.

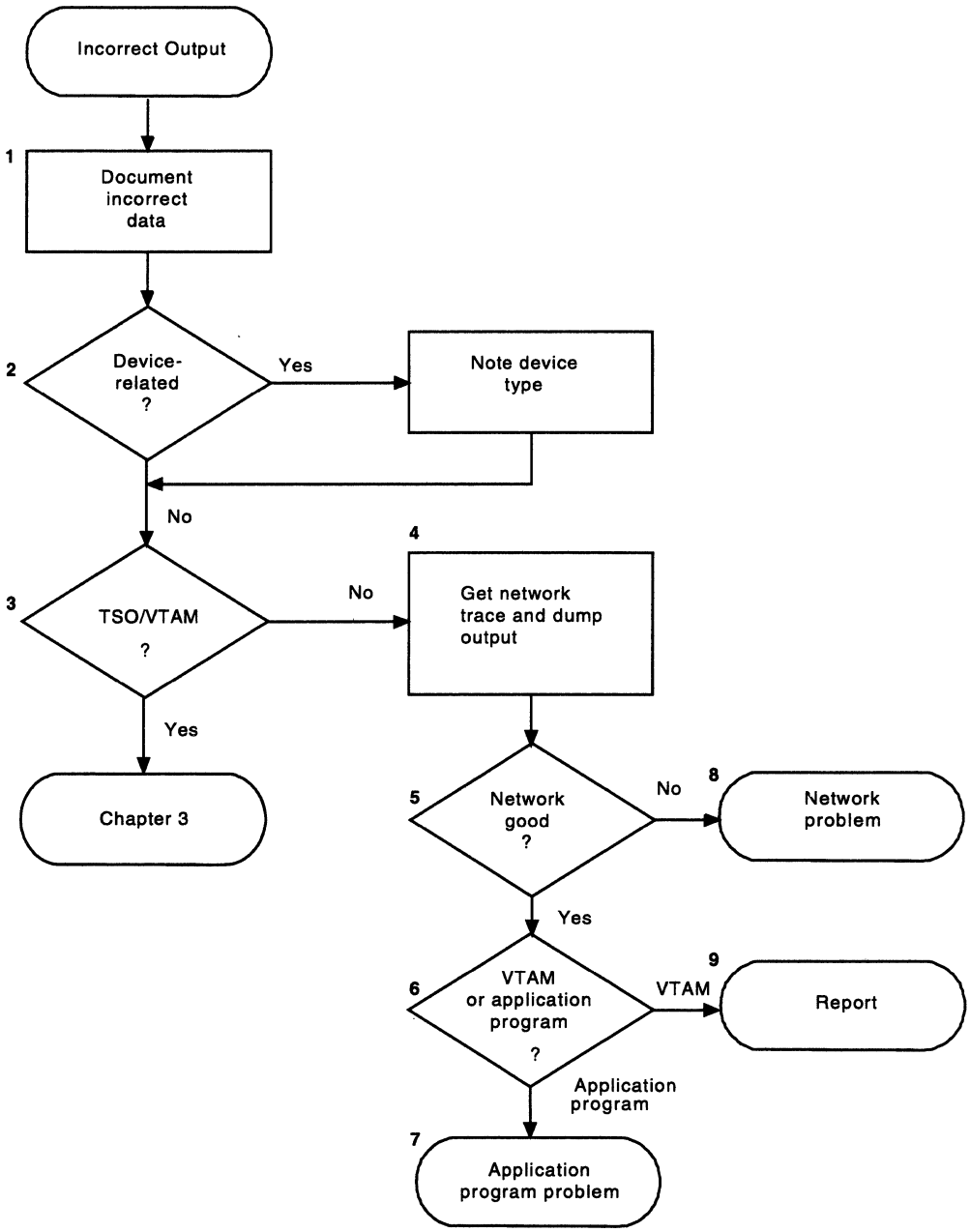


Figure 13. Overview of the Incorrect Output Procedure

Following is a description of each step shown in Figure 13.

1. **Document the incorrect data.**

   Most incorrect output problems pertain to data contained in a PIU. This can be incorrectly formatted user data, routing information that is not valid, or other types of missing or incorrect data. These types of problems are generally difficult to diagnose, as they show up only at the end-user locations. From the following list, choose all the symptoms that apply to your problem and add them to your problem documentation:

   - Cursor is in the wrong place or missing.
   - Format of data is incorrect.
   - Data is truncated.

- Data is incorrect.
- Data is missing.
- Problem is related to an application program macroinstruction.
- Screen is improperly formatted.
- Session is hung.
- Terminal is hung.

Note: The term "hung terminal" means the user is prevented by the system from entering data.

2. **Determine the device type.**

   If the problem is associated with a specific device type, add the device type (for example, 3277, 3278, or 3279 terminal) to your problem documentation.

3. MVS **Is it a TSO/VTAM user problem?**

   If the incorrect output problem involves TSO/VTAM, go to "Screen Management Problems" on page 121.

4. VM **Is it a VSCS user problem?**

   If the problem involves a VSCS user, follow the procedure in "Incorrect Output Problems" on page 178.

5. **Get network trace and dump output.**

   Re-create the problem with the following service aids active:

   - Start the VTAM buffer contents trace for the failing application program and terminal logical units.
   - If you have an IBM 3710 Network Controller, start the network controller line trace. It traces information passing to and from a 3710.
   - Start the VIT with MODE=EXT. Unless APPC is involved, do not specify the APPC option. Unless you suspect that a lock is not working, do not specify the LOCK option. Do not specify the MSG option. This shows the type of requests being processed between the application program and the end user or terminal, and the control information for routing, pacing, and so on, in each PIU sent in the network.
   - You should start the generalized PIU trace if the problem involves lines or devices attached to a communication controller. This shows how far the PIU got within the NCP and what the PIU looked like (its control information) when it was sent to the line.
   - VM VSE If the problem involves lines or devices attached through an integrated communication adapter (ICA), start the dynamic trace of the ICA. This shows any problem related to the line.
   - If you have the NetView program, you can use the session trace data to determine the requests and responses received and sent by VTAM and the other network nodes.
   - If it is available, you can use MVS GTF CCWTRACE and VM TRSOURCE and VSE the SDAID program to trace the CCWs, I/O interruptions, and all CCW data for each Start I/O issued by the system. For more information on these traces, see the diagnostic books for your operating system.

- As soon after the problem occurs as possible, take a dump of the application program, VTAM, and MVS TSO/TCAS. Stop all traces, and format the dump and trace output for online viewing.

The dump is used to reference storage addresses, such as control blocks and module entry points. The trace data shows at what point the data was modified, and what PABs the data was on as it was processed by VTAM. Take the dump during the re-create, when the traces are running. A dump taken earlier might not be accurate because the terminal device might have been deactivated and reactivated. This would allocate a different set of control blocks.

For more information on dumps and traces, see Chapter 6, "Using Dumps" and Chapter 8, "Using Traces." Operating system service aids are documented in operating system books.

6. **Examine the trace output.**

   Examine the individual trace entries to find the failure. If the problem concerns user data format, and the buffer contents trace or PIU trace does not show the incorrect data, use the output from the VIT trace with the SSCP option.

   Use one of the following commands to see whether data is correct when it is sent to the NCP or logical unit:

   - MVS GTF CCWTRACE (if available)
   - VM TRSOURCE
   - VSE SDAID.

   The VTAM internal trace records CC2, CI2, and CO2 contain the first 24 bytes of this data.

   When output data is correct:

   If the traces show that the data or the control information in the RH/TH as it leaves VTAM is correct, the problem is not in VTAM or the application program; go to step 9. If the data going to the network is not valid, continue with step 7.

   When input data is incorrect:

   If the traces show that VTAM is receiving invalid data from a source external to VTAM, the problem is in the network; go to step 9. If the data from the network is valid, the problem is in VTAM or an application program; continue with step 7.

7. **Is it VTAM or an application program?**

   The problem has been narrowed down to VTAM or the application program. Examine each trace entry to determine whether the information from the application program was incorrect. If VTAM seems to be responsible, go to "Reporting the Problem to IBM" on page 112; otherwise, continue with step 8.

8. **Is more application program help needed?**

   For IBM application programs such as CICS or IMS, you can find additional diagnostic help in the IBM application program books. If you decide that the problem is with an IBM application program, contact the appropriate IBM representative for that product.

9. **Is the problem with an external network device?**

The problem has been narrowed down to the VTAM network, but not to VTAM itself. Try to identify the device or program responsible. You can use service aids, such as the NCP line trace, generalized PIU trace, or transmission group trace, to trace data flow between the NCP and terminal logical units. For information on how to use these traces, see "Traces Provided by NCP" on page 384. For a 3720, 3725, or 3745, use the scanner interface trace (SIT) to distinguish between NCP problems and line or terminal problems.

If you suspect the NCP, see Table 80 on page 1165 to determine what book contains information on troubleshooting NCP problems.

Chapter 6, "Using Dumps" on page 213 explains how to use system dumps, including the NCP dump. Contact the appropriate IBM representative for the device or program identified as the cause of the problem.

VM VSE The ICA line trace can be used to discriminate between a VTAM error and one that is related to a device or line attached through an ICA. For more information, see the diagnostic book for the host processor.

10. **Report the problem.**

    Go to "Reporting the Problem to IBM" on page 112.

Performance Problem

If the problem is performance, use the procedure in Figure 14 on page 102 to collect the following documentation:

- System console log
  - MVS VM LOGREC
  - VSE SYSREC
- Tuning statistics
- SMS (buffer use) trace output
- Network controller line trace output (3710 only).

Note: Performance problems do not generally indicate a VTAM problem.

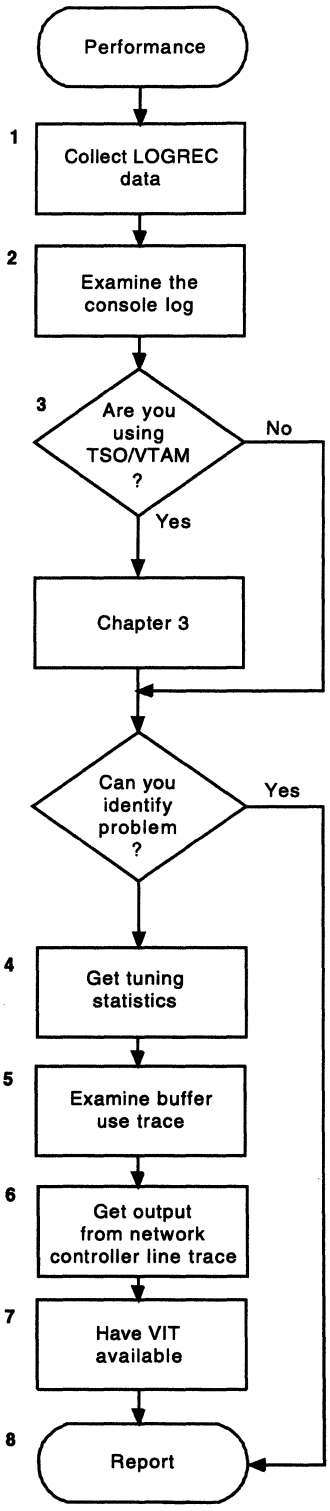


Figure 14. Overview of the Performance Procedure

Following is a description of each step shown in Figure 14 on page 102.

1. **Get LOGREC or SYSREC output.**

   Performance problems are often caused by hardware errors. These hardware errors cause software error recovery processing to occur, which degrades system performance. For this reason, you should get LOGREC or SYSREC output. LOGREC or SYSREC might show a large number of hardware errors for a particular device or group of devices. If the errors are limited to a single device, a hardware error is probably the cause. If the errors appear on many or all terminals of one type, software is more likely to be the problem, although hardware might still be at fault. If you suspect a particular device type, add it to your documentation list.

2. **Examine the system console log.**

   The system console log might contain messages to help diagnose a problem. Add the message ID to your documentation list:

MVS	The message prefix is either IST, ELM, IKT, or ISU.
VM	The message prefix is either IST or DTI.
VSE	The message prefix is IST.

   The system console log might also contain information about command problems. For example, operator commands might be taking too long to complete. Add the command name (for example, VARY ACT) to your documentation list.

3. MVS **For TSO/VTAM, see Chapter 3, "Collecting Documentation for TSO/VTAM Problems (MVS)" on page 113.**

   If you are using TSO/VTAM, go to "Performance Problems" on page 129. If you cannot resolve the problem with that procedure, return to this procedure.

4. VM **For VSCS, see Chapter 4, "Collecting Documentation for VSCS Problems (VM)" on page 131.**

   If you are using VSCS, and you have not identified the cause of the problem, go to "Performance Problems" on page 182. If you do not resolve the problem with that procedure, you should return to this procedure.

5. **Get tuning statistics.**

   If the performance problem is associated with traffic through a channel-attached host, a channel-attached communication controller, a channel-attached SNA physical unit, or multipath-channel-attached resources, it might be helpful to get tuning statistics for VTAM. (For more information on tuning statistics, see "Modify Tuning Statistics" on page 212.)

6. **Get output from the SMS (buffer use) trace.**

   You might have enough information to identify the problem. If so, go to "Reporting the Problem to IBM" on page 112. If you do not, continue with this step.

   a. Buffer pool expansion can cause performance problems. During VTAM initialization, error recovery, and VARY command processing, buffer usage is higher than normal. If buffer expansion is used, buffer pools should not expand except during such peak periods. Thus, what appears to be high buffer usage could be normal depending on the level of system activity.

      Run the buffer use trace (TYPE=SMS). For information on how to start the trace and examine the output, see "SMS (Buffer Use) Trace" on page 377.

Coding the SNAPREQ start option causes trace entries to be written more often, providing a more comprehensive picture of buffer usage. See "SNAPREQ" in the *VTAM Resource Definition Reference*.

b. MVS Using the time stamps in the system console and buffer use trace, correlate an excessive number of buffer pool expansions or large number of buffers used from a single pool with network activity recorded on the console. Constant high usage of a buffer pool might show that not enough buffers were allocated at VTAM initialization to properly support the level of network activity. Also look for a buffer pool that continually grows; buffers might not be released by some VTAM routines. Add the name of a particularly active buffer pool (for example, LPBUF or IOBUF) to your documentation list.

7. **Get output from the network controller line trace.**

   If an IBM 3710 Network Controller is installed, start the network controller line trace. This traces information passing over the lines to and from a 3710. (For more information about this trace, see "Network Controller Line Trace (3710 Only)" on page 389.) Print the trace output with TAP.

8. **Get additional documentation.**

   If no solid indication of a problem is apparent at this point, run the VIT with OPT=(PSS,API,SSCP,PIU[1]) and MODE=EXT. This creates a history of VTAM activity. At the time of performance degradation, stop VIT and take a console dump of VTAM. (See your operating system manuals for information on how to take a dump.) Load the dump and trace output for future reference.

9. **Report the problem.**

   Go to "Reporting the Problem to IBM" on page 112.

[1] If you are running an LU 6.2 application, include the APPC VIT option in this list.

Storage Problem Procedure (MVS)

This procedure focuses on storage problems that occur in the common service area (CSA) or private storage area. For general information on CSA storage and an explanation and description of storage pools, see Chapter 11, "Tuning VTAM for Your Environment" in the *VTAM Network Implementation Guide.*

Procedure Steps

The information in this section is taken from the following VTAM storage diagnosis information APARs:

II06752 An Introduction/Overview
II04548 Documentation Requirements
II07563 Private Storage Problems
II07564 CSA Storage Problems

1. **Check for common CSA and private storage messages.**

   Use the following messages to determine if the storage shortage is occurring in CSA or private storage. If the message is issued frequently or continuously, this indicates that a dump is needed to provide additional information.

   Table 8 lists the messages that are associated with CSA storage problems. See the description of the message in *VTAM Messages and Codes* for additional information.

Table 8. IST Messages Associated with CSA Storage Problems

Message number	Description
IST154I	Indicates that expansion failed for one of the fixed length buffer pools in ECSA subpool 231. The error code displayed in the message provides additional information.
IST561I	Indicates that SLOWPT has occurred in one of the fixed length buffer pools in ECSA subpool 231. • If this message only occurs occasionally, you might need to do some tuning in this area. See Chapter 11, "Tuning VTAM for Your Environment" in the *VTAM Network Implementation Guide* for additional information. • If the console is flooded with this message, you might have a CSA problem.
IST562I	Indicates that you have reached the limit you have set for CSA or CSA24. If this message repeats soon after you issue the MODIFY CSALIMIT or MODIFY VTAMOPTS command to increase the limit, further investigation of the contents of CSA will be required.
IST564I	Indicates that a GETMAIN failed for the CSA subpool specified in the message. The specified subpool might be the source of the CSA problem or the problem might be caused by another CSA subpool that is affecting the subpool displayed in the message. Further investigation into the contents of storage will be required.

Table 9 on page 106 lists the messages that are associated with private storage problems. See the description of the message in *VTAM Messages and Codes* for additional information.

Table 9. IST Messages Associated with Private Storage Problems

Message number	Description
IST563I	Indicates that the MAXPVT value has been reached. This value specifies how much private area subpool 229 storage VTAM can use within the address space of the application program displayed in the message. This indicates a problem with the application and not a VTAM problem.
IST565I	Indicates that a GETMAIN failed for the VTAM private area subpool displayed in the message. The specified subpool might be the source of the problem or the problem might be caused by another subpool that is affecting the subpool displayed in the message. Further investigation into the contents of storage will be required.
IST566I	This message is the same as IST563I except that MAXPVT was not specified on the APPL definition statement. This message does not indicate a VTAM problem.

2. **Request a full dump.**

   If storage-related messages are issued frequently or continuously, dump VTAM common and private storage areas. The dump can help you determine the location of the storage problem.

   - See "Formatting and Printing Dump Output (MVS) (VM)" on page 232 for information on the VTAM interactive problem control system (IPCS).
   - Several storage-related dump analysis tools are available. See "STORAGE" on page 320, "VTAM" on page 323, and "VTBUF" on page 327 for descriptions of these tools.
   - If external trace is active, see "Analyzing Storage" on page 417 for information about analyzing storage using the VIT analysis tool. See "Internal and External Trace Recording" on page 401 for additional information.

   Notes:

   a. The best dump for diagnosing VTAM storage problems is a full dump.

      If the dump is partial, examine the reason text of MVS message IEA911E to correct the problem so that you can obtain a full dump. The most common reason for a partial dump is that the dump dataset is not large enough. In this situation, calculate the DASD space requirements and reallocate the dump dataset.

      For a complete description of the required documentation for storage problems, see information APAR II04548.

   b. Although VTAM detects storage shortages in the common storage area, VTAM might not be causing the shortage because this area is shared by all address spaces.

3. **Use IBMLink to find additional problem determination information.**

   If you have access to IBMLink, take the following actions:

   - Review the appropriate VTAM storage diagnosis information APAR(s). See "Procedure Steps" on page 105 for a list of these APARs.
   - Use your error messages and dump to determine key words for searching IBMLink for additional information and known problems.

If you do not have access to IBMLink and need additional assistance, go to step 4 on page 107.

4. **If you need additional assistance, contact the IBM support center at 1-800-237-5511.**

Documentation Problem

Note: Before using this procedure, be sure that documentation is the problem. A VTAM problem might cause the documentation to appear wrong.

If the problem is documentation, use the procedure in Figure 15 to collect the following documentation:

- Incorrect information
- Form number of book.

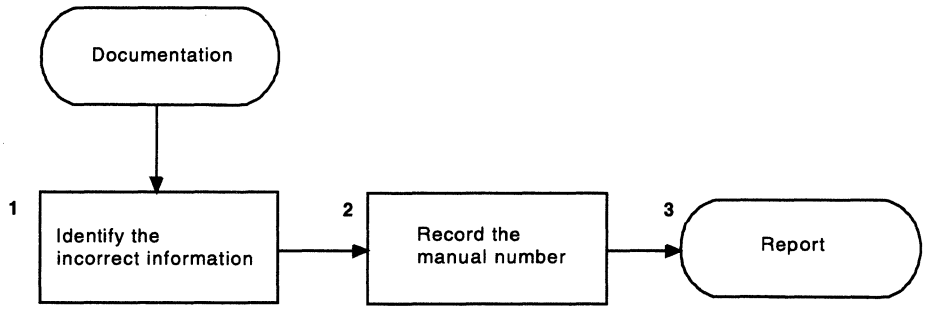


Figure 15. Overview of the Documentation Procedure

Following is a description of each step shown in Figure 15.

1. **Identify the incorrect information.**

   Add to your documentation list the name of the macro, operand, or procedure that is incorrectly defined or explained in the documentation (for example, "line trace.")

2. **Record the form number.**

   Add the form number of the VTAM book to your documentation list in the form *ccnnnnnnrr* (omit the dashes in the number; *rr* is the revision level). For example, report the form number of this book as LY43-0065.

3. **Report the problem.**

   Go to "Reporting the Problem to IBM" on page 112.

   Note: Report a documentation problem only when it causes a VTAM problem. For suggestions, comments, or questions about VTAM books, use the Reader's Comment Form at the back of the book.

Failing Module

Use this procedure if you have identified a failing VTAM module in one of the other procedures (abnormal end, message, or loop). Figure 16 shows an overview of the failing module procedure.

Use this procedure to get the following documentation:

- Module ID and PTF eye-catcher
- Caller of module
- VIT output.

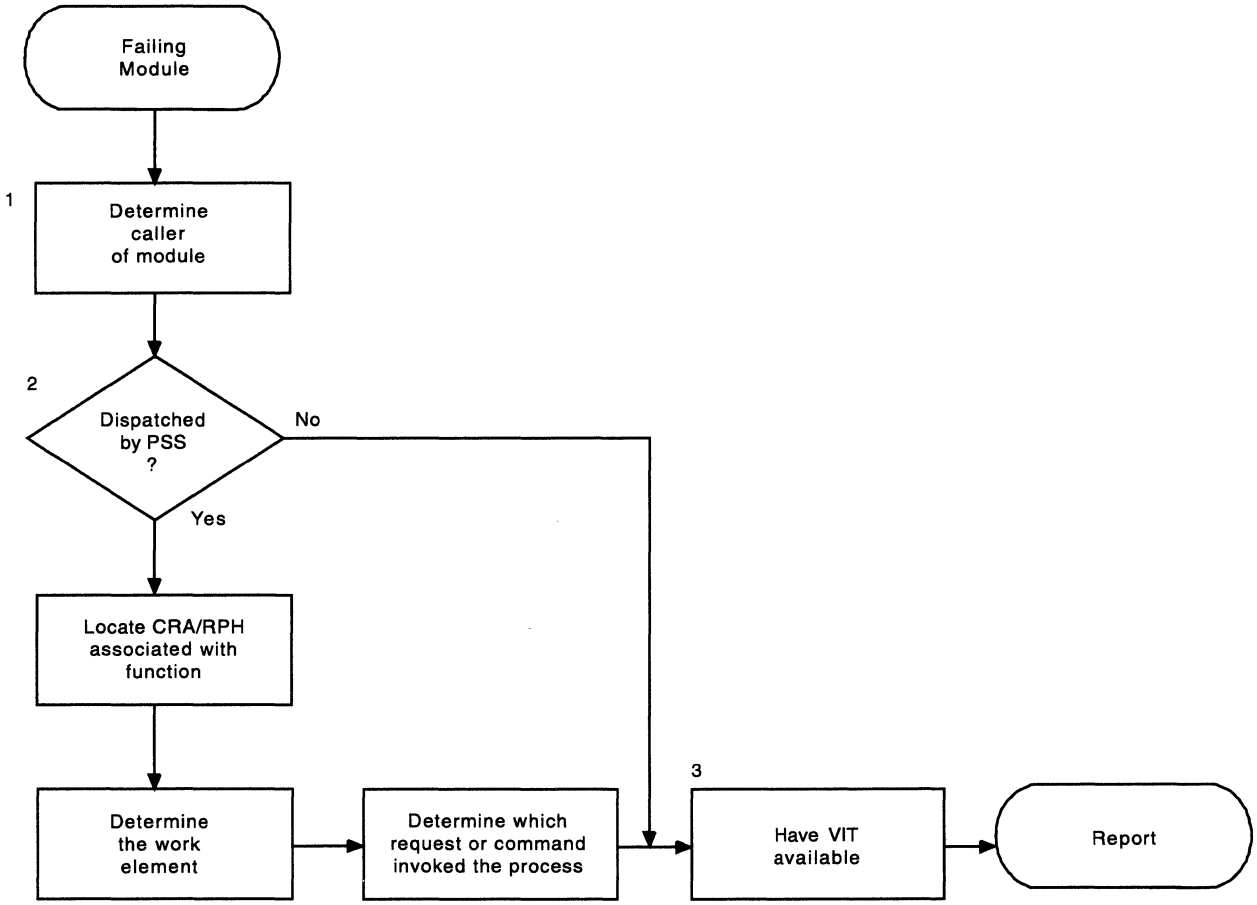


Figure 16. Overview of the Failing Module Procedure

Following is a description of each step shown in Figure 16.

1. **Determine the caller of the service routine.**

   The failure might have occurred in a VTAM service routine used for many purposes. Determine the caller of the service routine. Use the save area conventions in "Using Save Area Module Linkage Conventions—Subarea" on page 552, or if you know the save area address use the MVS ISTVSAVE CLIST.

2. **Examine the work element structure.**

   If your problem type is an abnormal end or loop, and the module is dispatched under control of VTAM PSS, find the CRA/RPH associated with the process. (See step 10 on page 79 in the wait procedure.) The RPHWEA field (at offset

X'1C') usually points to the work element associated with the process at the time it was dispatched. To identify the work element, see "Using Save Area Module Linkage Conventions—Subarea" on page 552.

From the work element, it might be possible to identify an SNA request/response type, an operator command, or an application program request that ultimately caused the process to receive control. Add this request or command name to your documentation list.

If the RPHWEA field does not point to the work element, continue with step 3.

3. **Have VIT information available.**

   Use the VIT options PSS and SMS to get more information about how the failing module received control or where the relevant control blocks are found. To obtain the address of the work element and the module name of the process entry point, use the last dispatch (DSP) entry for the failing process.

   Continue with "Reporting the Problem to IBM" on page 112.

Symptom String Structure

A symptom string is included in the dump for an abend, the dump for a program check, and the dump for a first failure support technology (FFST) probe point.

Message IST931I is issued for abend messages, and messages beginning with EPW are issued for FFST. Both the IST and the EPW messages contain the symptom text string. Figure 17 shows an example of a symptom text string for an abend dump. For information about FFST symptom strings, see "FFST Console" on page 224.

```
AB/S00C4  PIDS/5695-11701  LVLS/201  LVLS/91.025  RIDS/ISTCFCWM
RIDS/ISTAPCES#R  RIDS/ISTCFCR2#L  FLDS/PSW  ADRS/000006C4
VALU/HB0044770  FLDS/POWPSSQ  ADRS/00000F1C REGS/0C6A2  REGS/0A018
VALU/HE0044770  PIDS/568508501  PTFS/0000000  PRCS/00000010
```

Figure 17. Example of a Symptom Text String in Output

The meanings of the fields are given below:

AB Abend interrupt code, such as 0C4.

ADRS Offset into the failing VTAM module.

DEVS Device type and model related to the problem or generic device class.

FLDS Fields, control blocks, and DSECTs labels.

LVLS VTAM version and release level, or Julian date when the failing module was compiled. If LVLS appears twice in the symptom string, then it shows each of these values separately.

MS Program or device message.

OPCS Operation codes.

OVS Storage or core that was overlaid.

PCSS Statements, commands, JCL.

PIDS VTAM component ID.

PRCS Return code, status code, condition, feedback.

PTFS VTAM service level.

REGS The first two hexadecimal digits show the register number, and the next three hexadecimal digits show the displacement. The displacement value is the difference between the value of the PSW Instruction Address and the content of the register. Each REGS field is shown only if the value is less than the PSW, and if the difference is less than 4K.

If the REGS field has a value of X'FFFFF', then no register contents are less than the value of the PSW Instruction Address and within 4K of the PSW Instruction Address.

If the REGS field has a value of X'FE000', then the value of the PSW Instruction Address is less than decimal 512.

RIDS One of three kinds of modules:

- Recovery module, if followed by #R
- Load module, if followed by #L
- CSECT name of the failing VTAM module, if not followed by anything.

SIG System or device issued operator warning signal.

VALU Field value or overlay length.

WS Wait state.

Reporting the Problem to IBM

For non-VTAM problems, call your IBM branch office. For suspected VTAM problems, do either of the following steps:

- If you have access to IBMLink, search for known problems in this area. If no applicable matches are found, report the problem to IBM by using the electronic technical report (ETR) option on IBMLink.
- Contact the IBM Support Center at 1-800-237-5511.

  After asking for your account name and other customer identification, the service representative will ask for a brief description of the problem. Your documentation list should contain the answers to all questions related to the problem.

Chapter 3. Collecting Documentation for TSO/VTAM Problems (MVS)

This chapter shows you what documentation to collect for each type of common problem with the TSO/VTAM program. Use this chapter in conjunction with Chapter 2, "Collecting Documentation for Specific Types of Problems" on page 57.

Note: Most traces discussed in this chapter are described in Chapter 8, "Using Traces." The exceptions are SVC 93 and SVC 94 entries. See Table 80 on page 1165 to determine what book describes the SVC 93 and SVC 94 entries. For VTAM and TSO/VTAM command syntax, see *VTAM Operation*.

Initial TSO/VTAM Problem Analysis

To use this chapter, start below and follow the steps.

1. **Are you receiving one or more of the following messages?**
   - USS message 7 'LU-name UNABLE TO ESTABLISH SESSION — RU-name FAILED WITH SENSE sense', or similar USS message
   - USS message 10 (the user-defined logon message)
   - Message IKT029I or IKT028I at the operator's console
   - Message IKJ608I at the operator's console. See Table 80 on page 1165 to determine what book describes message IKJ608I.

   If so, go to "Logon Problems" on page 114.

   Otherwise, continue with step 2.
2. **Have you encountered one of the following abends?**
   - ABEND0AB
   - ABEND0AC
   - ABEND0AD
   - ABEND15D.

   If so, go to "TSO/VTAM Abends" on page 117.

   Otherwise, continue with step 3.
3. **Do you have a hung terminal?**
   - The terminal does not respond to any keys you press.
   - You must enter data from the terminal before processing will continue (in a situation where output is expected).

   If so, go to "Hung Terminal Problems" on page 119.

   Otherwise, continue with step 4.
4. **Are you having one or more of the following screen management problems?**
   - Data is in the wrong place on the screen.
   - Data stream errors occur (such as operation checks, commands are rejected, PROGxxx).

- Function errors occur (such as incorrect full-screen processing or incorrect line prompting in input mode of TSO EDIT).
- Data length is incorrect.
- Data content is incorrect.

If so, go to "Screen Management Problems" on page 121.

Otherwise, continue with step 5.

5. **Are you having one or more of the following screen size problems?**
   - The terminal does not operate in the expected screen size after log on.
   - The screen is not always the expected size during a TSO session.

   If so, go to "Screen Size Problems" on page 126.

   Otherwise, continue with step 6.
6. **Are you having one or more of the following performance problems?**
   - Slow response time
   - An increase in the number of detected waits
   - An increase in the number of swap-outs.

   If so, go to "Performance Problems" on page 129.

If your problem is not listed in the steps above, it is probably not a TSO/VTAM problem. Go back to Chapter 2, "Collecting Documentation for Specific Types of Problems" on page 57 and look for a more likely problem symptom. If you cannot find a more likely symptom, go to "Reporting the Problem to IBM" on page 112.

Logon Problems

This section provides documentation requirements and diagnosis procedures for logon problems.

The recommended documentation is:

- VTAM full buffer contents trace.

  To see the data in the buffer contents trace, set CONFTXT=NO in the TSOKEY00 member of SYS1.PARMLIB before starting TSO/VTAM.
- VTAM internal trace with MODE=EXT and OPTION=(API, MSG, NRM, PIU, SSCP, PSS).

  The VTAM internal trace may not be required. Review the diagnosis procedure for your problem to see if it is required.

  Note: API, MSG, NRM, PIU, and SSCP are always running internally.

1. **Did your first logon using USS commands fail?**

   If so, continue with the next step.

   Otherwise, go to step 5 on page 115.
2. Under "Unformatted System Services (USS) Tables" in the *VTAM Resource Definition Reference*, **review the process for setting up the USS table and using USS commands**. Check for the following errors:

- Is your logon command syntax incorrect?

  If so, try to log on using the correct command syntax.

  Otherwise, continue with the next step.

- Is the logmode name incorrect?

  If the logmode name is specified incorrectly, or if a default logmode entry that is inappropriate for the device type is used, you will get USSMSG7. Look up the accompanying sense code in Chapter 15, "Sense Codes" in *VTAM Messages and Codes* and correct the logmode name.

3. **Can you log on to TSO without using USS commands?**

   If you cannot log on at all, go to step 4.

   If you can log on, start the VTAM buffer contents trace and log on again. Look at the trace output to see what session parameters are contained in the BIND, and compare those parameters to the ones in your logmode table.

   If the session parameters in your logmode table are incorrect, make the necessary corrections. Also, make sure the DLOGMOD operand specifies the correct logmode table entry. (For more information on defining TSO/VTAM session parameters, see "Defining TSO/VTAM Session Parameters" in the *VTAM Network Implementation Guide*.)

   If you still cannot identify the problem, go to "Reporting the Problem to IBM" on page 112.

4. **Are you unable to log on at all?**

   - If this is your first logon attempt from the device as well as your first logon attempt using USS commands, go to step 5.
   - If this is not your first logon attempt from the device, go to "Reporting the Problem to IBM" on page 112.

5. **Did your first logon from a particular device fail?**

   If so, continue with the next step.

   Otherwise, go to step 7 on page 116.

6. **Check for an error in the terminal definition statement, the logmode table, or the MODEENT macro.**

   (These are described in "TERMINAL" and "Logon Mode Table" in the *VTAM Resource Definition Reference*.)

   a. If you receive message IKT024I with return code X'210000' or X'220000', the BIND has been rejected. The following steps should help you find the portion of the BIND that is not valid:

      1) Locate the BINFM in the BIND. BINFM must be X'02' or X'03'. (For more information on coding the BIND, see "BIND Area Format and DSECT" in *VTAM Programming*.)

      2) If a PSERVIC is coded, see "PSERVIC" in the *VTAM Resource Definition Reference* to make sure all fields are coded correctly.

   b. Check to see if the DLOGMOD name on the terminal definition statement is a valid logmode table entry.

If it does not match an entry in the logmode table, the first entry in the logmode table is used as the default. The parameters on the default logmode table entry may not be appropriate for your device type, and as a result, the wrong BIND image may be passed to the logon exit and a CLSDST PASS failure may occur.

Note: You can see this failure in the VTAM internal trace using the API option. For more information on the VTAM internal trace, see Chapter 9, "Using the VTAM Internal Trace (VIT)."

c. Check the logmode table entry to see if the parameters are specified correctly for your device type.

   If not, the wrong BIND image may be passed to the logon exit and a CLSDST PASS failure may occur.

d. Check to see if the MODEENT macro is defined correctly.

   If it is not, the terminal may reject the BIND, or the terminal may indicate to the logon exit that the terminal is not supported by TSO/VTAM.

e. If you have not identified the problem, and if users can log on to TSO from other terminals, start the VTAM buffer contents trace and check the BIND.

f. If logons fail for all terminals, or if the BIND in the buffer contents trace is what you expected, run the VTAM internal trace with options API, PIU, SSCP, and MSG.

g. If you still have not resolved the problem, go to "Reporting the Problem to IBM" on page 112.

7. **Did previous logons succeed, but now you cannot log on?**

   If so, continue with the next step.

8. **Did you get message IKJ608I?**

   If not, go to step 12 on page 117.

   If so, continue with the next step.

9. **Is message IKJ608I followed by messages IST804I, IST400I, and IST805I?**

   If so, the CLSDST PASS or OPEN ACB may have failed. Continue with the next step.

10. **Is this a cross-domain logon?**

    If not, go to step 11.

    If this is a cross-domain logon, you may have a VTAM definition problem. For more information on defining TSO/VTAM and logical units that must access TSO/VTAM in a cross-domain environment, see "Defining the TCAS Application to VTAM" in the *VTAM Network Implementation Guide*.

11. **Is there a TSO/VTAM APPLID that is not valid?** (For more information on defining APPLIDs, see "Defining an Interpret Table for Compatible Logons" in the *VTAM Network Implementation Guide*.)

    If so, correct the APPLID. Once you have corrected the APPLID, you must deactivate the application and reactivate it to cause VTAM to reload the correct APPLID. This should fix the problem.

    If not, continue with step 12 on page 117.

12. **Did you get message IKT111I?**

    If not, go to step 13.

    If so, check the message text to determine the reason for the logon failure. See *VTAM Messages and Codes* for additional diagnostic information for particular messages.

13. **Is an ABEND0AB with return code X'0105' or X'0203' associated with the logon attempt?**

    If not, go to step 14.

    If so, check the LOGREC entry for the additional information shown in Table 10.

Table 10. ABEND0AB Information in LOGREC

Reason Code	Register	Contents
X'0105'	6	RPLRTNCD, RPLFDB2, and RPLDAF
	7	RPLFDBK2 (the word of sense)
X'0201'	8	ACBERFLG (for OPEN ACB failure)
	9, 10	TVWA ACB Name (TSOnnnn)
X'0202'	8	ACBERFLG (for OPEN ACB failure)
	9, 10	TVWA ACB Name (TSOnnnn)
X'0203'	5	ACBERFLG (for CLOSE ACB failure)
	6, 7	TVWA ACB Name (TSOnnnn)

Then go to "Reporting the Problem to IBM" on page 112.

14. **If none of the previous situations apply**, start the VTAM buffer contents trace and the VTAM internal trace, and trace the logon attempt. Then go to "Reporting the Problem to IBM" on page 112.

TSO/VTAM Abends

TSO/VTAM issues several unique abends. This section briefly describes the causes and documentation requirements for each one. Use the information provided here and in Chapter 11, "Abend Codes (MVS) (VM)" in *VTAM Messages and Codes* to try to resolve the problem. If you are not able to do so, go to "Reporting the Problem to IBM" on page 112.

ABEND0AB

ABEND0AB occurs when a VTIOC module issues a VTAM macroinstruction that fails. Depending on the values of the RPLRTNCD and RPLFDB2 fields, the macro may be retried. If the retry fails, ABEND0AB is issued.

Table 11 (Page 1 of 2). ABEND0AB Information in a Dump of SDWA

Offset	Length (bytes)	Description
X'280'	8	Terminal name

Table 11 (Page 2 of 2). ABEND0AB Information in a Dump of SDWA

Offset	Length (bytes)	Description
X'289'	1	RPL request type: X'22' = SEND; X'23' = RECEIVE
X'28A'	1	RPLRTNCD
X'28B'	1	RPLFDB2
X'28C'	4	RPLFDBK2 (Sense code)

If this happens during the execution of a SEND or RECEIVE, the session is placed in reconnect status.

The recommended documentation is:

- Contents of register 15.

  This contains the reason code, which is explained in Chapter 11, "Abend Codes (MVS) (VM)" in *VTAM Messages and Codes*.

- Message text for message IKT116I.
- The software LOGREC entry.

  If you have a LOGREC entry, look at an unformatted dump of the SDWA. Table 11 on page 117 describes the pertinent data you should look for in the dump.

- The dump that is created automatically for this abend.

  For abends associated with I/O errors, a dump is not generated automatically unless the RCFBDUMP parameter of the TSOKEY00 member of SYS1.PARMLIB is set for it.

- For errors that occur during session initialization or termination, run the VTAM internal trace with the options API, PIU, MSG, SSCP, and PSS.

  Note: The options API, MSG, PIU, and SSCP are always running internally, but you may want to run the VIT with MODE=EXT to be certain that you get the expected output.

ABEND0AC

ABEND0AC occurs when an error halts TCAS processing.

The recommended documentation is:

- Contents of register 15.

  This contains the reason code, which is explained in Chapter 11, "Abend Codes (MVS) (VM)" in *VTAM Messages and Codes*.

- TWARSON (IKTCASWA + X'02').

  This also contains a reason code. See Chapter 11, "Abend Codes (MVS) (VM)" in *VTAM Messages and Codes*, or see Table 80 on page 1165 to determine what book contains the MVS system codes.

ABEND0AD

ABEND0AD occurs when the TSO/VTAM queue manager has a problem manipulating storage for the input and output queues.

The recommended documentation is:

- Contents of register 15.

  This contains the reason code, which is explained in Chapter 11, "Abend Codes (MVS) (VM)" in *VTAM Messages and Codes*.

- The dump that is created automatically for this abend.

ABEND15D

ABEND15D occurs when the issuer of a TGET, TPUT, or TPG macro passes a data area that is not valid to the SVC 93 modules. A TPUT or TPG request requires read access to the area, and a TGET requires write access. An ABEND0C4 occurs when IKTVTPUT or IKTVTGET tries to validate the data areas passed from the application program, and IKT93EST changes the ABEND0C4 to an ABEND15D.

The recommended documentation is:

- SLIP dump of the ABEND0C4. To determine what book describes the SLIP dump, see Table 80 on page 1165.
- GTF trace of SVC 93 entries.

Note: Either of these will show the address that is not valid.

Hung Terminal Problems

This section provides documentation requirements and diagnosis procedures for problems with hung terminals. Use this information if this problem occurs while you are using TSO/VTAM. This procedure helps you determine when the hang occurred and what was happening at that time.

1. **If the problem occurs during logon or logoff**, get the recommended documentation and go to "Reporting the Problem to IBM" on page 112.

   The recommended documentation is:

   - VTAM full buffer contents trace.

     To see the data in the buffer contents trace, set CONFTXT=NO in the TSOKEY00 member of SYS1.PARMLIB before starting TSO/VTAM.

   - GTF trace of SVC 93 and SVC 94 entries. See Table 80 on page 1165 to determine what book describes the SVC 93 and SVC 94 entries.
   - TGET/TPUT trace.

     The TGET/TPUT trace creates trace entries for all TGET/TPUT/TPG data except address space ID TPUTs.

     You can get the TGET/TPUT trace by issuing the MODIFY TRACE command with TYPE=TSO. The MODIFY command is described in "MODIFY TRACE Command" in *VTAM Operation*.

   - Dump of the nucleus, CSA, and user's address space.

If you think you might not be able to re-create the problem, take a dump *before* you try to clear the hang.

If input (such as ATTN, RESET, or ENTER) clears the hang, both a dump and traces may be necessary. You may want to start the traces, take a dump with the terminal hung, clear the hang, and then stop the traces.

- Collect additional general information:
  - Try to determine if the error is related to a certain type of hardware or a certain protocol (SNA or non-SNA).
  - Try to determine if the hang is related to a particular application program or type of application program (full-screen, graphics, and so on).

    If so, do other similar types of applications also hang?

2. **Was the last data that was sent from the application program to VTAM sent to the terminal before the hang occurred?**

   To determine this, look for a VTAM buffer contents trace entry that corresponds to the last TPUT trace entry.

   If you see these corresponding entries, the output was sent to the terminal.

3. **If the keyboard locked after data was sent to the terminal**, check the outbound buffer contents trace entry for:
   - Bracketing indicators in RH byte 2
   - Change direction indicator in RH byte 2
   - Write control character to unlock the keyboard in byte 2 of the output request unit

   If an end bracket was sent, the keyboard should be available.

   If a change direction was sent and the keyboard has been unlocked, the keyboard should be available.

   If a TGET is issued after a full-screen TPUT, TSO/VTAM should unlock the keyboard.

4. **If the keyboard has not been unlocked**, see what kind of TPUT was issued last. (The option flag bytes in the TPUT entry show what kind of TPUT it is.)
   - For a NOEDIT or TPG TPUT, TSO/VTAM will not unlock the keyboard. The application program is supposed to send a write control character to unlock the keyboard.
   - For other TPUT options, is a TGET outstanding?

     If a TGET is outstanding, TSO/VTAM should unlock the keyboard.

     If no TGETs are outstanding, contact the group responsible for the application program.

5. **If the last activity prior to the hang was input from the terminal**, was the data passed to the application program? (If it was, the TGET trace entry corresponds to the inbound VTAM buffer contents trace entry.)
   - If so, did the application program ever issue another TPUT?

If the application program never issued another TPUT, contact the group responsible for the application program.

- If data has been received by TSO/VTAM, but it has not been sent to the terminal, go to "Reporting the Problem to IBM" on page 112.

6. **If you have not identified the problem**, go to "Reporting the Problem to IBM" on page 112.

Incorrect Output Problems

Two main types of incorrect output problems are discussed in this section: "Screen Management Problems" and "Screen Size Problems" on page 126. Screen management problems involve mode errors, exception responses, and problems with the data on the screen. Screen size problems involve an incorrect or unexpected screen size, either in a particular mode or all the time. Choose the one that is most like your symptoms, and follow the procedure for that problem.

Screen Management Problems

This section provides the documentation requirements and diagnostic procedures for problems displaying data on the screen. This section deals with five types of screen management problems:

- Function error (incorrect screen management for mode).
- Exception responses.
- Extra or missing data.
- Data is not placed correctly on the screen.
- Data appears to be translated incorrectly.

Choose the one that most closely matches your symptoms and follow the procedure for that problem.

Note: Problems with incorrect screen sizes are addressed in "Screen Size Problems" on page 126.

The recommended documentation is:

- VTAM full buffer contents trace.

  To see the data in the buffer contents trace, set CONFTXT=NO in the TSOKEY00 member of SYS1.PARMLIB before starting TSO/VTAM.

- GTF trace of SVC 93 and SVC 94 entries. See Table 80 on page 1165 to determine what book describes the SVC 93 and SVC 94 entries.

- TGET/TPUT trace.

  The TGET/TPUT trace creates trace entries for all TGET/TPUT/TPG data except address space ID TPUTs.

  You can get the TGET/TPUT trace by issuing the MODIFY TRACE command with TYPE=TSO. The MODIFY command is described in "MODIFY TRACE Command" in *VTAM Operation*.

- Dump of CSA storage and the user's address space.

  This is only required for address space ID TPUT errors.

- May require a full PIU trace or CCW trace with data option.

Function Error

If the screen does not function properly for the current mode of operation, do the following steps.

1. **Review the SVC 94 entries to determine the mode.** The STFSMODE and STLINENO macros set full-screen mode on and off.

   - Full-screen mode.

     In full-screen mode, the application handles screen management.

     If the problem relates to full-screen processing, review the information on full-screen mode. See Table 80 on page 1165 to determine what book describes full-screen mode. If this does not describe the full-screen processing that you are experiencing, note the differences.

   - Line mode.

     In line mode, TSO/VTAM handles screen management.

     In line mode, the data generated by the application program is placed line by line down the screen. READY appears on the line below the data, and the cursor appears on the line below that. When the screen is full, TSO/VTAM sends a page prompt to the screen. When you press the ENTER key, TSO/VTAM clears the screen and sends any remaining data to the screen.

     If you enter data on the next to the last line of the screen, no page prompt is sent. Instead, TSO/VTAM clears the screen and reshows the data (or command) at the top of the screen.

     If this does not describe the line mode processing that you are experiencing, note the differences.

2. **Report to the group (TSO/VTAM or the application program) that appears to be responsible for the incorrect screen management.**

   - If you have access to IBMLink, search for known problems in this area. If no applicable matches are found, report the problem to IBM by using the electronic technical report (ETR) option on IBMLink.
   - If you do not have access to IBMLink, call the IBM Support Center at 1-800-237-5511.

Exception Responses

Follow these steps for exception responses.

1. **Does the error indicator reflect an error in the RU portion of the PIU (sense = 1003 or 1005)?**

   If so, continue with this step.

   Otherwise, go to step 2 on page 123.

   a. See what kind of TPUT was issued. (This is shown in the flag byte of the TPUT trace entry.)

   b. For a NOEDIT TPUT, TSO/VTAM should not change the data provided by the application program. Compare the data in the TPUT trace, which starts at X'2C' into the trace record, with the VTAM buffer contents trace.

If TSO/VTAM has not changed the data, contact the group responsible for the application program that issued the TPUT.

c. For a full-screen TPUT, was the data that is causing the error generated by the application program? (Data generated by the application program is present in the TPUT trace entry.)

If so, contact the group responsible for the application program that issued the TPUT.

If not, go to "Reporting the Problem to IBM" on page 112.

2. **Look at the error sense code.**

If it is X'800A', the PIU is too long. This is probably a definition error. In this case:

a. Look in the TH portion of the buffer contents trace entry to find out the length of the PIU that caused the error.

b. See how MAXDATA is defined on the PCCU definition statement in the NCP definition deck. The MAXDATA value that you code should be as large as the largest PIU that is sent to the terminal by an application program.

Note: If you increase the value of MAXDATA, this value should not exceed the product of the MAXBFRU and UNITSZ operands. See "HOST Definition Statement" in *VTAM Network Implementation Guide* for more information on defining the MAXDATA, MAXBFRU, and UNITSZ operands.

If you have more than one PCCU definition statement, check to see that the right one is being used.

3. **If there are other error indicators**, get the documentation shown in step 5 on page 98 and continue with that procedure.

4. **If you have not resolved the problem**, go to "Reporting the Problem to IBM" on page 112.

Extra or Missing Data

Follow these steps if you have an extra or missing data condition.

1. **Is the problem with input (data received by the application)?**

a. Is too much data being passed to the application?

- If previous input is added to the end of the current input, the modified data tags may have been set improperly in previous TPUTs. Go to "Reporting the Problem to IBM" on page 112.
- If the backspace key, character delete key, or line delete key is not functioning properly, look for SVC 94 entries that may have changed the default in the STCC macro.

  If this does not account for the problem, get a dump of the nucleus, the CSA, and the user's address space. Then go to "Reporting the Problem to IBM" on page 112.

b. Is insufficient data being sent to the application program?

- Find out which TGET option was used. TSO/VTAM edits data sent from the terminal before it passes it to the application program. The type of editing that TSO/VTAM does depends on the TGET option.

Certain characters may be deleted, such as control characters, aid characters, and set buffer address (SBA) sequences. Look at the flag bytes in the TGET trace entry to see which TGET option was specified. See Table 80 on page 1165 to determine what book describes TGET options.

- Find out if any user edit exits are involved. User edit exits are listed in Chapter 4, "Writing TSO/VTAM Installation-Wide Exit Routines (MVS)" in *VTAM Customization*.
- **If you have not resolved the problem**, go to "Reporting the Problem to IBM" on page 112.

2. **Is the problem with output (data sent by the application)?**

   a. Was extra data sent to the screen?

      TSO/VTAM should not generate any printable data. Compare the data portion of the TGET/TPUT trace with the data in the VTAM buffer contents trace.

      If you see printable data in the buffer contents trace that is not in the TGET/TPUT trace, go to "Reporting the Problem to IBM" on page 112.

      If you see the same data in both traces, contact the group responsible for the application program that issued the TPUT.

   b. Was data from the application lost?

      TSO/VTAM does not generally delete data sent by the application program unless it is doing reshow processing. In reshow processing, TSO/VTAM deletes the full-screen TPUT and sends a reshow character (X'6E') to the full-screen application program. This is shown in the TGET trace entry. Compare the data portion of the TPUT trace entry with the data in the VTAM buffer contents trace entry to see if any data has been deleted. Determine whether this is a TSO/VTAM problem or an application program error and contact the appropriate group.

      See Table 80 on page 1165 to determine what book contains more information on reshow processing.

3. **If you have not resolved the problem**, go to "Reporting the Problem to IBM" on page 112.

Data Is Misplaced on the Screen or Page

Follow these steps if data is misplaced on your screen or page.

1. **Does data wrap around the screen?** (Wrapping means that data fills the line and splits inappropriately between lines.)

   If it does, continue with this step.

   Otherwise, go to step 2 on page 125.

   a. Find out which TPUT option was issued. Bytes X'12' and X'13' of the TPUT trace entry contain the option flags.

   b. For a NOEDIT TPUT, TSO/VTAM does not edit data, and therefore does not change any SBA sequences that may be issued by the application program. Contact the group responsible for the application program that issued the TPUT.

c. For a full-screen TPUT, TSO/VTAM does not generate SBA sequences to place the data on the screen.

   If the symptom is incorrect screen size, go to "Screen Size Problems" on page 126. Otherwise, contact the group responsible for the application program that issued the TPUT.

d. If the data was sent without a full-screen or NOEDIT option, go to "Reporting the Problem to IBM" on page 112.

2. **If the problem is not on a display terminal**, go to "Reporting the Problem to IBM" on page 112.
3. **Are SBA sequences correct?** Check the type of TPUT that was issued. Bytes X'12' and X'13' of the TPUT trace entry contain the option flags. For full-screen and NOEDIT TPUTs, the application program usually generates the SBA sequences that determine where data is placed on the screen.
4. **Is the buffer address incorrect?**
   - If the buffer address is not valid for the screen size, contact the group responsible for the application program.
   - If the buffer address is valid for the terminal in its present screen size, go to "Reporting the Problem to IBM" on page 112.
5. **If this is the first non-full-screen TPUT following a full-screen TPUT**, look for an SVC 94 trace entry for STLINENO. This macro may be issued by a full-screen application to indicate which line the next non-full-screen data should appear on.

   If the data was placed by the STLINENO macro, contact the support group for the application program that issued the macro.
6. **If you have not resolved the problem**, go to "Reporting the Problem to IBM" on page 112.

Data Appears To Be Translated Incorrectly

Incorrect output is the main symptom of this problem. Sometimes the incorrect output is colons. The problem is probably related to the TPUT option that was specified.

1. **Look at the option flag bytes** (X'12' and X'13') in the TPUT trace entry to determine what options were used.
2. **Determine what editing occurs for each type of TPUT option**. See Table 80 on page 1165 to determine what book describes editing done by TPUT options.
3. **If the incorrect output consists of colons**, determine what data from the TPUT trace entry is being edited into the printable character X'7A', a colon.
4. **See if TSO/VTAM is editing correctly**.

   Note: Many applications use TPUT options that do extensive editing and translation. This allows many different hardware devices to communicate with the application program without causing I/O errors. You may need to write your own user edit exits to make sure that all characters that are valid for your terminals appear on the screen, especially if you are using type 1 logical unit devices.

5. **If the editing does not appear to conform to the options specified**, check for user edit exits or translation tables. (Chapter 4, "Writing TSO/VTAM Installation-Wide Exit Routines (MVS)" in *VTAM Customization* has a list of these exits.)

   If you have exits or translation tables, verify that they are not causing the problem.

6. **Did the application program send incorrect data to VTAM**? The data is shown in the data portion of the TPUT trace entry.

   If so, the problem is in the application program.

7. **If you have not resolved the problem**, go to "Reporting the Problem to IBM" on page 112.

Screen Size Problems

This section provides documentation requirements and diagnostic procedures for screen size problems. Two major types of errors occur:

- The screen never operates in the expected size.
- The screen is not always the expected size when you change modes.

Choose the one that most closely matches your symptoms and follow the procedure for that problem.

Recommended documentation includes the following list.

Note: All of the traces may not be required. Read the diagnostic procedure before you get them.

- VTAM buffer contents trace.

  To see the data in the buffer contents trace, set CONFTXT=NO in the TSOKEY00 member of SYS1.PARMLIB before starting TSO/VTAM.

- GTF trace of SVC 93 and SVC 94 entries. See Table 80 on page 1165 to determine what book describes the SVC 93 and SVC 94 entries.

- TGET/TPUT trace.

  The TGET/TPUT trace records all TGET/TPUT/TPG data except address space ID TPUTs.

  You can get the TGET/TPUT trace by issuing the MODIFY TRACE command with TYPE=TSO. The MODIFY command is described in "MODIFY TRACE Command" in *VTAM Operation*.

- May require a full PIU trace or CCW trace with data option.

Screen Is Never the Expected Size

This is probably a definition problem.

1. **If you are using a USS command to log on**, try logging on without it.

   If this corrects the problem, review the use of the USS command in "Unformatted System Services (USS) Tables" in the *VTAM Resource Definition Reference* and check the following:

   - Is the terminal a non-SNA 3270?

If it is, does the USS command include a USSPARM macro for the logmode?

If so, VTAM ignores the logmode name from the terminal definition statements and uses its own default BIND image instead.

- Does the USS command establish a default logmode name?

  If so, the default name overrides the name in the terminal definition statement.

2. **If this is not a USS command problem**, check to see if the PSERVIC operand of the MODEENT macro is coded correctly. "PSERVIC" in the *VTAM Resource Definition Reference* explains how to do this.
   - Are primary and alternate sizes coded correctly?
   - For screen switching, is BINPRESZ coded correctly as X'7F'?
3. **If you have not identified the problem**, look at the BIND that is sent. This is shown in a VTAM buffer contents trace of the logon.
   - If the BIND image is not what you expected, check the LU definition statement for an incorrect MODETAB or DLOGMOD parameter.
   - If no logmode table or DLOGMOD operand is specified, no PSERVIC is passed to the TSO/VTAM logon exit routine. In this case, TSO/VTAM issues an INQUIRE DEVCHAR macro and VTAM indicates that the terminal is a logical unit. TSO/VTAM then uses the SCRSIZE operand found in TSOKEY00. The default value for SCRSIZE is 480 (12 rows and 40 columns).
4. **If you have not resolved the problem**, go to "Reporting the Problem to IBM" on page 112.

The Screen Is Not the Expected Size for the Mode

Full-screen mode. The application program controls screen management in full-screen mode. The primary (small) screen size is considered by TSO/VTAM as the default size. The application program can control screen size by sending write commands in TPUTs that it issues. The write commands are X'F5', erase write, and X'7E', erase write alternate. The application program issues X'F5', erase write, to set the primary screen size, or X'7E', erase write alternate, to set the alternate (large) screen size.

If neither command is issued, the screen remains the size it is when the application program enters full-screen mode.

Line mode. TSO/VTAM controls screen management in line mode. It generally uses the large (alternate) screen size when processing TPUTs in line mode. You can use the TSO TERMINAL command STSIZE macro during a session to change the screen size for non-full-screen processing.

Using output from the VTAM full buffer contents trace, the TGET/TPUT trace for TSO/VTAM, and the GTF trace of the SVC 93 and SVC 94 entries, try to locate the source of the problem.

1. **Check the SVC 94 trace entries** to see whether the processing is in full-screen mode or line mode. The STFSMODE and STLINENO macros set these modes on and off.

If the processing is in line mode, go to step 6 on page 128. For full-screen mode, continue with step 2 on page 128.

2. **Note if the incorrect screen size is related to entering or exiting full-screen mode.**
3. **Locate the TPUT trace entry for the data that appears on the screen when the screen is the wrong size.** Determine the TPUT options for this TPUT and the one that precedes it by looking at the option flag bytes.

   If either is a full-screen TPUT, look at the first data byte.

   If the first data byte is an escape character (X'27'), the write command that follows has been specified by the application program. This write command should determine the screen size.

   If the write command is different in the buffer contents trace, go to "Reporting the Problem to IBM" on page 112.
4. **If the first data byte is not an escape character**, is a write command (X'F1', X'7E', or X'F5') provided?

   If one of these write commands exists, continue with this step.

   If not, go to the next step.

   If this write command is different in the VTAM buffer contents trace, go to "Reporting the Problem to IBM" on page 112.

   If the same write command appears in both traces, contact the group responsible for the application program.
5. **If a write command is not provided in the TPUT data**, and processing is in full-screen mode, does the write command generated by TSO/VTAM set the same screen size as the last write command provided by the full-screen application program? To determine this, compare the write command in the buffer contents trace entry with the last one provided in a TPUT trace entry. The write command is located in the data portion of the TPUT trace, at X'2C' into the entry.

   If the write commands are the same, go to "Reporting the Problem to IBM" on page 112.

   If the write commands are different, contact the group responsible for the application program.
6. **If processing is not in full-screen mode**, did the STSIZE macro set the screen size? To determine this, look in the SVC 94 trace entries. An entry code of X'0A' in the high order byte of register zero indicates that the STSIZE macro set the screen size.

   If the screen size was set by the application program, contact the group responsible for the application program.
7. **If you have not resolved the problem**, go to "Reporting the Problem to IBM" on page 112.

Performance Problems

This section provides documentation requirements and diagnosis procedures for performance problems. Use this section along with the information in "Performance Problem" on page 101. In addition to the documentation required in "Performance Problem," you will need a GTF trace of SVC 93 and SVC 94 entries. See Table 80 on page 1165 to determine what book describes the SVC 93 and SVC 94 entries.

1. **Are wait and hold options slowing response time?**

   The application program may be issuing TPUTs and TGETs with the wait or hold options. The wait option indicates that the application program should not regain control until output data has been placed on the output queue (TPUT) or input is available (TGET). A TPUT with a hold option indicates that control should not be returned to the application until the data has reached the terminal. These options may be necessary for screen management, but they prolong response time and increase the number of times the address space is swapped.

2. **Is an external system resource slowing response time?**

   If all users must access the same resource, such as a system catalog, performance deteriorates. This problem is especially severe when exclusive ENQs are used to control access to the resource. To improve performance, redistribute resources.

3. **Are high and low buffer extents set at inappropriate values?**

   If the high and low buffer extents are too close together, output wait conditions occur. Buffer extents are specified in TSOKEY00, a member of SYS1.PARMLIB. Tune the values of the high and low buffer extents to get optimum performance. Tuning is explained in Chapter 11, "Tuning VTAM for Your Environment" in the *VTAM Network Implementation Guide*.

4. **Are APPL definition statements coded correctly?**

   Code the AUTH=NVPACE operand on all APPL definition statements for TSO/VTAM.

   If you do not set NVPACE, VTAM indicates that it has already received input data, instead of queuing the response until it receives the input data. Also, the swap count is incremented by two every time the ENTER key is pressed.

5. **Are pacing values set correctly for local SNA terminals?**

   VTAM ignores the NVPACE operand for sessions with logical units in a local major node. Therefore, you must set non-zero pacing values for these logical units.

6. **Is the MVS performance group specified correctly?**

   Set the application program's performance group five to ten percent lower than VTAM's performance group. To see the application program's performance group specification, look at the dispatching priority in the task's TCB.

7. **If you have not resolved the performance problem**, go to "Performance Problem" on page 101.

Chapter 4. Collecting Documentation for VSCS Problems (VM)

This chapter defines symptoms of VSCS problems, groups the symptoms into VSCS problem types, and gives step-by-step procedures for generating the appropriate documentation to help resolve the problems. If you have to recreate a problem, follow the steps under the problem description that most closely matches your problem.

Documentation You Should Collect for All VSCS Problems

Always run VSCS with one or more trace options active. The more trace options that are running when a problem occurs, the easier it is for the VTAM change team to diagnose and correct a problem. The VSCS console communication services (CCS) trace option and the VTAM exit trace option default to ON. The CCS option traces data transferred, using IUCV, between VSCS and the CCS subcomponent of CP. The VTAM option traces activity in the VTAM services component of VSCS. The CCS and VTAM options do not significantly impact system performance.

To prevent the VSCS global trace table from wrapping, run traces externally, increase the size of the global trace table, or use the VSCS ILU trace option. For information on setting the size of the VSCS global trace table, refer to the TRASIZE start option of the DTIGEN macroinstruction in Chapter 5, "VM/SNA Console Support (VSCS)" in the *VTAM Resource Definition Reference*.

After you produce a VSCS dump, use a dump-formatting routine to format the dump output. Control blocks are easier to locate in formatted output. See "Dump Formatting Routine" on page 453 for information on formatting VSCS dump output.

Have the following documentation available when you report a problem to the IBM Support Center:

- VSCS link map (file VSCS LKEDIT on the VTAM RUN disk)
- VSCS maintenance list (file with filetype VMFSVLOG on the VTAM RUN disk)
- Console log(s) for the VTAM and VSCS virtual machines.

"General Procedures to Diagnose Hung Logical Unit Problems" and "Determining Device Characteristics" on page 135 explain how you can use the VSCS DISPLAY ID=*luname* command to determine the status of a particular logical unit. In some cases you can also determine where the problem is most likely to be, or which products are involved.

General Procedures to Diagnose Hung Logical Unit Problems

If the problem is for one or more logical units, but not for all logical units, use the following commands to determine additional information as to where the problem is, from an external viewpoint:

1. Issue a VSCS DISPLAY ID=*netid.luname* command. Use the display output to follow the steps below.

a. If the logical unit is not found (message DTIS69I is issued), then VSCS has successfully closed the LU by issuing a VTAM CLSDST. The LU-LU session has ended.

b. If VSCS can locate the logical unit, it displays message DTIS61I. This does not indicate that VSCS has an LU-LU session with the logical unit. If the logical unit has dialed another guest virtual machine using CP DIAL, "DIALED" is displayed.

c. Is message DTIS66I displayed? If so:

   - Note the current presentation services (PS) state of the LU.

     If the current state is one of the following—CIP, CLF, IIP, INVL, LGF, LIP, PRG or SVR—then the logical unit is in an internal transitory state and is waiting for another process to complete before continuing. These states should be extremely short; if they are displayed by two or more DISPLAY commands, the logical unit is hung.

     If the current state is LUP, a VM logo should be displayed at the LU. No IUCV connection to the CCS subcomponent of CP exists in this state.

     If the current state is AWL, the VM screen indicating "PRESS ENTER TO CONTINUE" should be displayed at the logical unit. VSCS is waiting for an interrupt from the LU before it displays the VM logo (LUP state). No IUCV connection to the CCS subcomponent of CP exists in this state.

     If the current state is full-screen support mode (FSS), the logical unit is controlled by the application in the virtual machine to which it is logged on, or it has dialed a guest virtual machine using CP DIAL. In either case, the application is responsible for:

     – Sending the correct screens to the logical unit
     – Reading the data from the logical unit when the data becomes available
     – Restoring the logical unit's keyboard.

     Note: If the application does not restore the keyboard, logical units are left in a hung condition and cannot be reset.

   - Determine which response mode is active for this logical unit.

     – Exception response mode (ER)
     – Definite response mode (DR)

     Note: ER and DR refer to the mode that is allowed for this logical unit, not the mode requested for the current operation. Printer LUs always use DR mode.

     See the explanation of the SCHED= operand on the DTIGEN macroinstruction in Chapter 5, "VM/SNA Console Support (VSCS)" in the *VTAM Resource Definition Reference* for more information on exception response mode.

   - Determine whether SCIP recovery is in progress for this logical unit.

     SCIP indicates that a network failure occurred and that VSCS is attempting to reacquire the logical unit.

- Determine whether exception-response error recovery is in progress or pending for this logical unit.

  RP indicates that recovery is pending. VSCS is notified that an exception-response VTAM SEND has failed, but VSCS cannot initiate error recovery because the logical unit's RPL is busy. When VSCS receives the response, it will initiate recovery (exception response mode only).

  RIP indicates that recovery is in progress. When VSCS is notified that an exception-response VTAM SEND has failed, it initiates error recovery.

Error recovery includes the following steps:

1) SEND of a CLEAR request
2) SEND of a Start Data Traffic Request
3) All PIUs are resent, starting with the one that failed first and proceeding until all subsequent PIUs are sent. **All** PIUs are sent in definite response mode. When the last PIU is sent, error recovery is finished (exception response mode only).

- Determine the current VTAM services (VS) state of the logical unit.

  If the current state is TERM, then VSCS is attempting to terminate the logical unit.

  If the current state is NORM, then the logical unit has not been terminated.

d. Is message DTIS65I displayed? If so:

- Does this message contain any IUCV PATHIDs?

  If so, an IUCV connection to the CCS subcomponent of CP is active. Two PATHIDs should be displayed. The first is the IUCV PATHID that VSCS uses for its IUCV requests to CCS. VSCS receives this PATHID when an IUCVCOM CONNECT GCS macroinstruction executes for the logical unit. The second is the IUCV PATHID that CCS uses for its IUCV requests to VSCS. VSCS receives the PATHID from CCS during the logical unit's CCS initialization. If only one PATHID is displayed, it is the first; VSCS has not received CCS's PATHID for this logical unit. This indicates a problem with CP.

  If this message just contains blanks, then a CP connection is not currently active.

- What is the current RPL command?
- Is the RPL currently busy?

  If so, VSCS is waiting for a VTAM RPL command to complete before doing anything else on the logical unit's behalf. Several conditions can cause this to occur, including:

  – Missing I/O interrupt
  – Blocked PAB in VTAM
  – Network problem with the device or routes
  – SIO never issued.

  Issue the VTAM D NET,ROUTES TEST=YES to determine if the route is intact. If it is intact, you need VTAM buffer use and line traces to determine whether the device responded to the last request.

e. Is message DTIS59I displayed?

Note: This message is displayed only if the RECEIVE SPECIFIC option is active.

If so, then is the RECEIVE SPECIFIC RPL busy?

If it is not busy, then VSCS cannot receive any input from the logical unit. Normally, this occurs only when the device is being returned to VTAM, is recovering from an error, or is processing input data for the logical unit.

If the current PS state is AWL, the logical unit may be hung. The AWL state indicates that VSCS is waiting for an interrupt from the logical unit before displaying the VM logo. If the keyboard is currently locked, then VSCS cannot receive the interrupt, and the logical unit is hung.

f. Is message DTIS68I displayed?

If so, then are the SEND or RECEIVE counts incrementing for the logical unit, or are both currently zero?

If both are zero, you need additional information to determine whether the problem is in VSCS or VTAM. For example:

1) If the problem occurred during logon, determine whether the device characteristics are correct for this logical unit. Check message DTIS65I. If the current RPL command is OPNDST and the RPL is busy, determine why VTAM is not completing the request.

2) If SCIP is included with message DTIS66I, determine why recovery of the logical unit is not completing.

3) If the current RPL command in message DTIS65I is CLSDST, and the RPL is busy, determine why VTAM is not completing the request.

2. If the device is a printer or the device was doing a copy function when it hung, issue the VSCS DISPLAY PRTTAB command. This shows all active printers defined to VSCS and the current status, which includes any associated display using a printer.

Additional information can be obtained about each device by using the VSCS DISPLAY ID= command for each logical unit in the printer table. This command provides more detail about the status of each logical unit associated with a printer copy function.

Determining Device Characteristics

To determine if the right VTAM logmode entry was used, issue the VSCS DISPLAY ID=*netid.luname* command. The resulting messages indicate how the device characteristics appear to VSCS.

1. Is message DTIS61I displayed?

If so, note the device type: 3277, 3278, 3767, 2741, TWX, or PRT.

Specifying the wrong VTAM logmode entry, or using the VTAM default logmode entry, leads to incorrect operation of the logical unit. In some cases, DTIPIPUT issues message DTIP27I or message DTIS99I when input is received from the logical unit.

Also note the device model. This determines the screen size that is used for display devices. (Screen size is displayed in message DTIS63I.)

2. Is message DTIS62I displayed?

If so, what is the logical unit type?

0 Non-SNA display or printer device
1 Keyboard/printer device
2 SNA display
3 SNA printer device.

What are the device characteristics?

- EDS indicates that extended data stream commands are supported (for example, use of the Write Structured Field commands). Non-SNA devices, for example, BSC and start-stop devices, *must* have FEATUR2=(EDATS) specified in the NCP definitions if the capability exists.

  Note: EDS is not supported for a non-SNA device connected through an integrated communication adapter (ICA).

- NEDS indicates that extended data stream commands are not supported.
- The APL NEW|OLD field indicates whether the device uses the NEW or OLD APL character set.
- COLOR indicates the device supports color. If the device does not support color, this field is blank.
- SHARED indicates that the device is a shared printer. If the device is not a shared device, this field is blank.

3. Is message DTIS63I displayed?

   Note: This message is issued for display terminals only.

   If so, note the screen size from the PSERVIC field of the VTAM logmode.

   Also note the alternate screen size from the PSERVIC field of the VTAM logmode (or, for some devices, from the Write Structured Field Query reply data).

4. Is message DTIS64I displayed?

   Note: This message is issued for keyboard/printers only.

5. If the device is a printer, use the VSCS DISPLAY PRTTAB command to show whether the printer was defined or acquired by VSCS and to determine its current status.

Determining VSCS Module Levels

A variety of problems can occur if the VSCS modules are not all at the correct level. This means that each module must be for the same release and if maintenance has been applied, all modules affected by the maintenance must be present.

To determine the module levels, take the following steps:

1. Obtain a machine readable dump of VSCS. Use the #CP VMDUMP 0:END DSS FORMAT GCS command, the GCS GDUMP 0- FORMAT GCS command or use an existing dump of VSCS.
2. Start PRTDUMP, and answer "Y" when you are asked if you want to format VTAM control blocks. When you are asked if you want to format VSCS control blocks, also answer "Y" to access the VSCS formatting options.
3. Select option 5, "VSCS module cross-reference." This produces two listings of VSCS modules, one sorted by module name and one sorted by module address.
4. Select option Q "Quit" when the previous option completes.
5. Obtain the printed output just created.

The resulting output looks similar to the condensed version in Figure 18 on page 137.

```
                                VSCS DUMP FORMATTING OUTPUT                                          PAGE   02
                                   SORTED BY MODULE NAME

ADDRESS     NAME     DATE   PTF/APAR        ADDRESS     NAME     DATE   PTF/APAR        ADDRESS     NAME     DATE   PTF/APAR
019E4410  DTICEXIT 93.321                  01A0CAA8  DTICGMSG 93.321                  01A0A288  DTICINIT 94.300
019F0C50  DTICIUCV 94.216                  01A0A9F0  DTICLOGO 93.321                  019D5728  DTICTERM 93.321
01A0D260  DTIIQUIT 93.321                  019D76A8  DTIISDA1 93.321                  01A1D928  DTIISDA2 93.321
:
                                VSCS DUMP FORMATTING OUTPUT                                          PAGE   05
                                   SORTED BY ADDRESS

ADDRESS     NAME     DATE   PTF/APAR        ADDRESS     NAME     DATE   PTF/APAR        ADDRESS     NAME     DATE   PTF/APAR
01403420  DTISLCMD 93.321                  019D4000  DTIISTRT 94.266                  019D5538  DTIITERM 93.321
019D5728  DTICTERM 93.321                  019D5A18  DTIUSER0 00000.                  019D5E40  DTIPRTRT 94.216
019D76A8  DTIISDA1 93.321                  019DF318  DTIPOWRT 94.299                  019E0510  DTIPRFLD 93.321
:
```

Figure 18. Module Cross-Reference in a VSCS Dump

The date field contains a Julian date (YY.DDD), which is:

- Two-character year (YY)
- Period
- Three-character day (DDD).

The exceptions to this format are:

1. **DTIIPARM** and **DTIUSER0-DTIUSER9**, which use the form of YYDDD, period at the end. The DTIUSER*x* field can contain just about anything if the DTIGEN parameter IDENT is supplied. See Chapter 5, "VM/SNA Console Support (VSCS)" in the *VTAM Resource Definition Reference* for more information on the DTIGEN macroinstruction.
2. **DTISMSGM**, which does not contain a date.
3. The data manipulation exit routines **DTIPDDSI**, **DTIPDDSO**, **DTIPDNTI** and **DTIPDNTO** may not be listed if user coding conventions are not similar to the VSCS conventions.

If maintenance has been applied, the PTF/APAR field is filled in with the information found in the module. Otherwise, the field is left blank. For example, in Figure 18, UV12345 has been applied to DTICEXIT, and DTICGMSG is at base level.

With this list you can now determine the following information about the VSCS modules:

- Base level of all modules by date.
- Current maintenance level of each module found.
- Whether user-provided start options are being used (indicated by the presence of module name DTIUSER*x*).
- Whether user-provided data manipulation exits are present (indicated by using the date, which will be greater than the base, missing, or not valid). The data manipulation exits are:
  - DTIPDDSI

  - DTIPDDSO
  - DTIPDNTI
  - DTIPDNTO.
- Where a particular module resides in storage, shown by the address in the first column to the left of the module name.

Determining the Status of a VSCS Printer LU

To determine the status of any VSCS printer, there are two VSCS commands available:

- VSCS DISPLAY PRTTAB, which shows the status of all printers that currently reside in the printer table, whether or not the printer is in session with VSCS
- VSCS DISPLAY ID=*netid.printername*, which displays the detailed status of a printer LU that is currently in session with VSCS.

Each command has its advantages, but the DISPLAY PRTTAB provides more of a general look at the printer status. From the output of the DISPLAY PRTTAB command, you can determine:

- How many printers can be defined
- How many printers are defined
- If printer sharing is supported
- How a printer was defined to VSCS
- If a printer is in use
- Which display LU is currently using a printer
- If a printer is currently being acquired
- If a printer is currently in session with VSCS
- If a request to release the printer has been received
- If a printer has been forced using the VSCS FORCE command.

The following figure explains some of the potential conditions that are displayed in the VSCS DISPLAY PRTTAB command output.

Display	STATE	TYPE	COND	Description
Yes	LOGON	OPER	INACT	Copy requested and display is still waiting for the printer to be acquired.
No	LOGON	OPER	INACT	Copy requested and display has been released because the printer could not be acquired within the specified time (TIMECPY=).
Yes	LOGON	OPER	RELES	Copy requested and the display is waiting for the printer to be acquired. However, a request to release the printer has been received.
Yes	LOGON	OPER	FORCE	Copy requested and the display is waiting for the printer to be acquired. However, a VSCS FORCE command has been issued for the printer.
No	IDLE	LOGON	ACT	This printer was attached to VSCS with the VTAM VARY command and is currently not in use.

Figure 19. DISPLAY PRTTAB Command Output Conditions

When more specific information is needed for a printer that is in session with VSCS, use DISPLAY ID=*netid.printername*. The output from the DISPLAY ID= command allows you to determine:

- LU characteristics
- SEND/RECEIVE counts
- Time the LU logged on to VSCS
- LU states.

Operational Differences between VSCS and CP

VSCS strives for total compatibility with native CP-controlled terminals, but that cannot be achieved for several reasons. Some of the reasons are:

- SNA versus non-SNA device differences
- Performance implications
- Interface restrictions
- Interface timing
- Operating inconsistencies
- Operational requirements.

These reasons for not achieving total compatibility result in the following VSCS restrictions.

- General
  - Protocol converters and OEM devices must react identically to IBM devices in all situations to obtain the desired results.

  - The proper VTAM logmode must be used when logging on to VSCS. See "Logon Mode Table" in the *VTAM Resource Definition Reference* for additional information on the logmode requirements for VSCS LUs.

- Display terminals
  - A VSCS LU should *never* log on as the operator console for VSCS, VTAM, or CP. This can cause major problems if the LU goes into CP mode.
  - Logon from logo:
    - Only supported if the command lines are provided to VSCS.
    - VM logo command lines must contain three fields: USERID, PASSWORD, and COMMAND.
    - The Insert Cursor must be located at the first available position of the USERID field.
    - Password mask must be in the PASSWORD field provided.
  - VMEXIT cannot be used as a valid user ID from a VSCS LU.
  - Color and highlighting changes to the input and status area are reflected more often on SNA LUs.
  - When issuing a message to yourself, the input is shown in the redisplay color and the incoming message is shown in the CP color. On a native device, only the redisplay color is used.
  - Stacked input data is redisplayed exactly as entered, not separated by command as on native devices.
  - For a PF key defined "CP SET PF*nn* TAB n1 n2 ...," VSCS inserts blanks and native CP inserts null characters. This may cause the results to look different.
  - Input data is redisplayed in the case entered.
  - Full-screen applications are expected to retrieve data when notified that input has been received by VSCS.
  - Full-screen applications are expected to reset the keyboard, in order to allow the LU to enter additional data.

    On logical unit type 2 devices (SNA displays), VSCS converts the SIGNAL Request-to-Send RU that is generated by the ATTN key into a PA1 key. The ATTN key functions regardless of the session state of the LU-LU session with VSCS. That is, the ATTN key functions even when the status indicators on the display show "X SYSTEM" or "X CLOCK." When these indicators are displayed, the keyboard is locked. The ATTN and SYS REQ keys are the only keys that function in these states.

- Keyboard/printer devices
  - The DIAL command is not supported.

Symptoms of VSCS Problems

This chapter divides VSCS problems into the following general problem types.

Note: Documentation requirements may differ if VSCS is running in a separate virtual machine from VTAM. It is also important to understand that VSCS errors do not always result in a message, user disconnection, or user logoff.

- "Initialization Problems" on page 143
  - Initialization never completes
  - VSCS terminates immediately upon starting
  - Loop occurs during initialization
  - Other initialization problems.
- "Termination Problems" on page 150
  - Termination never completes
  - VSCS terminates prematurely
  - User's session terminates prematurely
  - Operator commands terminate VSCS unexpectedly
  - Other termination problems.
- "Logon Problems" on page 157
  - First logon attempt fails
  - Logon never completes (logical unit is hung).
- "Logoff and Disconnect Problems" on page 159
- "Wait and Loop Problems" on page 159
  - All VSCS logical units are hung
  - One or more VSCS logical units are hung.
- "Message Problems: Message DTIS99I Indicates a VSCS Internal Error" on page 176
  - Message DTIS99I indicates an internal error.
- "Incorrect Output Problems" on page 178
- "Performance Problems" on page 182
  - Paging and storage problems
  - Slow response time on terminals.
- "Other VSCS Problems" on page 183
  - SNA dial problems
  - Printer sharing problems
  - Operator command is not processed properly.

Overview of Normal VSCS Initialization Process

This section contains an overview of what takes place during normal VSCS initialization. This is the procedure that VSCS follows each time it is initialized. Figure 20 on page 142 shows the components involved in initialization.

GCS processes IUCV requests from VSCS that affect path ID setup and breakdown. IUCV external interrupts are handled by GCS, which gives control to VSCS

after some validation is done. All other IUCV requests and responses issued by VSCS are handled directly by CP.

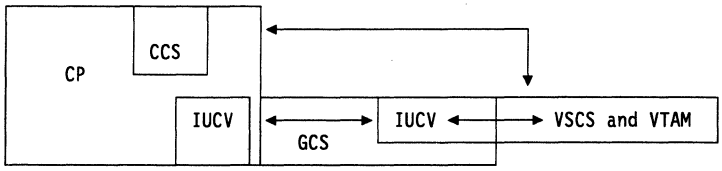


Figure 20. Components Involved in VSCS Initialization

1. VSCS storage is initialized with DTIUSERx, if specified.
2. VSCS issues message DTIC01I.
3. VSCS issues IUCVINI SET to GCS.

   GCS sets return code after processing request.
4. VSCS issues IUCV QUERY to CP.

   CP responds with MAXCONN and EIB size.
5. If VSCS trace is active, message DTIS74I is issued.
6. VSCS issues global IUCVCOM CONNECT and if successful:
   - CP provides path ID and a return code back to GCS
   - GCS passes path ID and a return code back to VSCS
   - VSCS issues a system WAIT at this point
   - CP generates an external interrupt for "CONNECT COMPLETE"
   - GCS passes "CONNECT COMPLETE" back to VSCS
   - CP generates an external interrupt for VM logos
   - GCS notifies VSCS of a pending message.
7. VSCS issues IUCV RECEIVE
   - VSCS processes "CONNECT COMPLETE"
   - CP sends VM logo data to VSCS.
8. VSCS processes VM logo information.
9. VSCS issues message DTIC02I.
10. VSCS attaches the utility services subtask to GCS.
11. VSCS issues VTAM OPEN ACB through GCS to VTAM.
12. VSCS schedules a work element containing the VTAM user ID to presentation services.
13. VSCS issues message DTIV04I.
14. VTAM interface should be functional; VSCS is fully functional.
15. VSCS issues an IUCV SEND to supply CP with the VSCS user ID.

Initialization Problems

First you should look for initialization problems in these areas:

1. A storage shortage could occur if:
   - VSCS is in the same virtual machine as VTAM in a large environment
   - VSCS is in a small virtual machine
   - VSCS DTIGEN parameters are incorrect:
     - Trace table is too large for virtual machine size
     - Printer table is too large for virtual machine size
     - Block multiplier (BLKMULT) is too large for virtual machine size.
   - Specified size for MAXCONN is too large. MAXCONN is a parameter in the directory entry for the user ID running VSCS.
2. Parameters are misspelled or missing; VSCS initialization uses most of the user parameters coded for:
   - CP
   - GCS
   - VTAM APPL definition
   - DTIUSER*x* (if specified)
   - VM directory entry for the VSCS virtual machine.
3. The wrong disk is linked to the VSCS load module.
4. The disk containing the VSCS load module has changed since it was last accessed by this virtual machine.
5. The VTAM application program major node is inactive or is not valid.
6. The VM directory is incorrect (the user ID for the IUCV option card must be '*CCS').

If none of the previous items is causing the problem, follow the procedure for your symptom:

- "Initialization Never Completes (A Wait Follows Message DTIC01I or DTIS74I)"
- "VSCS Terminates Immediately upon Starting" on page 145
- "A Loop Occurs during Initialization" on page 148
- "Other Initialization Problems" on page 149.

Initialization Never Completes (A Wait Follows Message DTIC01I or DTIS74I)

1. Is SNA enabled?

   A delay during initialization is normal; VSCS must wait for external interrupts for "CONNECT COMPLETE" and the VM logo. A long wait indicates a problem. ENABLE SNA is used to open a path from IUCV to CCS, and if SNA is disabled, CCS does not process the connection request.

   Note: If SNA is disabled and VSCS started, it may appear that most operator commands do not complete.

   To make sure that SNA has been enabled, use one of these CP commands:

   ```
   #CP ENABLE SNA
   #CP ENABLE ALL
   ```

#CP ENABLE SNA user ID

To determine what book describes CP commands for your operating system, see Table 80 on page 1165.

2. Are GCS and the VTAM recovery machine active?

   Issue the GCS command QUERY GROUP to see if GCS is active. If you get the message "HCP0045E GROUP not logged on" or "MK045E GROUP not logged on," GCS is not active. You must re-ipl GCS using the CP IPL GCS command. Use the CP QUERY NAMES command to see if the VTAM recovery machine is active. If it is not active, you need to initialize it.

3. Does GCS have control register 0 bit 30 on?

   To determine if control register 0 bit 30 is set on, display the control registers from the VSCS operator console (DISPLAY X'0') or look in a dump of the VSCS virtual machine (all registers are shown at the beginning of the dump). GCS must set the control register during GCS initialization in order to receive IUCV external interrupts from CP. If control register 0 bit 30 is off, verify that the VSCS directory entry is correct (or the VTAM directory entry, if VSCS will run in the same virtual machine as VTAM). Continue problem resolution with GCS.

4. Did CP process the IUCV CONNECT request without error? ("Without error" means that no error messages were issued by CP, GCS, or VSCS.)

   If so, continue with step 5. If not, the problem is in CP or GCS.

   This is shown in the CP internal trace entries hex 15 (IUCV) and hex 16 (CCS) in the CP internal trace table. Before starting VSCS, limit the amount of the external trace generated by issuing SET CPTRACE or TRSOURCE.

   To determine what book describes the CP trace for your operating system, see Table 80 on page 1165.

5. Did GCS reflect the external interrupts (for "CONNECT COMPLETE" and the VM logo) to VSCS?

   To determine this, look at:

   - CP internal trace of IUCV and CCS requests
   - GCS external trace of external interrupts received
   - VSCS trace with CCS active.

   If GCS did not reflect the external interrupts to VSCS, the problem is in GCS.

   If all of these conditions are met, the IUCV will be properly initialized and you will see message DTIC02I.

   To determine what book describes IUCV, CCS, and GCS commands for your operating system, see Table 80 on page 1165.

6. Was VSCS previously active, and now cannot be restarted?

   If this is the case, the problem may arise from an abend or premature termination (message DTIC07I) of VSCS during the previous VSCS initialization (assuming CP has not been reinitialized since then). If CP and GCS did not clean up after the abend, VSCS cannot restart. Check the operator console listing to see if any problems were encountered on the previous termination. If there were problems, re-IPL GCS. If initialization still fails, re-IPL CP.

7. If you do not have enough information, or cannot solve the problem, re-create the problem with the GCS, CP, and VSCS traces active externally. Use your best judgment in trying to decide which product has the problem. If you suspect GCS or VSCS is causing the problem, get a VSCS dump. Otherwise, pursue the problem with CP.

   To determine what book describes CP commands for your operating system, see Table 80 on page 1165.

8. Go to "Reporting the Problem to IBM" on page 112.

VSCS Terminates Immediately upon Starting

1. How far did initialization go?

   Get the operator console listing, and use "Overview of Normal VSCS Initialization Process" on page 141 to evaluate the console messages to see how far the initialization progressed. VSCS messages that occur during successful initialization are DTIC01I, DTIC02I, and DTIV04I, in that order.

   Note: Messages issued for starting the VSCS internal trace are not included here.

   If you did not get DTIC01I, the problem may be:

   - GCS is not active.
   - VSCS is not linked correctly.
   - VSCS virtual machine is not large enough to hold the DTIISTRT load module.
   - DTIGEN parameters are incorrect.
   - GCS is dumping the VSCS virtual machine.

   If you receive DTIC01I but not DTIC02I, the problem may be:

   - VM directory entry for the VSCS virtual machine is wrong.
   - GCS is dumping the VSCS virtual machine.
   - CP or GCS found a problem (other than an abend) while processing an IUCV request or response. VSCS should issue a message to identify the failure and reason. See step 2 on page 146.

   If you receive DTIC01I and DTIC02I, but not DTIV04I, the problem may be one of the following:

   - If you receive message DTIC14I, then errors were encountered processing the VM logo. The errors can be:
     - ABORT - Bad VM logo data received.
     - STOR - Storage could not be obtained for the VM logo.
   - VTAM OPEN ACB error. (VTAM return codes show what caused this.)
   - VTAM APPL definition is incorrect.
   - VTAM APPL definition is not active.
   - VTAM storage shortage.
   - VSCS storage shortage.

- VSCS DTIUSERx problem, such as incorrect parameters. (See "Other Initialization Problems" on page 149.)
- GCS is dumping the VSCS virtual machine.

If you receive DTIV18I, you must issue the SYSTEM RESET command or use the CP FORCE command to stop and autolog the GCS recovery machine to initialize VSCS.

2. Was an error message issued?

   If so, try to find the source of the message. If not, go to step 5 on page 147.

   - CP messages begin with HCP
   - GCS messages begin with GCT
   - CMS messages begin with DMS
   - VSCS messages begin with DTI
   - VTAM messages begin with IST.

   If GCS issued the message, and it is not an abend, get a VSCS dump. See "VSCS Dump" on page 453 for more information on the VSCS dump. Then go to "Reporting the Problem to IBM" on page 112.

3. Did you get an error message with a return code or IPTYPE?

   If so, find the explanation for it, and determine what component issued the message. If not, go to step 4 on page 147.

   Table 12 shows the sources of return codes and IPTYPEs.

Table 12. Source of IPTYPE or Return Code During Initialization

Message	Source of IPTYPE or Return Code
DTICxxx	• GCS • VM • VSCS (very rarely)
DTIIxxx	• GCS • VSCS
DTIPxxx	• VM • VSCS (very rarely)
DTISxxx	• GCS • VSCS
DTIVxxx	• GCS • VSCS • VTAM

- If you receive message DTIC05I with return code 878 or 804, then GCS had a GETMAIN failure getting storage for an IUCV request. See "Storage-Related Problems" on page 179 for more information.
- If you receive message DTIC07I with IPTYPE=03, the previous termination did not complete, or two IUCV CONNECTs with the same name have been issued from the same GCS group. IPTYPE=03 is an IUCV SEVER request from CP, indicating CP is rejecting the connection request.

  If you can re-create the problem, follow these steps:

  a. Use the DTIGEN operands to activate VSCS internal trace options, including the external trace option. See Chapter 5, "VM/SNA Console

Support (VSCS)" in the *VTAM Resource Definition Reference* for information on specifying VSCS trace options in the DTIGEN.

b. Initiate TRSOURCE.

c. Start VSCS using the newly created DTIUSER*x*.

d. After the failure occurs, stop TRSOURCE.

e. Process the output, which should contain the failing sequence. See "Collecting External Trace Data (VM)" on page 359 for information on using operating system utilities to collect, format, and print VSCS trace output.

- If you receive message DTIC09I, you should determine which component generated the return code. To determine what book describes the IUCV connect return codes for your operating system, see Table 80 on page 1165.
- If you receive an error message not yet mentioned, look in the system programmer's guide for your operating system to interpret the meaning of the return code. To determine what book describes the IUCV connect return codes for your operating system, see Table 80 on page 1165.

  If the message has an IPTYPE, the IPTYPE is one of the following:

  1 Pending connection
  2 Connection complete
  3 Severed connection
  4 Quiesced condition
  5 Resumed connection
  6 Incoming priority reply
  7 Incoming reply
  8 Incoming priority message
  9 Incoming message.

4. Did you get an error message without a return code or IPTYPE?

   If so, does the message text adequately explain the problem? If you can fix the problem by correcting user-specified parameters, do so. If not, get the VM directory entry for the VSCS virtual machine, VSCS installation information, and GCS specified parameters. Then go to "Reporting the Problem to IBM" on page 112.

   Note: Figure 21 on page 148 shows all the VSCS messages that can be issued for an initialization problem.

5. Is the problem an abend?

   If an abend occurs during initialization, the problem is most likely caused by an error in the installation process of VSCS. Find out how VSCS was installed and linked.

   An ABEND0C2 might also occur if one of the following is true:

   - VSCS is not running in an authorized virtual machine.
   - The VSCS APPL definition does not specify AUTHEXIT=YES.

   An ABEND0C1 might also occur if one of the following is true:

   - The VSCS APPL definition does not specify AUTHEXIT=YES.
   - The VTAM DCSS is too small to contain all of IST0019C1.

If an ABEND706 is received before any VSCS messages are displayed, then there was a failure linking the VSCS load module. That failure left the load module "non-executable."

Otherwise, if the failure results from a link failure with a DTIUSERn load module, message DTII11I is issued.

If you cannot find and correct the error, get a VSCS dump (created automatically for an abend) and go to "Reporting the Problem to IBM" on page 112.

6. Is the problem something else?

   If you still cannot identify the problem, get a VSCS dump and go to "Reporting the Problem to IBM" on page 112. The VSCS dump is described in "VSCS Dump" on page 453.

DTIC03I	*modname* VSCS INITIALIZATION FAILED—INSUFFICIENT STORAGE AVAILABLE
	Note: The virtual machine is too small to initialize VSCS or the DTIGEN operands are coded incorrectly.
DTIC04I	*modname* VSCS INITIALIZATION FAILED—UNRECOVERABLE ERROR DETECTED *description*
DTIC05I	*modname* VSCS INITIALIZATION FAILED—IUCVINI SET MACRO FAILED WITH RETURN CODE *nn*
DTIC06I	*modname* VSCS SEND OF VTAM USERID TO CCS FAILED
DTIC07I	*modname* VSCS INITIALIZATION FAILED—GLOBAL CONNECT FAILURE, IPTYPE = *iptype*
DTIC09I	*modname* IUCV GLOBAL CONNECT FAILURE, RETURN CODE = *nn*
	Note: This message is the result of an error from GCS or CP.
DTIC14I	*modname* DYNAMIC LOGO UPDATE *status reason*
DTII03I	DTIITASK COMMUNICATION SERVICES INITIALIZATION FAILED. VSCS TERMINATING.
DTII04I	DTIITASK ATTACH FAILED FOR DTISUTSK, RETURN CODE = *xx*
DTII06I	DTIITASK IDENTIFY FAILED FOR DTISUTSK RETURN CODE = *xx*
DTII14I	*modname* ABNORMAL TERMINATION OF VSCS IS IN PROGRESS INITIATED BY *mod2*
DTII15I	DTIUSER*n* COMPILED WITH THE WRONG LEVEL OF DTIGEN MACRO
DTII16I	DTIUSER=*n* SPECIFIED IN DTIGEN MACRO DOES NOT MATCH START PARAMETER
DTIP15I	*modname* IUCV RECEIVE FAILURE, IUCV RETURN CODE = *nn*
	Note: CP provides the IUCV return code.
DTIS38I	*modname* VSCS ABEND FOR (PSIVSIUS) PROCESS — SYSTEM COMPLETION CODE = *xxxx*
DTIS99I	INTERNAL VSCS ERROR — *modname n* [luname]
	Note: If this happens during initialization and you have supplied a DTIUSER*x* for initialization, it may be a DTIUSERx specification error.
DTIV01I	*modname* FAILURE FOR *reqtype* RETURN CODE = *rplrtncd rplfdb2*
DTIV07I	*modname* VSCS INITIALIZATION FAILED-OPEN ACB FAILURE, ACBERFLG = *xx*
DTIV18I	VSCS INITIALIZATION ABORTED, SYSTEM RESET REQUIRED

Figure 21. Messages Indicating VSCS Initialization Problems

A Loop Occurs during Initialization

If 90% – 100% of CPU time is being used and that is not a normal workload for the CPU, the problem could be one of the following:

- GCS is dumping the virtual machine.
- A VTAM temporary storage shortage has occurred, and VSCS is trying the VTAM request again.

- VSCS, VTAM, or GCS may have a control block chained to itself, resulting in an infinite loop.
- There is a loop in CCS or IUCV, so that the VSCS virtual machine appears to be looping, but is not.

If VSCS is running in the same virtual machine as VTAM, it is very difficult to determine where VSCS is looping because of the low priority at which VSCS is running. If you are using the TRACE BRANCH trace to record the loop, the trace must run for at least one minute.

If a dump message is issued, see "VSCS Terminates Immediately upon Starting" on page 145 for information on how to proceed.

1. Determine amount of storage available.

   To determine the amount of VTAM storage available and which storage pools need to be enlarged, use either the SMS (buffer use) trace or the VTAM DISPLAY BFRUSE command. These service aids are described in Chapter 11, "Using VSCS Service Aids (VM)."

2. If you cannot find any errors from the display of the VTAM buffer pools, collect this documentation:
   - A few seconds of the CP virtual machine branch trace. See Table 80 on page 1165 to determine what book describes the CP virtual machine branch trace.
   - VSCS dump.
   - VSCS operator console listing.
   - VSCS trace with CCS active, running before the loop started.
   - VSCS trace with any other options active, running before the loop started.
   - Series of loop addresses obtained with the CP DISPLAY PSW command or from the CP virtual machine trace output.

     To determine what book describes CP commands and the CP trace for your operating system, see Table 80 on page 1165.

3. Go to "Reporting the Problem to IBM" on page 112.

Other Initialization Problems

1. Are parameters specified correctly in the DTIGEN macro?

   Figure 22 on page 150 shows messages issued for a problem involving incorrect parameters.

   The parameter value representing the DTIUSER number must be a number in the range 0–9, and one byte in length. If you do not get the correct DTIUSERx module in message DTII07I, re-access the VSCS LOADLIB and make sure you enter this command correctly:

   ```
   VSCS START PARM=x   (where x is the DTIUSER number)
   ```

   If you are still experiencing problems, have an assembly listing of the DTIGEN macroinstruction available and go to "Reporting the Problem to IBM" on page 112.

DTII07I	USING INITIALIZATION PARAMETERS IN CSECT *csectname*
DTII08I	INITIALIZATION CSECT *csectname* NOT FOUND OR INVALID
DTII09I	INVALID INITIALIZATION START PARAMETER
DTII10I	DEFAULT INITIALIZATION PARAMETERS BEING USED
DTII11I	LOAD OF INITIALIZATION CSECT DTIUSER*n* FAILED, LOAD ABEND CODE = *yyy*, REASON CODE = *zzz*
DTII15I	DTIUSER*n* COMPILED WITH THE WRONG LEVEL OF DTIGEN MACRO
DTII16I	DTIUSER = *n* SPECIFIED IN DTIGEN MACRO DOES NOT MATCH START PARAMETER

Figure 22. VSCS Messages Useful in Analyzing a Parameter Problem

2. Is VSCS accessing the proper DTIUSERx?

   If you see message DTII10I, either you did not select the correct DTIUSERx, or the loadlib containing the proper DTIUSERx module is not accessible. Issue the QUERY LOADLIB command and see if the correct loadlib is displayed in the query output. If the correct loadlib is not displayed, reissue the GLOBAL LOADLIB command, specifying the proper loadlib and all of the other loadlibs displayed in the query output, then restart VSCS again. Make sure the VSCS LOADLIB contains the DTIUSER you are trying to use. If you have verified that the DTIUSERx is correct, and you still have problems, get the portion of the operator console listing showing VSCS initialization and a list of the VSCS LOADLIB members.

   If you see message DTII11I, then the requested DTIUSERx was found, but an error occurred attempting to load it. Check the LOADLIB containing DTIUSERx for errors.

3. Was the DTIUSERx assembled with the correct level of the DTIGEN macro?

   If VSCS initialization detected a problem with the size or contents of the DTIUSERx loaded, message DTII15I will be issued. Verify that you replaced the DTIGEN macroinstruction in the VTAM MACLIB and that, when assembling, the correct VTAM MACLIB is set up using the GLOBAL MACLIB command.

4. Go to "Reporting the Problem to IBM" on page 112.

Termination Problems

VSCS issues one of the following messages to indicate termination is in progress:

DTII13I *modname* NORMAL TERMINATION OF VSCS HAS BEEN INITIATED BY *mod2*

DTII14I *modname* ABNORMAL TERMINATION OF VSCS IS IN PROGRESS INITIATED BY *mod2*

The DTII14I message is usually followed by a request for a dump.

If you receive message DTII13I, and you did not enter the VSCS HALT or QUIT operator command, and VTAM is not terminating, check for previous VSCS or VTAM messages. They may indicate why termination is in progress.

Problems during termination include:

Choose the symptom most like yours and follow the procedure for that problem.

Termination Never Completes

1. Was termination initiated by a VSCS operator command?

   If so, get a VSCS dump and go to "Reporting the Problem to IBM" on page 112. If not, issue the VSCS HALT command.

   Note: If the VSCS Utility Services subtask abended previously, normal termination cannot take place, and you must issue the VSCS CANCEL command to terminate VSCS.

2. Is VSCS still in a wait state?

   Display all active sessions with the VTAM DISPLAY LU and VSCS DISPLAY ID=*luname* commands. Issue a VTAM VARY INACT command for each active session. This should allow VSCS to continue the termination. If this does not complete termination, use the VSCS FORCE ID=*luname* command.

3. Is termination still not completing?

   Get the following documentation:

   - VSCS operator console listing
   - VSCS dump (see "VSCS Dump" on page 453).

   Use the VSCS CANCEL command to terminate VSCS.

4. Go to "Reporting the Problem to IBM" on page 112.

VSCS Terminates Prematurely

1. Did message DTIP15I appear?

   This indicates that presentation services caused the termination. Get the following documentation:

   - VSCS dumps, if available (generated by previous VSCS abends or VSCS internal errors).
   - VSCS operator console listing showing any messages that tell which module invoked VSCS termination.
   - Trace output from any VSCS traces active at the time of VSCS termination.

   Go to "Reporting the Problem to IBM" on page 112.

2. Did message DTIV02I appear?

   If you receive this message, issue a VTAM BFRUSE command and see if the VTAM CRPL buffer pool has expanded. If the problem is not caused by buffer expansion, get the following documentation:

   - VSCS VTAM exit trace, if running at the time of the problem
   - VSCS dump.

   Go to "Reporting the Problem to IBM" on page 112.

Note: Message DTIV02I is issued each time a RECEIVE ANY macroinstruction request fails. Message DTIV05I is issued only when all RECEIVE ANY RPLs are inactive.

3. Did an ABEND0A9 occur?

   This abend code indicates that VTAM terminated while processing under control of the VSCS VTAM services subtask. You must get this documentation:

   - VSCS VTAM exit trace if running at the time of the problem
   - VSCS dump created for any previous VSCS abends.

   Go to "Reporting the Problem to IBM" on page 112.

4. Did VSCS terminate with abend code 878 or 804?

   If so, collect the following documentation:

   - VSCS dump created for the abend.
   - VSCS global and storage control blocks. Process the VSCS dump with PRTDUMP, using VSCS dump formatter options 1 and 2. These options format the VSCS global and storage control blocks.
   - VTAM storage shortage documentation.

   Go to "Reporting the Problem to IBM" on page 112.

5. Did VSCS terminate with another abend code?

   If recovery is possible, a message indicates that recovery is in progress and processing will continue.

   If recovery is not possible, VSCS issues message DTII14I and begins termination. Recovery cannot occur for these abend codes:

   ```
   0A8 0A9 122 13E 222 322 422 522
   ```

   All but 13E and 222 automatically produce a dump.

   In addition to this dump, get:

   - VSCS trace with CCS active, at the time of the abend
   - VSCS operator console listing showing messages around the time of the abend.

   Go to "Reporting the Problem to IBM" on page 112.

6. Is there a storage shortage?

   A storage shortage can cause VSCS to terminate. See "Storage-Related Problems" on page 179 for more information. Issue the VSCS STORAGE command and the first message, DTIS15I, indicates if there is currently a VSCS storage shortage.

7. Did any network failures just occur that involve VSCS logical units?

   If so, and if SCIP recovery is specified for VSCS, then both VSCS and VTAM require additional storage to perform recovery.

   See "Controlling VSCS SCIP Recovery" in the *VTAM Network Implementation Guide* for information on controlling VSCS SCIP recovery.

   Note: If the VTAM VARY LOGON command is used to attach all LUs to VSCS, do not use SONSCIP=YES in the APPL definition for VSCS.

8. Is there a VTAM storage shortage?

   Use the VTAM DISPLAY BFRUSE command to determine if any VTAM buffer pools are experiencing a shortage. If you cannot execute a DISPLAY command, use the VTAM SMS (buffer use) trace. These service aids are described in Chapter 11, "Using VSCS Service Aids (VM)."

   If the shortage is affecting VSCS directly, the following messages may appear:

   - One or more DTIV02I messages
   - IST561I (storage unavailable)
   - IST154I (storage expansion failure).

   When a VTAM RPL-based macroinstruction fails with *rplrtncd* = 08 and *rplfdb2* = 00, there is a shortage of either VTAM LPBUFs or VTAM CRPLs. Normally, 08 with 00 indicates that VTAM actually ran out of CSA while attempting to expand a buffer pool. The defined size of CSA may be too small. Determine if the size of the CRPL or LPBUF buffer pools are fixed in real storage by checking the F operand on the VTAM CRPLBUF and LPBUF buffer pool start options. If CRPL and LPBUF are fixed and VSCS is running in a separate virtual machine than VTAM, then VTAM may not be able to expand its buffer pool from the VSCS virtual machine. Determine if the DIAG98 directory option is specified for the VSCS virtual machine. The DIAG98 directory option allows VTAM to expand its fixed buffer pools in an application's virtual machine. DIAG98 prevents RPL-based macroinstructions from failing with a storage shortage when CSA is available.

   See "Buffer Pools" in the *VTAM Resource Definition Reference* for information on VTAM buffer specifications. If you cannot resolve the problem by adjusting the depleted VTAM buffer pools, go to "Reporting the Problem to IBM" on page 112.

9. If you do not know the cause of the termination, collect the following documentation:

   - VSCS dump
   - VSCS operator console listing
   - Output from any traces active at the time of the problem.

   Go to "Reporting the Problem to IBM" on page 112.

User's Session Terminates Prematurely

1. Are there any messages on the VSCS operator console listing?

   Check the console for VSCS messages indicating that the logical unit is being terminated, or has just been terminated. These messages may indicate a reason for the termination.

2. Was a VTAM VARY INACT, VSCS FORCE, or CP FORCE command issued? If any of these operator commands were issued, VSCS responded by terminating the logical unit.

3. Did any errors occur during request processing for this device? If so, get the following documentation:

   - VSCS trace running externally with all options active except FRE and GET
   - VIT, specify OPT=(API,MSG,PIU), MODE=EXT
   - If the failing LU is known, MODIFY TRACE TYPE=BUF, ID=*luname*.

4. Did the termination occur right after a particular command was issued or a specific application program was started from a particular device type? If so, report the command or application program.

   If you can, re-create the problem with all VSCS trace options active, and report that information. Otherwise, provide as many details as you can about the device you were using, and about the problem.

5. If you have not resolved the problem, go to "Reporting the Problem to IBM" on page 112.

   Note: Use of exception response mode may increase the frequency of session terminations because of limited error recovery, which is considered normal operation for VTAM V3R1.2 and later.

Operator Commands Terminate VSCS Unexpectedly

Do commands other than HALT, CANCEL, or QUIT cause VSCS to terminate? If so, note what these commands are. Get the operator console listing and any dumps that were produced, and go to "Reporting the Problem to IBM" on page 112.

Other Termination Problems

If you have a different problem with termination, or you do not know what the problem is, get:

- VSCS dump
- VSCS operator console listing
- Output from any VSCS traces that are active.

Go to "Reporting the Problem to IBM" on page 112.

VM Logo Problems on an SNA Device

Before looking into potential problems in VSCS logo processing, make sure that the logo appears correctly on a CP-controlled terminal.

If message DTIC16I is issued during initialization, the VM logo received from CP does not fit on a 24x80 display screen, or the VM logo contains extended data stream commands. This may indicate why the normal (standard) logo is not used.

If you have modified the VM logo, see if the default VM logo works. If it does, your modified logo may not be supported on VSCS.

If you decide to write your own logo, keep the following points in mind:

- Changes to the VM logo are considered to be user modifications. For information about VSCS limitations with the VM logo, see "VM Logo Usage" in the *VTAM Network Implementation Guide*.
- User modification of the command lines is not recommended.
- Use of start buffer address (SBA), repeat to address (RA), erase unprotected to address (EUA), or insert cursor (IC) commands are not allowed when modifying the logo body. These screen commands will be replaced with blanks (hex 40), which may alter the appearance of the logo. Other screen commands may not be accepted by all types of display devices, and may result in I/O errors.

- The default colors are white for the online message and blue for the logo body and input area. If you change the colors, you must start with the online message. If you use start field (SF) commands to control the colors, you must protect the online message and logo body.
- Imbedded screen control commands are taken into consideration when evaluating the width of a logo line.
- The value of the start field attribute byte must be within the range hex 40 through hex FE. If you do not define a value within this range, VSCS replaces the start field order (start field ID and attribute) with the value hex 4040. After replacement, the following can occur:
  - Logo data is improperly positioned (for example, logo lines are shifted all over the screen).
  - Alternate logo displays because the width of the user-defined logo exceeds the width of the screen.

See "VM Logo Usage" in the *VTAM Network Implementation Guide* for any VSCS logo uniqueness.

Check the following symptoms for your problem.

1. Logo lines shifted all over the screen and logo is not readable.
   a. Verify that the appropriate VTAM logmode entry for the logical unit was used. (See "Determining Device Characteristics" on page 135.)
   b. If VSCS data exit (DTIPDDSO) is coded, verify that the exit is not altering the start buffer addresses or the screen attributes in the outbound data stream.
2. Only the online message is displayed.

   This indicates that both the normal logo and the single-line (alternate) logo exceeded the capacity of the device.
   a. Verify that the appropriate VTAM logmode entry for the logical unit was used. (See "Determining Device Characteristics" on page 135.)
   b. Verify that either the normal or the single-line (alternate) logo will fit on the screen size of the failing logical unit.
3. Only a single-line (alternate) logo is displayed.

   This indicates that the combination of the normal logo, online message, and command lines would not fit on the screen because line width (of logo body) or number of lines allowed was exceeded.
   a. Verify that either the normal or the alternate logo will fit on the screen size of the failing logical unit.
   b. Verify that the appropriate VTAM logmode entry for the logical unit was used. (See "Determining Device Characteristics" on page 135.)
4. Screen colors or highlighting are incorrect or missing.
   a. Verify that the appropriate VTAM logmode entry for the logical unit was used. (See "Determining Device Characteristics" on page 135.)
   b. Check the modified logo to confirm that screen color and highlighting are inserted correctly.

c. If the VSCS data exit (DTIPDDSO) is coded, verify that it is not altering the start buffer addresses or the screen attributes in the outbound data stream.

5. The logo fields are unprotected.

   This can be normal when using the start field (SF) command to create highlighting or color in the logo body.

   a. Check the modified logo to confirm that screen color and highlighting are inserted correctly.

   b. Ensure that any embedded set attribute commands are paired (balanced) in the normal logo.

   c. If the VSCS data exit (DTIPDDSO) is coded, verify that it is not altering the start buffer addresses or the screen attributes in the outbound data stream.

6. Logo flashes on the screen several times, then the USSMSG is displayed.

   When this happens, there should be an indication at the LU of the type of error that is occurring. Check for any DTIC16I messages that occurred at initialization time along with the characteristics of the LU. (See "Determining Device Characteristics" on page 135.) The most likely causes are:

   - Logo contains characters or commands not supported by the LU.
   - The wrong command code was used to send the logo.
   - The wrong VTAM logmode was used.

7. Command line problems.

   Note: Alteration of the command lines is not recommended.

   a. Cursor positioned wrong, or missing.

      The command lines provided by VM contain an Insert Cursor command in the first available position of the user ID field. If this command is missing, then the cursor can be positioned anywhere.

   b. Command lines protected

      1) Ensure that any embedded set attribute commands are paired (balanced) in the normal logo.

      2) If the VSCS data exit (DTIPDDSO) is coded, verify that it is not altering the start buffer addresses or the screen attributes in the outbound data stream.

   c. Input from command lines results in incorrect input.

      1) Verify that the appropriate VTAM logmode entry for the logical unit was used. (See "Determining Device Characteristics" on page 135.)

      2) If the VSCS data exit (DTIPDDSI) is coded, verify that it is not altering the start buffer addresses or data in the input data stream.

8. Did the logo problem occur after receiving message DTIC13I?

   Check for messages DTIC15I and DTIC16I. If you receive either of these messages, correct the problem, update the logo, and try to log on again.

9. Did message DTIC11I appear?

   The DTIC11I message indicates that the CCS subcomponent of CP has requested that the global path be severed. VSCS should produce a dump.

10. Did the logo update fail?

    If you failed to update the logo, you should receive message DTIC14I, indicating why the update failed.

Go to "Reporting the Problem to IBM" on page 112.

Logon Problems

1. Issue the VTAM DISPLAY and VSCS DISPLAY ID=*netid.luname* commands and answer the following questions:
   - Does the session exist?
   - If the session exists, do the device characteristics match the logical unit characteristics?
   - If the RPL is busy, what is the current RPL command?

   See "General Procedures to Diagnose Hung Logical Unit Problems" on page 131 and "Determining Device Characteristics" on page 135 for more information.

2. Are you logging on to VSCS for the first time?

   If so, and if you are not using the VTAM-supplied logmode entries, the logmode entries are probably incorrect. If you are logging on from an MVS or VSE host, that host must provide customized logmode entries. Review the process for building your own MODETAB and logmode entries in "Session Parameters for Devices: Logon Mode Tables" in the *VTAM Network Implementation Guide* (VSCS requirements in the PSERVIC field).

3. Are you trying to log on in logo mode?

   When you turn on the power for your terminal, the VM logo appears on the screen. To log on, use one of the following methods:
   - Press ENTER and the PA1 or the PA2 key, and enter your logon information.
   - Enter the user ID and the password in the appropriate fields on the logo.
   - Enter the logon user ID in the command field on the logo.

   Note: If message DTIC15I is received during VSCS initialization, you cannot log on from the logo screen.

   If you cannot log on in logo mode, try logging on in no-logo mode. If the problem still exists, you are probably using the wrong logmode entry. (See "Determining Device Characteristics" on page 135.)

   If the problem goes away, a user-designed logo may be causing the problem. (See "VM Logo Problems on an SNA Device" on page 154.)

   To pursue the problem further, collect the following documentation:
   - VSCS trace for the failing logical unit with all options active except FRE and GET.
   - VTAM internal trace, OPT=API
   - VTAM buffer contents trace for the failing logical unit
   - Output from the VSCS DISPLAY ID=*netid.luname* command, if possible.

- Description of what is on the screen
- VSCS console log.

4. Did you get message DTIC11I indicating IUCV global path severed, or message DTIP14I with a return code indicating that the IUCV global path is severed?

   If so, get data from the VSCS CCS trace and any VSCS internal error dumps taken. Get the documentation if available, and then go to "Reporting the Problem to IBM" on page 112.

   If not, issue VMEXIT, which should return the logical unit to VTAM. Then issue the VSCS DISPLAY command. If the logical unit is not found, the problem is with VTAM.

5. Is the terminal hung?
   - If so, has the VM logo been removed from the screen?
     - If so, you need the following documentation:
       - VSCS trace for the failing logical unit with all options active except FRE and GET.
       - VTAM internal trace. Ensure that the API option is set (OPT=API).
       - VTAM buffer contents trace for the failing logical unit.
       - VSCS console log.
     - If not, go to step 3 on page 157.
   - If not, is the VM logo or VTAM USSMSG10 on the screen?
     - If so, you need the following documentation:
       - VSCS trace for the failing logical unit with all options active except FRE and GET.
       - VTAM internal trace. Ensure that the API option is set (OPT=API).
       - VTAM buffer contents trace for the failing logical unit.
       - VSCS console log.
     - If not, go to step 6.
6. Get other documentation.

   In addition to any documentation you have collected so far, get the VSCS CCS and VTAM exit trace for the failing logical unit.

7. Are all terminals hung?

   If so, go to "All VSCS LUs Are Hung or Slow (Affects Logons and Current Sessions)" on page 161.

8. Go to "Reporting the Problem to IBM" on page 112.

Logoff and Disconnect Problems

If the terminal is hung, go to page 172.

1. Record the status of the terminal. For example, on a display screen, note what is in the status area, the last input or the last output to the logical unit.
2. Issue the CP QUERY NAMES command and note whether the logical unit name is known to CP.
3. Issue the VTAM DISPLAY and VSCS DISPLAY ID=*netid.luname* commands and note the following:
   - Whether the session exists
   - What the current RPL command is
   - Whether the RPL is busy.

   See "General Procedures to Diagnose Hung Logical Unit Problems" on page 131 for more information.
4. Collect more documentation:
   - Device type, model, and current state (use VSCS DISPLAY ID=*netid.luname*)
   - CCS and VTAM trace output
   - Any other VSCS trace output available
   - VSCS dump using DUMPLU. If the LU is not found, use GDUMP.

   Note: The VSCS dump formatter will print a single logical unit on request.
5. Find the logical unit control blocks in the VSCS dump and look at the following:
   a. The current state of each logical unit. It will be PLB LBCSTATE or VLB LBCSTATE. *VTAM Data Areas for VM/ESA* describes these states in more detail.
   b. What is the status of the SEND RPL? Note whether it is busy, and note whether it contains any errors.
   c. Note whether the logical unit has any work to process. This is indicated by the existence of DTIWEBs for this LU in the dump.
6. After collecting all documentation, go to "Reporting the Problem to IBM" on page 112.

Wait and Loop Problems

Complete the preliminary procedure below, and if the problem persists, choose the symptom that most closely matches yours. Follow the procedure given for your symptom.

- "All VSCS LUs Are Hung or Slow (Affects Logons and Current Sessions)" on page 161
- "One or More Logical Units Hung" on page 163
- "VSCS Operator Commands Do Not Complete" on page 175
- VSCS is looping, see "Presentation Services Is in a Large Enabled Loop" on page 162.

If you still cannot identify the condition, continue with "Loop" on page 86 or "Wait" on page 67.

Note: The term "hung terminal" means the user is prevented by the system from entering data.

Preliminary Procedure for Problems Involving All Logical Units

This is a fairly short procedure that can help you isolate some common reasons that cause all logical units to hang.

1. Is VSCS dispatchable?

   Use the following commands to see if VSCS can process anything:

   - Does VSCS respond to the VSCS QUERY command?
   - Does "VSM" show up when the #CP QUERY NAMES command is issued?
   - Does VTAM respond to the VTAM DISPLAY commands?

   If the answer to all three questions is yes, VSCS is dispatchable and the global IUCV connection is intact. Continue with step 2.

   If one or more answers is no, the problem could be one of the following:

   - Higher-priority task is looping
   - VSCS is looping
   - VSCS (or GCS) is waiting for an IUCV instruction to complete.

   If the loop or wait occurs during initialization, go to "Initialization Problems" on page 143.

   If the problem occurs during termination, go to "Termination Problems" on page 150.

   If the problem occurs during logon, go to "Logon Problems" on page 157.

   If the problem occurs during logoff, go to "Logoff and Disconnect Problems" on page 159. Get a VSCS dump (see step 3), and go to "Reporting the Problem to IBM" on page 112.

   If the problem occurs after VSCS is up and running, go to "VSCS Operator Commands Do Not Complete" on page 175.

   Note: If a VSCS dump is in progress, it will appear that all users are hung and no console commands will complete. When the dump has completed, processing will resume.

2. Look at operator console output.

   The VSCS, VTAM, and CP console listings may contain messages that explain your present condition. Check for VSCS or VTAM messages that indicate that a network failure occurred and error recovery is in progress. For example, message DTIS31I might tell you that VSCS is taking a dump.

3. Get a VSCS dump and a VSCS internal trace.

   If a dump is not in progress and the VSCS console does not respond to commands issued in "running" status, use the #CP VMDUMP 0:END FORMAT GCS command in CP mode. For more information on the VSCS dump, see "VSCS Dump" on page 453.

To determine what book describes the CP operator commands for your operating system, see Table 80 on page 1165. If possible, obtain the VSCS internal trace before you recover the logical unit.

4. Examine the logical unit control blocks and trace output.

   In the dump, find the logical unit control blocks TAB, PLB, and VLB. The VSCS internal trace, used along with information from the control blocks, may show the reason for the hang.

   Note: The VTAM VARY INACT, VSCS FORCE, and CP FORCE commands cause bits to turn on and off in the PLB and VLB. Those bits are used to determine the reason for a hung logical unit. Therefore, attempt to gather documentation *before* using those commands, unless you suspect one of them is the cause of the problem.

5. Is it a network problem?

   A network problem, such as VTAM failing to terminate a session successfully, or outstanding I/O, may cause logical units to hang. To diagnose VTAM problems, see Chapter 2, "Collecting Documentation for Specific Types of Problems."

   Note: Before determining that the problem is a VSCS problem, issue the VSCS DISPLAY ID=*netid.luname* to determine whether I/O is active (RPL is busy).

All VSCS LUs Are Hung or Slow (Affects Logons and Current Sessions)

If all VSCS logical units are hung or experiencing slow response time, the problem is likely to be in one of the following areas:

Problem	Symptoms
Global IUCV path severed	• VSCS issues great quantities of DTI messages • Users cannot log on (DTIV12I) • Message DTIP14I • Message DTIC11I.
Presentation services in a large enabled loop	• Loop symptoms – LU response time is slower than normal – VSCS virtual machine is using a much larger amount of CPU (90 to 100%) – VSCS console does not respond • All VSCS trace entries are the same.
VSCS task in a wait in VTAM	• VSCS DISPLAY of LUs known to be hung shows SEND RPL IS BUSY
No VTAM RECEIVE ANY RPLs active	• Message DTIV05I • Message DTIV02I • VSCS dump shows inactive RPLs.

Determining whether VSCS is the cause of a loop can be difficult. The following can cause the loop:

- VTAM is recovering from a network failure (if VSCS is in the same virtual machine as VTAM).
- VSCS is in SCIP recovery after a network failure.

- SCIP recovery has occurred after a large network outage and the values of SCIPNUM and SCIPTIM are 0.
- GCS is dumping the virtual machine.
- A VTAM temporary storage shortage has occurred, and VSCS is trying the VTAM request again.
- VSCS, VTAM, or GCS may have a control block chained to itself, resulting in an infinite loop.
- Because of a loop in CCS or IUCV, the VSCS virtual machine appears to be looping but is not.

Note: Check the VSCS console log to see if error recovery is in progress.

Choose the problem most like yours, and follow the procedure for it.

Presentation Services Is in a Large Enabled Loop

1. Is VSCS showing symptoms of a loop?

   These symptoms are typical of a loop problem:

   - Extremely slow response time
   - 90 to 100% of CPU time used
   - VSCS console does not respond.

2. Do any VSCS trace entries indicate a loop?

   Look at the VSCS internal trace table on the screen or in a dump and see if all of the trace entries are the same. (See "VSCS Trace Facility" on page 458 for a description of the VSCS trace records). If so, the VSCS CCS trace entries could indicate if VSCS or CP is continuously sending IUCV requests or replies. The VSCS dispatcher trace entries show whether VSCS is in an internal loop processing a single work element.

   On the screen, use the VSCS QUERY command (see "VSCS QUERY Command" in *VTAM Operation*) to find the starting address of the VSCS internal trace table. Then display the address of the current entry with the command

   ```
   D xxxxxx.20
   ```

   where *xxxxxx* is the current trace entry (the VSCS trace table address +8 is the address of the current entry). Record what you see. Wait a minute or two and repeat the procedure above. If the VSCS trace entries are the same, and only the VSCS CCS trace is active, then VSCS is in a loop, VSCS is waiting for an event to complete, or there is currently no work to process.

   To find the internal trace table in a dump, use the pointer at SABTRHDR in the DTISAB. If VSCS is in a loop, the current and preceding trace entries may look the same, depending on the trace options that are active.

3. Find the current logical unit.

   You can find the logical unit being processed by presentation services in the current (last used) entry in the VSCS internal trace table.

4. Use the dispatcher trace to verify the loop.

If VSCS is looping on one work element block (DTIWEB), you will see it re-dispatched for the same logical unit over and over. For information on the dispatcher trace, see "VSCS Trace Facility" on page 458.

Note: If you limit the number of users on the system at the time of the loop, the extent of the loop may be more apparent in the save areas.

5. Examine the dispatcher work element queue in a VSCS dump.

   The presentation services dispatcher work element queue is located in module DTIISDA1 in a dump. The address of this module in a dump is shown in the VSCS trace table header and in the link-edit listing. If the queue shows a long chain of work to do, VSCS may be in a loop or a permanent wait. If so, the problem will be associated with the current user identified by the dispatcher.

6. Get the following documentation:
   - VSCS trace for the failing logical unit with all options active except FRE and GET.
   - VIT, specify OPT=(API,MSG,PIU), MODE=EXT.
   - A few seconds of virtual machine branch trace on the VSM, to trace only the VSCS address range.

     The beginning of the VSCS load module can be found by use of the VSCS QUERY command (if VSCS TRACE is active). Message DTIS74I contains the starting address of the VSCS trace table. Use CP DISPLAY Txxxxx.50 and *VTAM Data Areas for VM/ESA* to find the addresses in the THDRSTRT and THDRSDA2 fields. Those addresses represent the extent of the VSCS load module.

     Note: Use the CP TRACE command instead of the PER command to trace only address ranges of VSCS that you want to trace. PER is accepted only as an alias for the CP TRACE command. See Table 80 on page 1165 to determine what book describes the virtual machine branch trace.
   - VSCS dump
   - VSCS operator console listing.
7. Go to "Reporting the Problem to IBM" on page 112.

One or More Logical Units Hung

If multiple terminals are hung, you need to determine if all VSCS terminals are hung with only a few reporting, or the terminals are hung for different reasons. If you cannot log on to VSCS, you have a global hang and should use the procedure described in "All VSCS LUs Are Hung or Slow (Affects Logons and Current Sessions)" on page 161. If you can log on to VSCS, then follow the procedure for a single hung terminal for each terminal.

If a single terminal is hung, follow these steps:

1. Use the "General Procedures to Diagnose Hung Logical Unit Problems" on page 131.
2. Get the required documentation from "Getting the Required Documentation" on page 164.
3. Try to clear the hang.

- For local non-SNA VSCS terminals, press the RESET key.
- For remote VSCS terminals, do one of the following:

  a. Turn the power off, then on.

  b. Use the TEST ON/OFF switch.

  c. For SNA logical units, press the ALT and SYS REQ keys, then enter LOGOFF to VTAM.

  d. Press the ATTN key.

  e. Have the operator issue one or more of the following commands from the VM or VTAM console:

     - `VTAM TERMSESS`
     - `VTAM VARY INACT,FORCE,ID=`*`luname`*
     - `VTAM VARY INACT,IMMED,ID=`*`luname`*
     - `VSCS FORCE ID=`*`netid.luname`*
     - `CP FORCE` *`userid`*

Getting the Required Documentation

VSCS terminal hang conditions can be grouped into two categories: re-creatable hangs and non-re-creatable hangs. Each category requires specific procedures for gathering the documentation needed by IBM support personnel to identify and resolve the problem.

A re-creatable hang exists when you can anticipate a failure occurring and can begin gathering critical information. Usually, you can re-create the problem while running various traces. After recreating the problem, you can gather other pertinent documentation including dumps and logs.

A non-re-creatable hang exists when changes occur in the time, frequency, and LU associated with the problem. Variations in contributing conditions, therefore, do not allow you to anticipate the problem. For example, you may know that a certain hang problem occurs twice each day. However, you do not know when specifically or for which LU the problem will occur. Because of these uncertainties, you must run traces that capture information for all LUs running in the system. The large volume of data produced by this global tracing requires you to consider a number of factors.

The VSCS tracing facility has features to handle both re-creatable and non-re-creatable hangs. For re-creatable hangs, you can run VSCS traces externally to allow the output to be spooled while other traces in the GCS group are running (for example, the VTAM buffer trace). For non-re-creatable hangs, you can run VSCS traces internally and route output to a VSCS global trace table, or to each LU's individual trace table.

Gathering Trace Information for Re-creatable Hangs: For re-creatable problems, gather the trace information described in this section.

Note: You must activate the following system external tracing facilities to capture output from traces running in external mode:

- TRSAVE and TRSOURCE
- GCS ETRACE.

See "Collecting External Trace Data (VM)" on page 359 for more information on collecting, formatting, and viewing external trace output. Do *not* use ACF/TAP to format VSCS traces.

- VSCS trace output for the individual LU (*netid.luname*) having the problem. Specify all options except FRE and GET, as follows:

  `VSCS TRACEON netid.luname (CCS DISP VTAM DAT EXT`

- VTAM buffer contents trace for the individual LU. Specify the trace as follows:

  `VTAM MODIFY TRACE,TYPE=BUF,ID=luname`

- If you suspect the problem is in VTAM, or if the problem is very disruptive, run the VTAM internal trace (VIT) in external mode, as follows:

  `VTAM MODIFY TRACE,TYPE=VTAM,OPT=(API,MSG,PIU,PSS,SMS,SSCP),MODE=EXT`

  One indication of a VTAM problem is a VSCS display of message DTIS65I, showing that the RPL is BUSY.

  Notes:

  1. It is *normal* for message DTIS59I to indicate that "RECEIVE SPECIFIC RPL IS BUSY." Message DTIS59I is displayed only if you are using the RECEIVE SPECIFIC option of VSCS.
  2. The VIT traces *all* activity in VTAM. If possible, run the VIT when the system is not being used heavily.

After all necessary traces are active, re-create the problem. It is helpful to activate the traces before the LU actually logs onto the VSCS application. This allows the trace to contain the BIND. When activating a VTAM buffer trace, VTAM must know about the LU in the host where the trace is activated. However, if the LU is cross-domain or cross-network to this VTAM, then VTAM defines the LU as a cross-domain resource (CDRSC) at logon time. Unless it is explicitly defined in a CDRSC major node, a CDRSC is not known by the VTAM that it is logging onto. Consequently, VTAM does not allow you to start a buffer trace for the LU until the LU is logged onto the application program.

To capture the logon sequence in a trace, use the following techniques to enable a buffer trace to start *before* a CDRSC logs onto an application:

- Define the CDRSC in a CDRSC major node. (VTAM allows you to define a CDRSC major node while VTAM is running.)
  1. Create the major node file in CMS. (All major node files have a filetype of VTAMLST.)
  2. Link to the disk from the VTAM virtual machine using CP LINK.
  3. ACCESS the disk from the VTAM virtual machine.
  4. Issue VTAM VARY ACT,ID=*name command*.
- Use the VTAM function that "remembers" a CDRSC for a limited amount of time after it establishes a session with this host. The VTAM CDRSCTI start option specifies the amount of time VTAM retains dynamically defined CDRSCs after the last session with this host has terminated. The default is 8 minutes. Assuming the default value is in effect, you can log the LU onto VSCS, then log the LU off VSCS and have 8 minutes to activate the VTAM buffer trace and re-log the LU onto VSCS.

When you have re-created the problem, immediately stop all traces. Do not attempt any recovery until you have gathered all necessary documentation.

Gathering Dump Information for Re-creatable Hangs: For re-creatable problems gather the dump information described in this section:

- Unless the problem itself automatically causes a dump, obtain a dump of the VTAM virtual machine immediately after you re-create the problem.

  If the VSCS controlling this LU is not running in the same virtual machine as VTAM, obtain a dump of the virtual machine in which VSCS is running. To determine which VSCS virtual machine controls this LU, issue the following CP command:

  `#CP QUERY NAMES`

  QUERY NAMES lists the LUs controlled by a specific VTAM service machine (VSM). See Table 80 on page 1165 to determine which book contains more information on CP operator commands.

  To obtain a dump of VSCS without significantly impacting your system, issue the following command:

  `VSCS DUMPLU ID=`*`netid.luname`*

  The DUMPLU command executes quickly, but dumps only the areas of storage that actually pertain to the LU. Usually, the limited information supplied by DUMPLU is sufficient to diagnose a problem with an LU.

  If more information on the VSCS virtual machine is needed, you may need to obtain a full VSCS virtual machine dump. To obtain a full dump, issue the following CP command:

  `#CP VMDUMP 0-END FORMAT GCS DSS TO `*`userid`*

  Note: This VMDUMP command causes the entire virtual machine to stop while the dump is running.

- Collect the VTAM and VSCS, if separate, console logs.

In summary, use the following commands to activate traces and dumps for re-creatable hangs:

- `VSCS TRACEON`
- `VSCS TRACEON `*`netid.luname`*` (ALL EXT`
- `VTAM MODIFY TRACE`
- `VTAM MODIFY TRACE,TYPE=BUF,ID=`*`luname`*

  If VSCS is running in a different virtual machine than VTAM:

- `VSCS DUMPLU ID=`*`netid.luname`*

  If VSCS is running in the same virtual machine as VTAM:

- `#CP VMDUMP 0-END FORMAT GCS DSS TO `*`userid`*

The following command may be required, depending on the circumstances and severity of the hang condition:

- `VTAM MODIFY TRACE,TYPE=VTAM,OPT=(API,MSG,PIU,PSS,SMS,SSCP),MODE=EXT`

  For the VTAM virtual machine:

- `#CP VMDUMP 0-END FORMAT GCS DSS TO userid`

Gathering Trace Information for Non-re-creatable Hangs: In order for IBM support personnel to diagnose and correct your non-re-creatable hang problem, the VSCS trace should be active with the following options:

Option	Equivalent DTIGEN Operand
CCS	CSTRACE=Y
DISP	DPTRACE=Y
DAT	DTTRACE=Y
VTAM	VTTRACE=Y
ILU	ILUTRAC=Y

For best results, it is recommended that you specify these options using the DTIGEN macroinstruction instead of the VSCS TRACEON command.

When these trace options are active, VSCS records trace entries in the VSCS global trace table and in each individual LU's trace table. The ILU (ILUTRAC) option causes each LU logged onto VSCS to have its own trace table with entries specifically related to the LU. When an LU hang occurs, the LU's trace table should "hang" as well. Consequently, the events immediately preceding the hang are captured in the table. The activity of other LUs does not cause the hung LU's trace table to be overwritten (wrap).

The SIZE (ILUSIZE) trace option controls the size of the individual LU table. It is recommended that you do not change this value unless IBM support personnel request the change when you are working on a problem. The default SIZE is 40 entries, which allows a sufficient amount of information in the table without using an excessive amount of storage. Each entry is 32 bytes in size. Before you specify and activate the ILU and SIZE options, ensure that you have sufficient private storage available in the VSCS virtual machine. For example, assume your system has a maximum of 1000 users logged onto VSCS at one time. If you use the default size of 40 entries, the following amount of storage is dedicated to individual LU trace tables: 40+1 entries/table * 32 bytes/entry * 1000 users = 1250K (approximately 1.25 megabytes).

Note: 1 is added to the number of entries because the size of the individual LU trace table header is also 32 bytes.

Gathering Dump Information for Non-re-creatable Hangs: For non-re-creatable problems gather the dump information described in this section.

Note: You *must* take the dumps described in this section before you attempt recovery actions with commands such as VSCS FORCE, CP FORCE, or VTAM VARY INACT.

- If you suspect the problem is in VTAM, or if the problem is very disruptive, take a dump of the VTAM virtual machine immediately after the problem occurs. One indication of a VTAM problem is a VSCS display of message DTIS65I, showing that the RPL is BUSY.

  Note: It is *normal* for message DTIS59I to indicate that "RECEIVE SPECIFIC RPL IS BUSY." Message DTIS59I is displayed only if you are using the RECEIVE SPECIFIC option of VSCS.

  Take a dump of the controlling VSCS virtual machine as soon as possible after the problem occurs. If you already have a dump of the VTAM virtual machine

and the VSCS that controls this LU is running in the same virtual machine as VTAM, then you do not need an additional dump.

If the VSCS controlling this LU is not running in the same virtual machine as VTAM, you need to obtain a dump of the virtual machine in which VSCS is running. To determine which VSCS virtual machine controls this LU, issue the CP QUERY NAMES command. QUERY NAMES lists the LUs controlled by a specific VTAM service machine (VSM). See Table 80 on page 1165 to determine which book contains more information on CP operator commands.

To obtain a dump of VSCS without significantly impacting your system, issue the following command:

```
VSCS DUMPLU ID=netid.luname
```

The DUMPLU command executes quickly, but dumps only the areas of storage that actually pertain to the LU. Usually, the limited information supplied by DUMPLU is sufficient to diagnose a problem with an LU.

If you need more information on the VSCS virtual machine, you can obtain a full VSCS virtual machine dump. To obtain a full dump, issue the following CP command:

```
#CP VMDUMP 0-END FORMAT GCS DSS TO userid
```

Note: Using VMDUMP to dump the entire VTAM and VSCS virtual machines causes the virtual machines to stop while the dump is running. Consequently, VSCS and VTAM users are affected by VMDUMP.

- Collect the VTAM and VSCS, if separate, console logs.

In summary, use the following commands to activate traces and dumps for non-recreatable hangs:

- `VSCS TRACEON (CCS DISP VTAM DAT ILU`

  or the equivalent VSCS DTIGEN operands:

- `CSTRACE=Y,DPTRACE=Y,VTTRACE=Y,DTTRACE=Y,ILUTRAC=Y`
- `VSCS DUMPLU ID=netid.luname`

  or

- `#CP VMDUMP 0-END FORMAT GCS DSS TO userid`

Determine When the Hang Occurred

Choose from the following list the description that most closely matches your situation, and go to that procedure.

1. Logical unit hangs while end user is using a full-screen application program, such as SPF or XEDIT (see page 169).
2. Logical unit hangs while end user is in console mode or CMS mode (see page 170).
3. Logical unit hangs when end user switches between full-screen mode and console or CMS mode (see page 171).
4. Logical unit hangs after error condition during purge, logoff, or cleanup. This may be indicated by a message (see page 172).

5 Logical unit hangs after message DTIC10I because the path is already severed (see page 173).

6 Logical unit remains hung after operator issues VTAM VARY INACT, VSCS FORCE, or CP FORCE commands (see page 173).

7 Other (see page 174).

1 Logical Unit Hangs While You Are Using Full-Screen Mode or When the Logical Unit Is Dialed

1. Is a full-screen application program (such as XEDIT or SPF) responsible for the failure?

   In full-screen mode, the application program is responsible for formatting the screen, unlocking the keyboard, and reading input data. If the application program does not handle these functions properly, the device can be hung.

   Indications of this are:

   - PROGxxx message
   - Message limit is reached
   - Message 'NOT ACCEPTED' appears when you press the enter key. This indicates that the application issued a WRITE with keyboard restore, but did not retry the previous input data.
   - The keyboard is locked.

   If you have indications of a full-screen application program failure, get a dump of the user's virtual machine and go to "Reporting the Problem to IBM" on page 112.

2. Is the logmode defined incorrectly?

   Signs of this are:

   - Incorrect characters appear on the screen.
   - The colors are wrong.
   - The screen size is wrong.
   - The logical unit is disconnected.

   Use the VSCS command DISPLAY ID=*netid.luname* to see if the device is defined correctly. (See "Determining Device Characteristics" on page 135 for more information.) If the logmode is defined incorrectly, see "Logon Mode Table" in the *VTAM Resource Definition Reference* and correct the logmode definition.

   If the logmode is correct, continue with this procedure.

3. Did an OPCHECK occur?

   Signs of this are:

   - The application program ends.
   - The output for a color display shows the wrong colors.
   - The logical unit is disconnected.

   If OPCHECK conditions exist, get the VSCS CCS trace and VTAM buffer contents trace on the failing logical unit for documentation.

   Go to "Reporting the Problem to IBM" on page 112.

4. Is the logical unit hung or data lost after a CP message was displayed or input was entered?

   While a logical unit is in full-screen state, VSCS should display *no* CP messages (except warnings and priority messages) until input is received (at which time the full-screen application is notified of input available).

   If you coded DTIGEN option FSREAD=N, then you can expect loss of data:

   - If the screen is interrupted before input is received from the logical unit
   - If modified data input is available, CP messages were displayed, and the buffer data is requested by the full screen application.

   Note: The full-screen application program is responsible for restoring the keyboard after input is received. If the keyboard is not restored, the logical unit remains hung.

   Get the VSCS trace with CCS, DISP, and VTAM active, and a VSCS dump. (See "VSCS Trace Facility" on page 458 and "VSCS Dump" on page 453 for more information on getting this output.)

   Go to "Reporting the Problem to IBM" on page 112.

5. Is the logical unit still hung? If so, get this documentation:

   - VSCS CCS, DISP, and VTAM exit traces for the failing logical unit
   - VSCS dump of the logical unit using DUMPLU.
   - VTAM SMS (buffer use) trace for the failing logical unit.

   Go to "Reporting the Problem to IBM" on page 112.

2 Logical Unit Hangs While You Are Using Console Mode or CMS Mode

Note: The CMS full-screen command puts the logical unit into full-screen state.

1. Is the logical unit waiting for the previous command to be displayed again?

   If so, then one of the following occurred:

   - The command has not completed and CP did not notify VSCS (via command completion or timer expiration) to display the input again.
   - The command was executed by the full-screen application program and the application did not display the information again.
   - The input has not been accepted.

   Get a VSCS CCS trace for the failing logical unit and a VSCS dump. Then go to "Reporting the Problem to IBM" on page 112.

2. Is the logical unit hung after a 'MORE', 'HOLDING', or 'NOT ACCEPTED' condition?

   If so, use DUMPLU to get a VSCS dump of the logical unit and get the VSCS CCS and dispatcher traces. Then go to "Reporting the Problem to IBM" on page 112.

3. Are the VSCS data translation exits coded?

   The following VSCS user exits affect the logical units' reactions to any outbound data streams and CP's reaction to inbound data streams:

   DTIPDNTI LU1 inbound data
   DTIPDNTO LU1 outbound data

DTIPDDSI display input data
DTIPDDSO display outbound data.

If the problem does not appear to be a problem in the user-coded exit, get the following documentation:

- VSCS CCS trace
- VSCS DATA trace
- VTAM buffer contents trace for the failing LU.

Go to "Reporting the Problem to IBM" on page 112.

4. Is the problem with a 3279 color terminal (type 0 logical unit)?

   Make sure FEATUR2=EDATS is specified in one or more of these NCP definitions:

   LINE
   GROUP
   CLUSTER
   TERMINAL.

   Note: FEATUR2=EDATS is not supported for non-SNA devices attached through ICAs.

   Also, check the coding of the user exits (DTIPDDSI and DTIPDDSO).

   If color or highlighting still does not work correctly, get the following documentation:

   - VSCS CCS and VTAM exit traces, and data for the failing logical unit
   - VTAM buffer contents trace for the failing logical unit.

   Go to "Reporting the Problem to IBM" on page 112.

5. Is the device dialed to another system?

   If so, go to the full-screen procedure on page 169.

   Note: Dialing to PVM and passing through to another CMS system gives the appearance of normal console and CMS screens. However, VSCS receives the entire screen already formatted, including the status area, and processes the data as full-screen data.

6. Documentation for problems not described above:

   If you can re-create the problem, get:

   - VSCS trace for the failing logical unit with all options active except FRE and GET
   - VSCS dump of the logical unit using DUMPLU
   - VIT, specify OPT=(API,MSG,PIU,PSS,SSCP), MODE=EXT
   - VTAM buffer contents trace for failing logical unit.

   If you cannot re-create the problem, get:

   - VSCS trace with CCS active, for only the failing logical unit, if possible
   - VSCS dump of the logical unit using DUMPLU
   - VTAM logmode being used.

   Go to "Reporting the Problem to IBM" on page 112.

3 Logical Unit Hangs When You Switch Modes

1. Does your terminal hang when you switch between modes? For example:
   - Console mode to CMS mode
   - CMS mode to console mode
   - CMS or console mode to full-screen mode
   - Full-screen mode to CMS or console mode.
2. If so, get the following documentation:

   If you can re-create the problem, get:

   - VSCS dump of the logical unit using DUMPLU
   - VTAM buffer contents trace
   - VSCS CCS and dispatcher traces with just the failing logical unit traced
   - VIT, specify OPT=(API,MSG,PIU,PSS,SSCP), MODE=EXT.

   If you cannot re-create the problem, at least obtain the following:

   - VSCS dump of the logical unit using DUMPLU
   - VSCS trace with CCS and DISP active.
3. Go to "Reporting the Problem to IBM" on page 112.

4 Logical Unit Hangs During Logoff or Disconnect Processing

1. Have any messages been issued?

   Get the operator console log and look for messages that pertain to the hang.
2. Determine the component that sent the message.

   If the message has a return code, use Table 13 to determine the component that may have sent the message. Then look in *VTAM Messages and Codes* to see what the return code means.

   If the message has no return code, use the message prefix to determine where the message is coming from. Page 146 has a list of the message prefixes for each component.

Table 13. Source of Return Codes

Message	Source of Return Codes
DTIPxxx	• GCS • VM • VSCS
DTISxxx	• VM • VSCS
DTIVxxx	• VTAM

3. Are other components connected to the logical unit?

   Answering these questions may indicate which components are still communicating with the logical unit.

   - Does the VSCS command DISPLAY ID=*netid.luname* show the status of the device? (See "Determining Device Characteristics" on page 135 for more information.)
   - Is the user still logged on to VM? Try the CP QUERY NAMES command. If the logical unit name and user ID appear, the logical unit is connected to VM and the CP IUCV connection is intact. If the response is gibberish, the

logical unit is not connected to VM and the CP IUCV connection has been severed.

- Does VTAM show the logical unit still in session with VSCS? (Use the VTAM DISPLAY command.)

4. Note the status of the logical unit at this time. For example, on a display screen note what is in the status area, and what was the last input or the last output to the logical unit.
5. Collect documentation before recovery.

   If you can re-create the problem, get:

   - VSCS dump of the logical unit using DUMPLU
   - VTAM buffer contents trace
   - VSCS trace for the failing logical unit with all options active except FRE and GET.

   If you cannot re-create the problem, get:

   - VSCS dump of the logical unit using DUMPLU
   - VSCS trace with CCS active.

6. Recover the logical unit with one or more of these commands:

   - CP FORCE
   - VTAM VARY INACT
   - VSCS FORCE.

7. Go to "Reporting the Problem to IBM" on page 112.

5 Logical Unit Hangs After Receiving Message DTIC10I

1. Did these symptoms occur?

   Message DTIC10I indicates that an IUCV request or response was received on the global path after an IUCV SEVER was issued by VSCS. Most of the time this does not cause a problem, because CP logs off or disconnects the logical unit when the IUCV SEVER is processed. If it is a problem, it could produce these symptoms:

   - Abend
   - VSCS message DTII14I
   - Loss of operator communications
   - Message GCTIXT413T indicates that the signal services path is severed.

2. Collect documentation.

   - VSCS dump of the logical unit using DUMPLU
   - VSCS trace with CCS active, running for all logical units.

3. Go to "Reporting the Problem to IBM" on page 112.

6 Logical Unit Stays Hung after You Use VARY INACT or FORCE Command

1. Try to use both the VTAM VARY INACT and CP FORCE commands. If you issue the CP FORCE command and the resource is unknown, CP might be waiting for an IUCV reply before completing the termination process for the logical unit. Try to use the VTAM VARY INACT command.

   If the logical unit is still hung, continue with step 2.

2. Determine which components are still connected to the logical unit.

- Use the VSCS command DISPLAY ID=*netid.luname* to determine the status of the logical unit and of the SEND RPL.
- Is the user still logged on to VM?
- Does the logical unit name appear when you use the CP QUERY NAMES command? If so, the VM connection has not been severed.
- Does VTAM show the logical unit still in session with VSCS? Use the VTAM DISPLAY command to determine this.

From this information, try to determine which components are connected to the logical unit.

3. Note the status of the logical unit at this time. For example, on a display screen, what is in the status area, and what was the last input or the last output to the logical unit?

   Note: If the user logged off and the screen says "HOLDING," you must clear the screen before the logical unit can be disconnected from VM. (This is normal for VSCS.)

4. Collect documentation.

   If you can re-create the problem, get:

   - VSCS dump of the logical unit using DUMPLU
   - VSCS operator console listing
   - VTAM buffer contents trace
   - VSCS trace for the failing logical unit, running externally, with all options active except FRE and GET.
   - VIT, specify OPT=(API,MSG,PIU,PSS,SSCP), MODE=EXT.

   If you cannot re-create the problem, get:

   - VSCS dump of the logical unit using DUMPLU
   - VSCS operator console listing
   - VSCS CCS and dispatcher trace.

5. Go to "Reporting the Problem to IBM" on page 112.

7 Logical Unit is Hung for Other Reasons

1. Answer these questions.

   a. Issue the CP QUERY NAMES command on the CP console. Does the logical unit name appear?

   b. Is VSCS still controlling the logical unit? (Use the VTAM DISPLAY command to discover this.)

   c. Have any messages appeared on the VTAM or VSCS operator console for this logical unit?

   d. What is the last sequence of messages issued? (See the VSCS operator console listing.)

   e. Issue VSCS DISPLAY ID=*netid.luname*. Is the RPL busy?

2. Collect documentation.

   If you can re-create the problem, get:

- VSCS dump of the logical unit using DUMPLU
- VSCS operator console listing
- VTAM buffer contents trace
- VSCS traces for the failing logical unit, running externally, with all options active except FRE and GET.
- VIT, specify OPT=(API,MSG,PIU,PSS,SSCP), MODE=EXT.

If you cannot re-create the problem, get:

- VSCS dump of the logical unit using DUMPLU
- VSCS operator console listing
- VSCS CCS and dispatcher trace.

3. Go to "Reporting the Problem to IBM" on page 112.

VSCS Operator Commands Do Not Complete

1. Has the VSCS utility services subtask abended?

   Check the operator console listing to see if the utility services subtask abnormally terminated. If this occurred, VSCS can do no further processing of operator commands. Use the VSCS CANCEL command to terminate VSCS, then get the dump generated by the abend and go to "Termination Problems" on page 150.

2. Does a GCS message indicate a problem with the GCS console task? If that task abended or could not get storage, operator commands cannot reach VSCS.

3. If the problem is not an abend or GCS console task, try these commands:

   - If VSCS is running in the same virtual machine as VTAM, enter a VTAM command such as DISPLAY LU. Does it complete?
     - If yes, VSCS may be in a loop or a permanent wait.
     - If no, the problem is in the VTAM virtual machine. Continue down this list to determine where the problem is.
   - Enter the CP QUERY TIME command. This shows whether GCS is processing and passing CP commands to CP.
   - Enter #CP QUERY NAMES. This shows whether CP commands are being processed.

4. Does GCS process some commands, but not others? Does the PA1 key put the terminal in CP READ mode?

   If the answer to both questions is yes, GCS is looping or waiting for an event to complete. In CP READ mode, issue SET RUN OFF. Then enter the CP command D PSW to display the current PSW. Display the storage around the address shown in the second half of the PSW. If the last instruction issued was X'B2F0xxxx', CP has not completed an IUCV request, and VSCS and the virtual machine are permanently waiting for a response. If the problem does not involve an IUCV instruction, check to see if the WAIT bit is on in the PSW. If so, the problem is in the VSCS virtual machine. Get a VSCS dump and go to "Reporting the Problem to IBM" on page 112. The VSCS dump is described in "VSCS Dump" on page 453.

Message Problems: Message DTIS99I Indicates a VSCS Internal Error

For messages other than DTIS99I, go to "Message Problem" on page 93 in Chapter 2, "Collecting Documentation for Specific Types of Problems."

Message DTIS99I tells you that a VSCS internal error occurred for module DTI*xxxxx* and a number (*n*). The number identifies the reason the message was issued. Use the lists below to determine which component of VSCS issued the message.

Note: If an LU is involved with the internal error, the LU name receives the DTIS99I message. See the following example:

```
DTIS99I INTERNAL VSCS ERROR – DTIPRSVR 1
```

This indicates that VSCS attempted to issue an IUCV SEVER for the VSCS global path. For this error, a dump is taken (if dumping is enabled) and the SEVER request is ignored.

1. Presentation services issues messages with this information:

Module	n	Error Description
DTIPDADV	1	SEND-in-progress bit error
DTIPDFSS	1	Full screen request DTIWEB not valid
DTIPICPY	1	SEND-in-progress bit error
DTIPIFSI	1	SEND-in-progress bit error
DTIPIQRY	1	WSF query reply contained a character-sets structured field with a descriptor length (DL) of 0
	2	WSF query reply contained an implicit-partition structured field with a length of 0
	3	WSF query reply contained a structured field with a length of 0
DTIPISYN	1	SEND/REPLY WEB type is not valid
DTIPOCWR	1	SEND-in-progress bit error
DTIPOFWR	1	SEND-in-progress bit error
DTIPOWIC	1	SEND-in-progress bit error
DTIPOWRT	1	SEND-in-progress bit error
	2	SEND-in-progress bit error
DTIPRACT	1	Received a WEB that is not valid
DTIPRLGO	1	SEND-in-progress bit error
DTIPRSVR	1	Attempting to issue a SEVER for global path ID
DTIPRWEB	1	WEB is not valid
DTIPSCFS	1	FS response WEB is not valid
DTIPSCPY	1	SEND-in-progress bit error
	2	SEND-in-progress bit error
DTIPSFCR	1	SEND-in-progress bit error
DTIPSPIN	1	SEND-in-progress bit error
DTIPSTAM	1	Queue tag error for work element queue
	2	WEBFUN or WEBMODE or LBCSTATE is not valid
	3	Return code received is not valid

2. VTAM services issues messages with this information:

[2] APAR VM51150: DTIPIPUT does not issue message DTIS99I for error conditions.

Module	n	Error Description
DTIVCLSX	1	Schedule request to DTIVTRME failed
	2	No storage available
DTIVLOFF	1	IUCV connection exists
DTIVPURG	1	Schedule failed
DTIVRECA	1	Error in building WEB
DTIVRECX	1	More than 16 zero-length middle-in-chain RU's were received for this LUNAME in one chain
	2	Input chain is not valid. RPLONLY, RPLFIRST, RPLMIDLE, and RPLLAST were all off
DTIVREJR	1	Logical unit not in termination
DTIVSIML	1	Schedule of WEBNVCLU to VTAM Services failed after SIMLOGON retry count was exhausted
DTIVSEND	1	Request not valid
	2	RPL busy and recovery not in progress
	3	Recovery queue empty during recovery
DTIVSNDX	1	RPL response is not valid
	2	Dequeue error in recovery
	3	Data response not valid in recovery
	4	RPL response is not valid
	5	Recovery queue empty during recovery
	6	Recovery queue empty during recovery
	7	Attempting to schedule DTICMT
DTIVSTAM	1	Queue tag error for work element queue
	2	WEBFUN or WEBMODE or LBCSTATE is not valid
	3	Return code received is not valid

3. System services issues messages with this information:

Module	n	Error Description
DTISCHED	1	Process identifier is not valid
	2	Queue tag header is incorrect
DTISDEQU	1	Queue tag does not match queue header
DTISDSTO	1	Logical unit's VS PIBSTAT was not PIBRUN after dispatch of last work element.
	2	Logical unit's PS PIBSTAT was not PIBRUN after dispatch of last work element.
DTISENQU	1	Chain pointer is not valid or queue tag does not match
	2	Pending queue depth exceeded
DTISFREB	1	SMP tag is not valid
	2	Request is not valid (releasing storage that is already free)
DTISFREM	1	SMP tag is not valid
DTISGETB	1	Control block error or request is too big to obtain
	2	Free block count is not valid
	3	No storage obtained
DTISLDMP	1	Internally requested partial dump failed with other than return code 4. Full dump taken.
DTISGMGR	1	PDBUSCNT (chain pointer) is not valid

4. Documentation to collect for VSCS internal error problems:
   - Issuer of the message: presentation services, VTAM services, or system services
   - Reason the message was issued

- VSCS dump (produced automatically as a result of message DTIS99I)
- VSCS operator console listing
- Any VSCS traces that were active at the time of the error.

Incorrect Output Problems

1. Describe the incorrect output. The following list shows some of the possible symptoms:
   - The size of the display screen is incorrect.
   - The status area of the display screen does not appear as expected.
   - Highlighting or color does not work as expected.
   - Characters that are not valid appear in the output, such as EBCDIC, APL, TEXT, or double-byte kanji characters. (Characters that are not valid are any characters that are obviously wrong or that you do not expect.)
   - Input data is lost.
   - PROGxxxx occurs on SNA 3270s. If this happens during full-screen mode, the contents of the data stream may not be valid. If the terminal is not in full-screen mode, this message is more likely to indicate a VSCS problem or a logmode problem.
   - The device does not function like a VM-supported terminal and the problem is not due to a known restriction. See "Operational Differences between VSCS and CP" on page 139 for additional information.
   - The incorrect output does not involve full-screen data or the VM logo, and one or more of the following symptoms occur:
     - Data is misplaced on the screen.
     - Data is missing.
     - Content of data is incorrect.
     - Device errors occur or the logical unit is disconnected when you execute a command or a program.
2. Check to see that product restrictions are not causing the incorrect output. See "Operational Differences between VSCS and CP" on page 139 for additional information.
3. Does the incorrect output involve the logo? See "VM Logo Problems on an SNA Device" on page 154 for more information.

   Note: If you enter both the user ID and the password on the user ID line, you may receive a different message (HCP) than you do if you enter only the user ID.
4. Is the device defined correctly to VSCS?

   Use the VSCS command DISPLAY ID=*netid.luname* to determine if the device is defined correctly to VSCS. If the DISPLAY command is not successful, get a dump of VSCS. The logical unit control blocks in a dump also show the LU definition.

   If the LU definitions are not correct for the device, correct them and retest the logical unit. See "LU Group Major Node" in the *VTAM Resource Definition Reference* for more information on defining logical units.

5. Check to see if the appropriate VTAM logmode for the device is being used. See "Logon Mode Table" in the *VTAM Resource Definition Reference* for more information on logmodes.
6. Are user translation exits involved?

   If your system has user-written translation exits, try to re-create the problem without them. Use the DTIGEN parameters or the VSCS CHANGE command to disable the user exits and see if the problem goes away.
7. Is kanji support involved?

   Double-byte character support for kanji does limited editing. Check the data stream sent to see if it is within the limitations of double-byte support.

   Note: VSCS does not check whether the device supports kanji. The application has this responsibility.
8. Are TWX 3101 configuration switches set up as documented by VM? To determine what book describes how TWX 3101 configuration switches should be set up, see the "terminal configuration switches" entry in Table 80 on page 1165.
9. Other possible causes of incorrect output are listed here. You may choose to investigate these further.
   - Functional limitation due to the release of VM used
   - Hardware problem
   - VSCS problem
   - VM problem
   - VM application problem.
10. If the problem has not been solved, collect the following documentation:
    - For status area problems, lost input, device errors, and disconnections:
      - VSCS dispatcher trace of the failing logical unit
      - VSCS trace with CCS active, of the failing logical unit
      - VSCS VTAM exit trace of the failing logical unit
      - VTAM buffer contents trace.
    - For all other incorrect output problems:
      - VTAM buffer contents trace of the failing logical unit
      - VSCS trace with CCS active, of the failing logical unit
      - If a VSCS dump is necessary, limit the number of users on the system and use the VSCS BFRFIFO command before re-creating the problem.

Storage-Related Problems

Issue the VSCS STORAGE command whenever you suspect that storage is causing problems.

1. Determine whether the virtual machine in which VSCS resides is large enough. Estimate this using the *Estimating Storage for VTAM* diskette.

Notes:

a. If VSCS resides in the same virtual machine as VTAM, add the storage requirements for VTAM.

b. If either DTIGEN start option STCHKTM or STRELTM is not greater than 0, you do not get VSCS storage pool information.

If the virtual machine is large enough for your environment, continue. Otherwise, correct the VTAM virtual machine size.

2. If you are using new VSCS start options, evaluate changes made with DTIGEN parameters:

   - Trace table size
   - Increase in the number of printers
   - Receive buffer size
   - BLKMULT greater than one
   - VSAMLM number increased
   - Transmit buffer sizes for both logical units and displays
   - RECEIVE SPECIFIC option
   - Exception response option.

   Note: The VSCS DISPLAY OPTIONS and the VSCS STORAGE commands display the majority of the DTIGEN parameters and start options that are currently used by VSCS.

   If VSCS ran properly before these changes, review the size of the VSCS virtual machine before reporting a problem.

   You can use the VSCS STORAGE command to see:

   - Current number of VTAM sessions and CP connections
   - Number of RECEIVE ANY RPLs being used
   - Indicator of storage shortage in message DTIS15I
   - Number of GETMAIN failures
   - Some of the DTIGEN options that are currently active
   - Storage management parameters that have been specified
   - Number of VEIBs allocated (controlled by VSAMLM).

   If VSCS is running in the same virtual machine as VTAM, use the VTAM DISPLAY BFRUSE command to determine VTAM's storage usage.

3. Is the network in slowdown or recovery mode?

   If the network is in slowdown mode, it takes additional storage to maintain data destined for the logical unit. This symptom could also be caused by one of the following:

   - VTAM is recovering from a network failure.
   - VSCS is in SCIP recovery after a network failure.
   - VSCS is in a different virtual machine than VTAM, and a dump is being taken for VSCS or VTAM.
   - A VTAM temporary storage shortage has occurred.
   - The I/O buffer pool has expanded for a VTAM-controlled device.

4. Are several logical units hung?

The storage being used by hung logical units remains allocated until the logical unit is released or terminated. If you are using the exception response mode and RECEIVE SPECIFIC options or the ILU trace option, the amount of storage used could be enough to cause a storage shortage. See "General Procedures to Diagnose Hung Logical Unit Problems" on page 131 before continuing.

If you can re-create the problem or if it is occurring frequently, gather the following documentation:

- VSCS internal trace with options CCS, DISP, FRE, and GET active
- Dump of the VSCS virtual machine.

5. Did you receive message DTIS99I for DTISFREB or DTISGETB?

   This indicates that VSCS is having a problem managing storage blocks.

   If you can re-create the problem or if it is occurring frequently, gather the following documentation:

   - VSCS internal trace with the options CCS, DISP, FRE, and GET active
   - Dump created when message DTIS99I is issued.

   Get a VSCS dump and go to "Reporting the Problem to IBM" on page 112.

6. Did you receive message DTIS10I, followed by DTIV12I?

   This sequence of messages indicates that either the virtual machine is too small, or all available storage has been depleted. If the VSCS virtual machine is large enough, the available storage may be allocated to a different VSCS storage pool than the current storage requester.

   For example, if storage is available in the presentation services pool, but VTAM services needs it, then GCS cannot provide additional storage. Logons are rejected until presentation services releases the storage.

   You can control the distribution of dynamic storage by coding the DTIGEN start options STCHKTM and STRELTM to allow swapping of storage segments from one pool to a different pool, or to return them to GCS.

   Note: You can use the CHANGE command to alter the DTIGEN storage options that control redistribution.

   Get the following documentation:

   - Before the problem occurs, start the VSCS trace with CCS, DISP, FRE, and GET options active for all logical units. If possible, run the VSCS trace in external mode.
   - VSCS dump.

   Go to "Reporting the Problem to IBM" on page 112.

7. Is it an IUCV pacing problem?

   The values specified in the DTIGEN start options (DPACE for displays and KPACE for keyboards/printers) control IUCV pacing.

   For displays, the default value is the screen depth, which is also the default if the pace is set to 0. The screen depth varies by display type.

   For keyboard/printers, the default is 10. If the pace is set to 0 for keyboard/printers, there is no IUCV pacing.

Specifying no IUCV pacing can cause increased IUCV traffic, which can cause a VSCS storage shortage if VSCS is running in a large network or in a small virtual machine.

For TWX logical units, the default is 1, and cannot be changed in VSCS.

If IUCV pacing is not set properly, change the DTIGEN start option and reinitialize VSCS with the new DTIGEN. Use the VSCS CHANGE command to alter the pace parameters temporarily. If the problem persists, get the following documentation:

- Before the problem occurs, start the VSCS trace with CCS active.
- VSCS dump.

Go to "Reporting the Problem to IBM" on page 112.

Note: Use the VSCS DISPLAY OPTIONS command to determine the current values of DTIGEN DPACE and KPACE start options.

8. Other

   If the problem cannot be isolated from previous steps, then you must take a dump of the VSCS virtual machine. After obtaining the dump, use the VSCS dump formatter to print the VSCS storage control blocks. If VSCS is in the same virtual machine as VTAM, then also evaluate the VTAM storage.

   Get the output from the VSCS formatter and the console log, and go to "Reporting the Problem to IBM" on page 112.

Performance Problems

1. Do not use a large value for the DTIGEN start option VSAMLM. This can cause excessive paging in a medium-to-large environment. See Chapter 5, "VM/SNA Console Support (VSCS)" in the *VTAM Resource Definition Reference* for more information on VSAMLM, and see the *Estimating Storage for VTAM* diskette to determine whether the virtual machine size is correct.

   Note: If VSCS is running in the same virtual machine as VTAM, include the VTAM storage size required when you determine the virtual machine size.

   One possible solution is to move VSCS to a separate virtual machine or to add a second VSCS virtual machine to share the work load. It is more costly to run VSCS in a separate virtual machine, but it helps prevent wasted time due to page faults.

   If you think your performance problem is related to paging and storage (for example, you are getting slow response time for operator commands or logical units), collect this documentation:

   - VSCS dump
   - DTIGEN parameters
   - GCS internal trace for GETMAIN and FREEMAIN.

   Note: Use the VSCS DISPLAY OPTIONS command and VSCS STORAGE command to determine what most of the DTIGEN initialization parameters are currently set to.

2. If you think your performance problem is related to logical units (for example, you are getting slow response time for I/O operations), collect this documentation:

- VTAM I/O trace
- VTAM buffer contents trace
- VSCS trace for the failing logical unit, running externally, with all options active except FRE and GET
- GCS internal trace for SIO and I/O.

3. If you do not know the cause of the performance problem, collect all of the documentation in the two lists above, plus supply the values for MAXBFRU, UNITSZ, and IOBUF.
4. When you have collected the necessary documentation, go to "Reporting the Problem to IBM" on page 112.

Other VSCS Problems

The following topics are covered in this section:

CHANGE Command Not Working

The VSCS CHANGE command allows the operator to alter many of the VSCS start options dynamically. Dynamic alteration may not always provide the expected results. Consider the following before assuming there is a problem with the CHANGE command:

- Timer values:

  Alteration of a timer value takes effect immediately. For example, if STCHKTM=3600 (one hour) and you change it to STCHKTM=60 (one minute), the previous timer is cancelled.

- Buffer values:

  Alteration of a buffer value takes effect the next time the changed buffer is allocated.

- LU logon-related values:

  Alteration of values used during the logon or connection to CP does not take effect until another LU logs on to VSCS or CP.

- Recovery values:

  Alteration of recovery values takes effect immediately.

- User exits:

  Alteration of the user exit options (DEXIT and KEXIT) takes effect immediately.

Consider what the option you are changing is used for before determining that the CHANGE command did not work correctly.

One other consideration is the buffer values set for DPXMTL, KPXMTL and RCVBFRL. When set with the CHANGE command, the value displayed with the DISPLAY OPTIONS command matches what was entered. However, when you

code the values on the DTIGEN macroinstruction, VSCS optimizes the values to utilize the maximum amount of storage available for the size specified. The optimization performed does not increase the storage required for those options. For more information on the operands that can be modified with the VSCS CHANGE command see "VSCS CHANGE Command" in *VTAM Operation*.

SNA Dial Problems

SNA dial keeps the logical unit in full-screen state from the time the dial is complete until the dial is dropped or reset. All of the information on the screen is supplied by the resource that was dialed.

1. If the screen size changed after one of the following occurred, take the indicated action:

   a. A VTAM logical unit dialed another virtual machine.

      Make sure the virtual machine you are dialing supports the features provided by your logical unit.

   b. A VTAM logical unit, dialing PVM, dialed another GCS group or host containing VTAM.

      When dialing PVM, the logical unit is defined as a logical address, which may prevent the real device characteristics from being known. If you are dialing another VTAM machine, supply a logmode name that describes the type of terminal you are using.

   If the screen size is still incorrect, collect the following documentation:

   - VSCS trace with CCS active
   - VSCS dump
   - VTAM buffer contents trace (in both hosts if connecting to another VTAM machine).

2. Is the logical unit hung?

   If you are using PVM, find out whether the resource you are attempting to dial is active. An inactive resource could result in a hung logical unit. If the resource is active and the logical unit is hung, the resource, PVM, VSCS, CP, or GCS may be experiencing problems. Note whether other PVM users' terminals are also hung.

   If you cannot determine the reason, get the following documentation:

   - VSCS dump
   - VSCS trace with CCS active (must be active before the hang occurs).

   In the VSCS dump, find the logical unit control blocks and the last few trace entries for that specific logical unit. From this, answer the following questions:

   a. Is the PLB LBCSTATE of the logical unit 20, 40 or 60?
   b. Is the SEND RPL busy, indicating VSCS is waiting for a response?
   c. Did VSCS respond to all outstanding requests?

   If the situation can be re-created, get the following documentation:

   - VSCS dispatcher and VTAM exit traces
   - VTAM internal trace, OPT=API
   - VSCS dump.

Go to "Reporting the Problem to IBM" on page 112.

3. Is the I/O interrupt missing?

   If the situation is caused by missing I/O, run these traces externally on just the failing logical unit:

   - VSCS CCS and VTAM exit traces
   - VIT, specify OPT=(CIO), MODE=EXT
   - CP SIO, SIOF, and I/O traces
   - GCS SIO and I/O traces.

     To determine what book describes the CP and GCS traces for your operating system, see Table 80 on page 1165.

   Go to "Reporting the Problem to IBM" on page 112.

Printer Sharing Problems

1. Is VSCS printer sharing failing?

   If you coded a DTIGEN macroinstruction with PRTSHR=N, do the following:

   a. Change it to PRTSHR=Y.
   b. Reassemble DTIUSER.
   c. Link DTIUSER to the VSCS LOADLIB.
   d. Reinitialize VSCS with the correct DTIUSER number.

   Note: You can change PRTSHR with the CHANGE command.

   Use the VSCS DISPLAY PRTTAB command to determine the status of PRTSHR and to determine which display (if any) is associated with each printer.

   If you are using default parameters or you are using a DTIUSER with PRTSHR=Y coded, get the following documentation:

   - VSCS dispatcher and VTAM exit traces
   - VSCS dump.

   Go to "Reporting the Problem to IBM" on page 112.

2. Does VTAM SIMLOGON fail after you use the VSCS PRINTER command and receive a COPY function for the defined printer?

   If the VTAM SIMLOGON fails, then make sure:

   - The printer name was spelled correctly
   - The resource is known and available to VTAM (use VTAM DISPLAY command)
   - The VTAM logmode for the printer logical unit is correct.

3. Does "NOT AVAILABLE" appear in the status area of the display?

   In order for VSCS to access a VTAM printer, either the VSCS PRINTER command or a VTAM VARY ACT command must be issued first. See "VSCS PRINTER Command" and "VARY ACT Command" in *VTAM Operation* for more information on commands.

   If the proper command was not issued or the printer name was misspelled, issue the command again with the correct name. Use the VSCS FORCE ID= command to remove the incorrect printer name from the printer table.

If the proper command was issued to identify the printer to VSCS, get the following documentation:

- VSCS dispatcher and VTAM exit traces
- VTAM internal trace, OPT=API
- VTAM buffer contents trace for the printer logical unit
- VSCS dump.

Go to "Reporting the Problem to IBM" on page 112.

4. Is VSCS not releasing a printer when requested to?

   If the printer was allocated to VSCS by using the VTAM VARY ACT,LOGON=VSCS command, then a VTAM VARY INACT command is the only way to release it.

   If the VTAM SIMLOGON is used (check the VSCS operator console listing), VSCS uses the time value specified on the DTIGEN TIMEREL start option to determine when to release the printer.

   Note: Make sure that SIMLOGON is for an LU printer device. See Chapter 5, "VM/SNA Console Support (VSCS)" in the *VTAM Resource Definition Reference* for more information on TIMEREL.

   If the correct value for TIMEREL is specified and VSCS was initialized with the proper DTIUSER module, get the following documentation:

   - VSCS dispatcher and VTAM exit traces
   - VSCS dump
   - VTAM internal trace, OPT=API
   - VTAM buffer contents trace for the printer logical unit before the release was requested.

   Go to "Reporting the Problem to IBM" on page 112.

5. Other printer problems:

   For problems not described above, get the following documentation:

   - VSCS dispatcher and VTAM exit traces
   - VTAM internal trace, OPT=API
   - VSCS dump.

   Go to "Reporting the Problem to IBM" on page 112.

Operator Commands Are Rejected

Does VSCS reject a command that was entered correctly and issue an error message? You may have entered the command at an inappropriate time. For example, a VSCS START command is rejected when VSCS is already active. Try using some other commands to see if they work.

Note: Avoid using VSCS HALT, CANCEL, or QUIT; they could cause further problems.

If the problem is unique to one command, or to a certain sequence of commands, get the operator console listing and go to "Reporting the Problem to IBM" on page 112.

Part 2. Diagnostic Procedures

Chapter 5. Using DISPLAY and MODIFY Operator Commands

You can control and monitor the VTAM program network with start options and operator commands. These commands include the following:

- DISPLAY commands
  - "Using VTAM DISPLAY Commands for Problem Determination" on page 194
  - "Display Buffer Pool Use" on page 195
  - "Display NCP Storage" on page 196
  - "Display Resource Status" on page 196
  - "Display Path Tables" on page 197
  - "Display Resources in a Pending State" on page 197
  - "Display Route Status" on page 198
  - "Display Route Test" on page 205.
  - "Display Traces" on page 208
- MODIFY commands
  - "Using VTAM MODIFY Commands for Problem Determination" on page 209
  - "Modify Input/Output Problem Determination" on page 209
  - "Modify Message Module Identification" on page 210
  - "Modify NCP Intensive Mode Recording" on page 210
  - "Modify SDLC Link Level 2 Test" on page 211
  - "Modify Tuning Statistics" on page 212
  - "MODIFY VTAMOPTS Command for Changing Start Option Values" on page 212

For information on VTAM start options, see Chapter 4, "Start Options" in the *VTAM Resource Definition Reference*.

For information on commands, see Chapter 2, "VTAM Operator Commands" in *VTAM Operation*.

Note: You can also use the NetView program to monitor and collect error statistics from the VTAM network. *Planning for NetView, NCP, and VTAM* describes this licensed program.

Using VTAM DISPLAY Commands for Problem Determination

VTAM provides DISPLAY (D) commands to show status and other information about network resources. The following list shows what kind of information is displayed for each of the VTAM DISPLAY commands. For more information on the syntax and output of these commands, see Chapter 2, "VTAM Operator Commands" in *VTAM Operation*.

Command	Information Displayed
D ADJCLUST	Adjacent cluster table definitions in the current ADJCLUST table
D ADJCP	Status of adjacent CP major nodes
D ADJSSCPS	Adjacent SSCP tables
D APPLS	Status of application program major and minor nodes
D BFRUSE	VTAM buffer usage
D BNCOSMAP	Native and nonnative COS mapping defined for a border node
D CDRMS	Status of cross-domain resource manager major and minor nodes
D CDRSCS	Status of cross-domain resources (including independent LUs)
D CLSTRS	Status of clusters (PUs in NCP, local SNA, and switched major nodes)
D CNOS	Change-number-of-sessions characteristics for LU 6.2 application programs
D CONVID	Conversations with LU 6.2 application programs
D COS	Class-of-service table names
D CPS VSE	Contents of the call progress signal (CPS) table
D DIRECTRY	Information maintained by central directory server
D DISK	Disk contents of 3720 or 3745 Communication Controller
D DLURS	All dependent LU requesters for which this host acts as Dependent LU Server
D EXIT	Status of user-written exit routines
D GROUPS	Status of line groups
D ID	Individual major or minor nodes
D LINES	Status of lines and channel links
D LMTBL	LU-mode table for LU 6.2 application programs
D LUGROUPS	LUGROUP major nodes, model LU groups, and model LUs
D MAJNODES	Status of major nodes
D MODELS	Model PUs and LUs
D NCPSTOR	Storage contents of 3720 or 3745 Communication Controller
D NETSRVR	Network node server information
D PATHS	Dial-out path information
D PATHTAB	Status of explicit routes and virtual routes
D PENDING	Resources in a pending state

D ROUTE	Status of explicit routes and virtual routes; existence of routes; whether a route is operational
D RSCLIST	Resources whose names match a particular pattern
D SESSIONS	Session status information
D STATIONS	Status of cross-subarea link stations
D STATS	Storage information for use by the *Estimating Storage for VTAM* diskette
D STORUSE	Storage usage for storage pools and MVS VSE data spaces
D TABLE	Table type, use count, and users
D TERMS	Status of device-type LUs (terminals)
D TGPS	Transmission group profiles
D TOPO	Topology of APPN network (information about nodes and transmission groups)
D TRACES	Status of VTAM and NCP traces
D TRL	Information about the TRL major node or about a single TRLE definition statement
D TSOUSER MVS	Status of a TSO user ID
D USERVAR	USERVARs and the application programs associated with them
D VTAMOPTS	Start options

Display Buffer Pool Use

You can use the DISPLAY BFRUSE command to display information about buffer use. In response to this command, VTAM indicates that the display is for buffer use and issues a series of messages that contain monitoring information. For each buffer pool, this information includes:

- Buffer pool ID.
- Flags (Q or F): Q shows a request is queued for this pool; F shows dynamic buffering has failed for this pool.
- Size of each buffer in this pool.
- Current total number of buffers in this pool.
- Current count of buffers available (the number not in use).
- Largest number of buffers this pool has expanded to at any time.
- Largest number of buffers in use at any time.
- Cumulative count of the number of times each buffer pool has expanded.
- Expansion and contraction thresholds.
- The expansion increment (the number of buffers to be added to a buffer pool during dynamic expansion).
- VTAM's intermediate routing node buffer use limit (IRNLIMIT), current buffer use, and maximum buffer use.
- MVS VM VTAM's CSA buffer use limit (CSALIMIT), current buffer use, and maximum buffer use.
- MVS VM Maximum amount of CSA in use since VTAM was started.

- VSE VTAM's SGA buffer use limit (SGALIMIT), current buffer use, and maximum buffer use.
- Current amount of VTAM private storage and maximum amount of VTAM private storage.

If the DISPLAY BFRUSE command is used while an SMS (buffer use) trace is running, the fields MAX TOTAL, MAX USED, and TIMES EXP reflect buffer usage only since the last trace record was written, because the SMS trace resets these fields. For more information about the syntax and output of the DISPLAY BFRUSE command, see "DISPLAY BFRUSE Command" in *VTAM Operation*.

DISPLAY BFRUSE output can help you identify possible sources of problems. The following chart shows some problem symptoms and the corresponding buffers to check in SMS trace output:

For this symptom:	Check this buffer pool:
I/O hang	IOBUF
Session failure	CRPLBUF and LPBUF
VTAM hang	LPBUF

For information on interpreting DISPLAY BFRUSE output for each type of buffer pool, see "Monitoring Common Storage Areas" in the *VTAM Network Implementation Guide*.

Display NCP Storage

You can display any selected portion of NCP storage by using a DISPLAY NCPSTOR command. Up to 256 bytes can be displayed for each command. For the 3720 or 3745 communication controller, you can also use the DISPLAY NCPSTOR command to display up to 256 bytes of an NCP dump or state vector stored on the disk.

VTAM issues message IST245I to display the storage contents.

Notes:

1. The NCP storage to be displayed may change while it is being formatted for transmission.
2. You cannot display NCP storage at a data host, because a data host does not own any NCPs.

For more information on the syntax and output of the DISPLAY NCPSTOR command, see "DISPLAY NCPSTOR Command" in *VTAM Operation*.

Display Resource Status

You can use the DISPLAY ID command to display status information about any major or minor node. For example, a request to display a physical unit or a logical unit indicates whether that node has been added by dynamic reconfiguration. DISPLAY ID also indicates whether a logical unit, a physical unit, or a link is supported by the Network Terminal Option (NTO).

Note: From a data host you cannot display the status of either an NCP or the NCP's subordinate resources, because a data host does not own any NCPs.

Two types of node status are displayed when you use the DISPLAY ID command:

The STATUS field shows the *current state* of the node
The DESIRED STATE field shows the *desired state.*

The desired state is the condition that VTAM processing is attempting to establish for the node. Previously entered operator commands or recovery processing may establish the desired state for a node. When processing is completed, the desired state and the current state should be the same. See "Resource State Code Values" in *VTAM Messages and Codes* for descriptions of the resource status codes used in response to the DISPLAY ID command.

If SNA network interconnection is in use, the DISPLAY command shows:

- The network ID associated with a resource (if any)
- For a cross-network CDRSC, the real resource name
- For a cross-network CDRM, the real name of the gateway node through which the SSCP-SSCP session passes, and the network address as known in the requesting host's network.

Display Path Tables

This is the place to start when route problems are detected. This display provides information about the route status known by this host VTAM.

You can use the DISPLAY PATHTAB command to display the status of explicit routes and their associated virtual routes for a local host. You can display information about all routes or you can limit the information using the ADJSUB and DESTSUB operands. The resulting display shows the host path table contents.

For more information on the syntax and output of the DISPLAY PATHTAB command, see "DISPLAY PATHTAB Command" in *VTAM Operation.*

Display Resources in a Pending State

You can use the DISPLAY PENDING command to display information about resources in the domain that are in one of the following pending states:

- Transient state to or from a fully active state
- State of "recovery pending" or "recovery in progress" for application programs suspended because of the failure or takeover of an application program enabled for persistence. You **must** enter the DISPLAY PENDING command on the system in which the application program resides.

The resource can be a major node, a minor node, an application program, a physical unit, or a logical unit.

For more information on the syntax and output of the DISPLAY PENDING command see, "DISPLAY PENDING Command" in *VTAM Operation.*

Display Route Status

The DISPLAY ROUTE command shows the status and availability of virtual and explicit routes. VTAM displays the status of selected routes and, if TEST=YES is specified, does a route test on the routes selected in the DISPLAY command. (See "Display Route Test" on page 205.) The VTAM operator may select the origin of the routes to be displayed or tested. The origin may be either a host processor or an NCP.

For more information on the syntax and output of the DISPLAY ROUTE command, see "DISPLAY ROUTE Command" in *VTAM Operation*.

For a sample path problem, see "Example: Solving Path Problems" on page 199.

The display of status for the routes selected is formatted as shown in Figure 23.

```
Displaying one explicit route to a destination subarea:
d net,route,destsub=01,netid=netc,origin=a03n43a,er=5
IST097I DISPLAY ACCEPTED
IST535I ROUTE DISPLAY 7 FROM SA 4 TO SA 1
IST808I ORIGIN PU = C0453LE DEST PU = C01NPU NETID = NETC
IST536I VR  TP    STATUS    ER       ADJSUB  TGN  STATUS  CUR MIN MAX
IST537I  0   0    ACTIV      5            1    1  ACTIV3
IST537I  0   1    INACT      5            1    1  ACTIV3
IST537I  0   2    INACT      5            1    1  ACTIV3
IST314I END
```

Figure 23. Example of DISPLAY Route Status Output

MVS VM If you are using SNA network interconnection, the DISPLAY ROUTE command can be used to show the status and availability of adjacent VTAM networks. The resulting display is the same as shown in the previous example, except that message IST808I contains an additional field, NETID=*netid*, to show the ID of the adjacent network.

See *VTAM Messages and Codes* for a complete description of the variable data contained in the messages that result from the DISPLAY ROUTE command.

It is possible for the test results for an explicit route to be lost before they are displayed. For example, if a node or a link along the explicit route fails between the time the test request flows outbound and the time the test results flow inbound, the results will be lost. If this occurs, re-enter the DISPLAY ROUTE command for that explicit route.

Note: You can use the NetView session monitor to collect more information about routes. If all the required session monitors along the route are in session, you can test the entire route, from one session end to the other. See Table 80 on page 1165 to determine what book has more information on the NetView session monitor.

Example: Solving Path Problems

This example takes you through a sample path problem and show you how to diagnose and solve the problem. It includes the following sections:

Rules for Routing: The rules that you need to keep in mind when working with routes are:

- Virtual routes must end in the subareas where the session end points reside.
- Virtual route numbers must be defined the same in both directions but only at each end point. A virtual route definition does not need to be specified in every node in the path.
- Explicit routes do not have to flow in both directions, but must be the same in one direction from endpoint subarea to endpoint subarea.
- Explicit and virtual route rules apply in each network, not across SNI network boundaries.
- Transmission group numbers on the VR that you are using must be the same in both directions, but only between two nodes, not along the entire path.

Configuration and Situation: Following is the configuration:

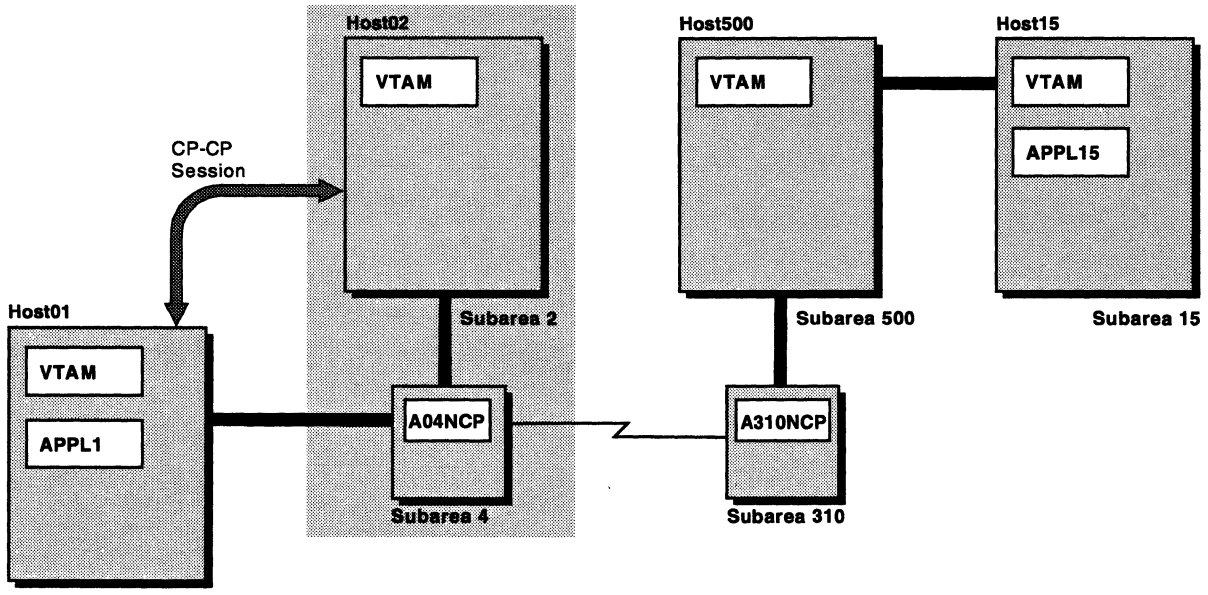


Figure 24. Path Problem Example Network Configuration

The situation is:

- Host01 is a network node and has a CP-CP session with Host02.
- Host15 and Host500 are subarea data hosts.
- You need a session from APPL1 to APPL15.

- The session is failing with a path problem sense code (8013xxxx).

Analyzing the Problem: Since Host01 is an APPN node, the route does not end in Host01; it ends in subarea 4 (A04NCP). The route must follow the path: subarea 4 (A04NCP) to 310 (A310NCP) to 500 (Host500) to 15 (Host15) and back.

Following the rules for routing, you need the following for this session:

- One virtual route number going from Host500 to A04NCP.
- One explicit route number going from Host15 to A04NCP.
- One explicit route number going from A04NCP to Host15.

Finding the Problem: To find the problem, you must first display the routes across your session path. Then, you need to map the information that you received from the displays to locate the problem.

Steps for Displaying Routes: You can display the routes between the subareas in your routes to identify the problem area.

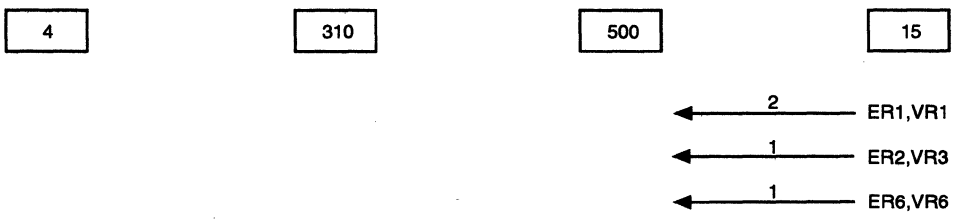
Note: Displays shown are abbreviated. If you run this display on your system, it will list information for all explicit routes from 0 to 15.

1. Display the route from Host15 to Host500 with A04NCP as the destination subarea. From Host15, issue the following command:

   ```
   D NET,ROUTE,DESTSUB=4
   ```

   ```
   IST097I DISPLAY ACCEPTED
   IST535I ROUTE DISPLAY 1 FROM SA 15 TO SA 4 225
   IST808I ORIGIN PU = ISTPUS DEST PU = A04NCP NETID = NETA
   IST536I VR  TP    STATUS   ER       ADJSUB  TGN  STATUS  CUR MIN MAX
   IST537I                     0            3    1  INOP
   IST537I  1   0    ACTIV    1          500    2  ACTIV3    7   5  15
   IST537I  1   1    INACT    1          500    2  ACTIV3
   IST537I  1   2    ACTIV    1          500    2  ACTIV3   12   5  15
   IST537I  3   0    INACT    2          500    1  INOP
   IST537I  5   0    INACT    3           30    1  INOP
   IST537I  5   1    INACT    3           30    1  INOP
   IST537I  5   2    ACTIV    3           30    1  INOP
   IST537I  6   0    INACT    6          500    1  ACTIV3
   IST537I  6   1    INACT    6          500    1  ACTIV3
   IST537I  6   2    ACTIV    6          500    1  ACTIV3   23  15  45
   IST537I                    15                  UNDEF
   IST314I END
   ```

   From subarea 15 to subarea 500, you have ER1, ER2, and ER6 defined.

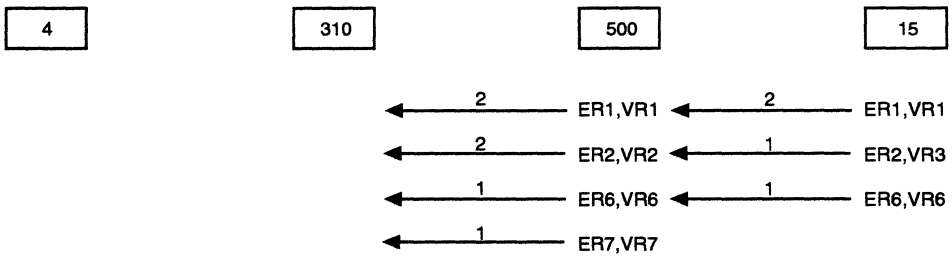


2. Display the route from Host500 to A310NCP with A04NCP as the destination subarea. From Host500, issue the following command:

```
D NET,ROUTE,DESTSUB=4
```

```
IST097I DISPLAY ACCEPTED
IST535I ROUTE DISPLAY 1 FROM SA 500 TO SA 4
IST808I ORIGIN PU = ISTPUS DEST PU = A04NCP NETID = NETA
IST536I VR  TP    STATUS   ER       ADJSUB  TGN  STATUS  CUR MIN MAX
IST537I                     0            2    1  INOP
IST537I  1   0    ACTIV     1          310    2  ACTIV3   20  15  45
IST537I  1   1    INACT     1          310    2  ACTIV3
IST537I  1   2    ACTIV     1          310    2  ACTIV3   33  15  45
IST537I  2   0    ACTIV     2          310    2  ACTIV3    9   5  15
IST537I  6   0    INACT     6          310    1  ACTIV1
IST537I  6   1    INACT     6          310    1  ACTIV1
IST537I  6   2    ACTIV     6          310    1  ACTIV3    7   5  15
IST537I  7   0    INACT     7          310    1  INOP
IST537I                    15                  UNDEF
IST314I END
```

From subarea 500 to subarea 310, you have ER1, ER2, ER6, and ER7 defined.

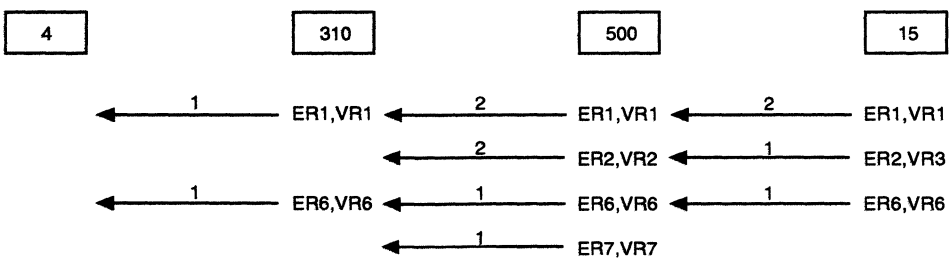


3. Display the route from A310NCP to A04NCP. From Host500, issue the following command:

```
D NET,ROUTE,ORIGIN=A310NCP,DESTSUB=04
```

```
IST097I DISPLAY ACCEPTED
IST535I ROUTE DISPLAY 1 FROM SA 310 TO SA 4 225
IST808I ORIGIN PU = A310NCP DEST PU = A04NCP NETID = NETA
IST536I VR  TP    STATUS   ER       ADJSUB  TGN  STATUS  CUR MIN MAX
IST537I                     0            2    1  INOP
IST537I  1   0    INACT     1            4    1  INOP
IST537I  6   0    INACT     6            4    1  ACTIV3
IST537I  6   1    INACT     6            4    1  ACTIV3
IST537I  6   2    ACTIV     6            4    1  ACTIV3    3   2   6
IST537I                    15                  UNDEF
IST314I END
```

From subarea 310 to subarea 4, you have ER1 and ER6 defined.



So, you can use either ER1 or ER6 to go from subarea 15 to subarea 4.

4. Display the route from A04NCP to A310NCP with Host15 as the destination subarea. From Host02, issue the following command:

   D NET,ROUTE,ORIGIN=A04NCP,DESTSUB=15

```
IST097I DISPLAY ACCEPTED
IST535I ROUTE DISPLAY 1 FROM SA 4 TO SA 15 225
IST808I ORIGIN PU = A04NCP DEST PU = A15PU NETID = NETA
IST536I VR  TP    STATUS   ER       ADJSUB  TGN  STATUS  CUR MIN MAX
IST537I                     0            2    1  INOP
IST537I                     2               UNDEF
IST537I  7   0    INACT    7          310    1  ACTIV3
IST537I  7   1    INACT    7          310    1  ACTIV3
IST537I  7   2    ACTIV    7          310    1  ACTIV3    3   2   6
IST537I                    15               UNDEF
IST314I END
```

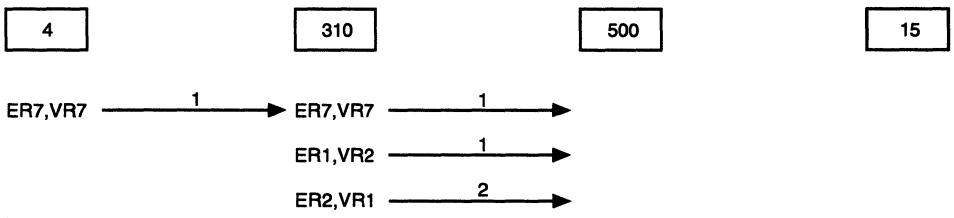
   From subarea 4 to subarea 310, you have ER7 defined.

5. Display the route from A310NCP to Host500 with Host15 as the destination subarea. From Host500, issue the following command:

   D NET,ROUTE,ORIGIN=A310NCP,DESTSUB=15

```
IST097I DISPLAY ACCEPTED
IST535I ROUTE DISPLAY 1 FROM SA 310 TO SA 15 225
IST808I ORIGIN PU = A310NCP DEST PU = A15PU NETID = NETA
IST536I VR  TP    STATUS   ER       ADJSUB  TGN  STATUS  CUR MIN MAX
IST537I                     0            2    1  INOP
IST537I  1   1    INACT    2          500    2  INOP
IST537I  2   0    INACT    1          500    1  ACTIV3   20  15  45
IST537I  7   0    ACTIV    7          500    1  ACTIV3   29  20  60
IST537I  7   1    ACTIV    7          500    1  ACTIV3
IST537I  7   2    ACTIV    7          500    1  ACTIV3   40  20  60
IST537I                    15               UNDEF
IST314I END
```

   From subarea 310 to subarea 500, you have ER1, ER2, and ER7 defined.

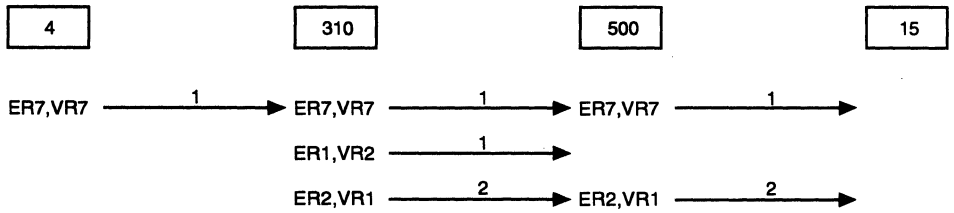


6. Display the route from Host500 to Host15. From Host500, issue the following command:

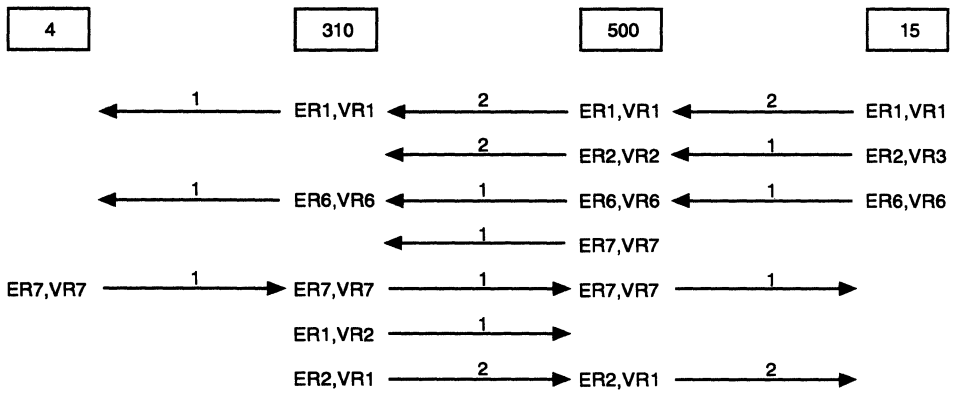
```
D NET,ROUTE,DESTSUB=15

IST097I DISPLAY ACCEPTED
IST535I ROUTE DISPLAY 1 FROM SA 500 TO SA 15 225
IST808I ORIGIN PU = ISTPUS DEST PU = A15PU NETID = NETA
IST536I VR  TP    STATUS   ER       ADJSUB  TGN  STATUS  CUR MIN MAX
IST537I                     0            2    1  INOP
IST537I  1   1    INACT    2           15    2  INOP
IST537I  4   0    INACT    5          310    1  INOP
IST537I  4   1    INACT    5          310    1  INOP
IST537I  4   2    INACT    5          310    1  INOP
IST537I  7   0    INACT    7           15    1  ACTIV3    7   5  15
IST537I  7   1    INACT    7           15    1  ACTIV3
IST537I  7   2    ACTIV    7           15    1  ACTIV3   35  20  60
IST537I                    15                  UNDEF
IST314I END
```

From subarea 500 to subarea 15, you have ER2 and ER7 defined.



Fixing the Problem: Here is what your routing looks like now:



From these examples, you can see that the easiest way to fix the problem is to define ER2 from subarea 4 to subarea 310 and map VR1 to ER2. You can do this dynamically using dynamic path update, or you can change the NCP generation.

Dynamic Path Update: To fix this problem using dynamic path update, define ER2 and map VR1 to ER2.

Warning: Whenever you are making a change in paths, make sure that you are not deleting a path that you need for another route. Following is the NCPPATH statement that defines the new path.

```
A04NCP   NCPPATH NETID=NETA
P002     PATH DESTSA=310,ER2=(310,1),VR1=2
```

To change the path, use the VARY ACT command to activate your dynamic path update member.

Note: If this was a problem from a host to another subarea, you would use a VPATH definition to fix the problem

Coding the Path in the NCP: In the NCP generation, find the path for destination subarea 310:

```
PATH DESTSA=310,
      ER0=(2,1),ER1=(71,80),
      ER3=(2,1),ER4=(3,80),ER5=(310,80),
      ER6=(310,80),ER7=(310,1),ER8=(71,80),
      ER9=(2,1),ER10=(1,1),ER11=(400,80),
      ER12=(1,1),
      VR0=6,
      VRPWS00=(1,3),VRPWS01=(1,3),VRPWS02=(1,3),
      VR1=9,
      VRPWS10=(2,6),VRPWS11=(2,6),VRPWS12=(2,6),
      VR2=3,
      VRPWS20=(2,6),VRPWS21=(2,6),VRPWS22=(2,6),
      VR3=8,
      VRPWS30=(2,6),VRPWS31=(2,6),VRPWS32=(2,6),
      VR4=4,
      VRPWS40=(2,6),VRPWS41=(2,6),VRPWS42=(2,6),
      VR5=11,
      VRPWS50=(2,6),VRPWS51=(2,6),VRPWS52=(2,6),
      VR6=10,
      VRPWS60=(3,9),VRPWS61=(3,9),VRPWS62=(3,9),
      VR7=7,
      VRPWS70=(3,9),VRPWS71=(3,9),VRPWS72=(3,9)
```

Change the definition for ER2 to destination subarea 310 and TGN1 and map VR1 to ER2:

```
PATH DESTSA=310,
      ER0=(2,1),ER1=(71,80),ER2=(310,1),
      ER3=(2,1),ER4=(3,80),ER5=(310,80),
      ER6=(310,80),ER7=(310,1),ER8=(71,80),
      ER9=(2,1),ER10=(1,1),ER11=(400,80),
      ER12=(1,1),
      VR0=6,
      VRPWS00=(1,3),VRPWS01=(1,3),VRPWS02=(1,3),
      VR1=2
      VRPWS10=(2,6),VRPWS11=(2,6),VRPWS12=(2,6),
      VR2=3,
      VRPWS20=(2,6),VRPWS21=(2,6),VRPWS22=(2,6),
      VR3=8,
      VRPWS30=(2,6),VRPWS31=(2,6),VRPWS32=(2,6),
      VR4=4,
      VRPWS40=(2,6),VRPWS41=(2,6),VRPWS42=(2,6),
      VR5=11,
      VRPWS50=(2,6),VRPWS51=(2,6),VRPWS52=(2,6),
      VR6=10,
      VRPWS60=(3,9),VRPWS61=(3,9),VRPWS62=(3,9),
      VR7=7,
      VRPWS70=(3,9),VRPWS71=(3,9),VRPWS72=(3,9)
```

Warning: Whenever you are making a change in paths, make sure that you are not deleting a path that you need for another route.

Display Route Test

If a route test was requested, results of the test are sent asynchronously to the console of the operator requesting the display. If the route test failed, the results are also sent to the console of the host that owns the rejecting subarea node. If the host owning the rejecting subarea is the same host that initiated the route test, that host will receive the test results twice.

To be tested, the explicit route must be known to VTAM. This means the explicit route must be defined to VTAM, or at some time must have been operative.

Successful Route Test

If TEST=YES is set and the route test is successful, the following asynchronous messages follow the route status display messages previously described.

The test results are formatted as shown in Figure 25.

```
IST538I   ROUTE TEST ### IN PROGRESS
IST533I   ER n SUCCEEDED IN ROUTE TEST ###
IST797I              FROM   VIA     ADJACENT       DEST   ER LENGTH
IST644I          ffffffff    TG     aaaaaaaa   dddddddd
IST534I               sss     t          xxx        yyy           1
IST798I              nnnn
```

Figure 25. Output of a Successful Route Test

In this example,

- *ffffffff* is the name of the origin physical unit.
- *aaaaaaaa* is the adjacent node.
- *dddddddd* is the name of the destination physical unit.
- *sss* is the subarea number of *ffffffff*.
- *t* is the transmission group number.
- *xxx* is the subarea number of *aaaaaaaa*.
- *yyy* is the subarea of *dddddddd*.
- 1 is the explicit route length.
- *nnnn* is the network ID of the node being displayed.

Failed Route Test

If the explicit route test fails because VTAM is unable to send the Explicit Route Test RU into the network, a message tells why the test cannot be performed. This message is shown in the following example.

```
IST510I   ROUTE TEST ### FAILED - reason
```

If the explicit route test is initiated by VTAM but fails, the messages in Figure 26 on page 206 show the reason for the test failure.

```
IST533I   ER 0 FAILED IN ROUTE TEST 8
IST797I             FROM   VIA     ADJACENT       DEST   ER LENGTH
IST644I           ffffffff   TG     aaaaaaaa   dddddddd
IST534I                sss    t          xxx        yyy
IST798I               nnnn
IST572I          REJECTING   TG     ADJACENT              ER MASK
IST816I                rrr    g          zzz                 mmmm
IST523I     <ER NOT DEFINED>
            <A REQUIRED TG IS INACTIVE>
            <ER NOT REVERSIBLE>
            <ER EXCEEDS MAXIMUM LENGTH>
            <MIGRATION ER NOT SUPPORTED>
            <MIGRATION NODE DOES NOT SUPPORT THIS ER>
            <MIGRATION NODE ENCOUNTERED>
            <UNEXPECTED TYPE BYTE X'##'>
```

Figure 26. Output of a Failed Route Test

In this example,

- *ffffffff* is the name of the physical unit which originated the ER_TEST.
- *aaaaaaaa* is the adjacent physical unit.
- *dddddddd* is the name of the destination physical unit.
- *sss* is the subarea number of *ffffffff*.
- *t* is the transmission group number.
- *xxx* is the subarea number of *aaaaaaaa*.
- *yyy* is the subarea of *dddddddd*.
- *nnnn* is the network ID of the node being displayed.
- *rrr* is the rejecting subarea.
- *g* is the transmission group number.
- *zzz* is the adjacent subarea.
- *mmmm* is the explicit route mask.

Location of Failure in a Route Test

The variable text in message IST523I can help you determine which direction the route test was going when it failed. There are three possibilities. Either the failure is in the adjacent subarea or in the link from the adjacent subarea to the rejecting subarea, or the failure is in the rejecting subarea, or the location of the failure could not be determined. If the reason is "UNEXPECTED TYPE BYTE X'##'," then the location of the failure could not be determined. This condition should not occur.

If the reason is:

"A REQUIRED TG IS INACTIVE"
"MIGRATION ER NOT SUPPORTED"
"MIGRATION NODE DOES NOT SUPPORT THIS ER"
"MIGRATION NODE ENCOUNTERED"

the adjacent subarea *follows* the rejecting subarea in the route being tested. Therefore, the problem is in the adjacent subarea or the link to the adjacent subarea from the rejecting subarea. (See Figure 27 on page 207.)

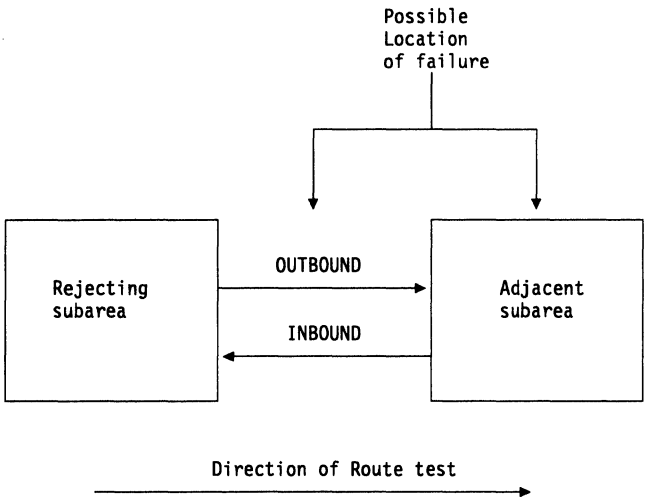


Figure 27. Route Test Failure (TG Inactive or Migration)

If the reason is "ER NOT REVERSIBLE," "ER EXCEEDS MAXIMUM LENGTH," or "ER NOT DEFINED," the adjacent subarea *precedes* the rejecting node in the route being tested. (See Figure 28.)

Check to see if the problem is a path definition error. If not, it may be a VTAM error.

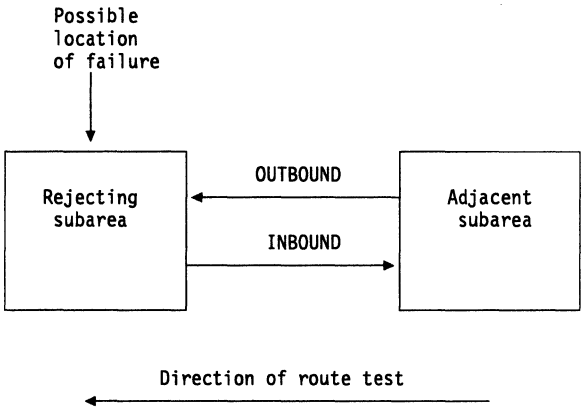


Figure 28. Route Test Failure (ER Not Reversible, Exceeds Maximum Length, or Not Defined)

The ER MASK field indicates which ER numbers the rejecting subarea can use to send data back to the host that requested the test. (These explicit routes are called *reverse explicit routes.*) For example, an ER MASK field of hex 88 (binary 10001000) means that the 0 and 4 bits of the mask are turned on, so ER numbers 0 and 4 can be used to send data back to the host that requested the test. If message IST523I says an explicit route is "NOT REVERSIBLE," that means the rejecting subarea does not have the reverse explicit routes in its path definitions.

It is possible for the test results for an explicit route to be lost before they are displayed. For example, if a node or link along the explicit route fails between the time VTAM sends the test request and the time VTAM receives the test results, the

results will be lost. If this occurs, re-enter the DISPLAY ROUTE command for that explicit route.

See *VTAM Messages and Codes* for complete explanations of the messages resulting from the DISPLAY ROUTE command.

Display Traces

Use the DISPLAY TRACES command to display the status of a trace for a resource and its subordinate nodes. DISPLAY TRACES applies to the following types of traces:

BUF	Buffer contents trace
CNM MVS	Communication network management trace
GPT	Generalized PIU trace
IO	Input/output trace
LINE	NCP line trace
NETCTLR	3710 network controller trace
SIT	Scanner interface trace
SMS	Storage management services buffer use trace
TG	Transmission group trace
TSO MVS	TSO user ID trace
VTAM	VTAM internal trace (VIT)

See "DISPLAY TRACES Command" in *VTAM Operation* for information on the syntax and output of the DISPLAY TRACES command.

Using VTAM MODIFY Commands for Problem Determination

This section includes the following information:

See *VTAM Operation* for additional information on VTAM commands.

Modify Input/Output Problem Determination

The input/output problem determination (IOPD) facility detects pending I/O requests when VTAM has sent a request to another part of the network and no response has been received after a certain period of time (determined by the IOINT start option). The IOPD facility issues IST530I or IST1278I, IST1051I, and IST1062I for each operation that has been pending longer than the specified time interval.

- These messages include a 10-character I/O operation identifier (usually an RU name or operator command), the names of the origin and destination nodes, the name of the node for which I/O is pending, and the event ID and event code associated with the I/O operation.

  See the explanation of message IST530I or IST1278I in *VTAM Messages and Codes* for a description of these messages. For more information on event codes and IDs, see Chapter 18, "Wait State Event Codes and IDs" in *VTAM Messages and Codes*.

- These messages are only an indication that a problem *may* exist. The longer an operation remains pending (for example, the more messages issued for the same request unit), the more likely it is that a problem exists. See "Wait" on page 67 for more information on identifying pending I/O problems.

To enable the IOPD facility, use the MODIFY IOPD command, the IOINT start option, or the MODIFY VTAMOPTS command. The value of the IOINT start option can be changed using the MODIFY VTAMOPTS command.

- For more information on this command, see "MODIFY VTAMOPTS Command for Changing Start Option Values" on page 212.
- For information on the MODIFY IOPD command, see "MODIFY IOPD Command" in *VTAM Operation*.
- For information on the IOINT start option, see "IOINT" in the *VTAM Resource Definition Reference*.

Note: During initialization of a large VTAM network, you may see more pending I/O operations than usual. If you are using the IOPD facility to track I/O problems during initialization, the number of message groups issued can degrade your network's performance.

You can use the IOMSGLIM start option to instruct the IOPD facility to write just one message group for each *type* of pending I/O operation, rather than one group for *each* operation. The resulting reduction in the number of

messages issued can improve your network's performance during initialization.

See "IOMSGLIM" in the *VTAM Resource Definition Reference* for more information on the IOMSGLIM start option.

Modify Message Module Identification

You can choose to include in VTAM messages the last five characters of the VTAM module that issued the message. The module name abbreviation appears between the message ID and the beginning of the message text. To insert or delete module name abbreviations, use the MODIFY MSGMOD command, the MSGMOD start option, or the MODIFY VTAMOPTS command.

Specifying MSGMOD=NO stops the insertion of module IDs into messages. MSGMOD=NO is the default. The issuing module name does not appear in messages unless MSGMOD=YES is specified in either one of the MODIFY commands or in the MSGMOD start option.

- For more information on the MODIFY MSGMOD command, see "MODIFY MSGMOD Command" in *VTAM Operation*.
- For more information on the MSGMOD start option, see "MSGMOD" in the *VTAM Resource Definition Reference*.
- For more information on the MODIFY VTAMOPTS command, see "MODIFY VTAMOPTS Command for Changing Start Option Values" on page 212.

For example, VTAM message xxxxx (where xxxxx is an operating system-unique message number) normally appears as:

```
xxxxx   DUMP OF ncpname COMPLETE
```

If you specify MSGMOD=YES in either the MODIFY MSGMOD command or the MSGMOD start option, the message appears as:

```
xxxxx   INFXI   DUMP OF ncpname COMPLETE
```

Notes:

1. Any message that exceeds the maximum message length after the insertion of the module ID is truncated.
2. If your installation has changed the message text and omitted the message ID, the module name is the first item in the message.

Modify NCP Intensive Mode Recording

When you receive a large number of temporary errors for a line, but not enough to create a permanent error, you can use intensive mode recording to find the cause of the error. Intensive mode error recording allows each temporary error over a link (and its cause) to be recorded and signaled to VTAM.

Start intensive mode recording with the MODIFY IMR command. The MODIFY IMR command causes the owning SSCP to send a request to the NCP. The NCP then builds and sends RECMS RUs to the SSCP each time an error occurs. The SSCP writes these error records on the MVS VM LOGREC or VSE SYSREC file. It can also optionally pass them to a user-defined communication network manage-

ment (CNM) application program, such as the NetView program. For more information on the MODIFY IMR command, see "MODIFY IMR Command" in *VTAM Operation.*

Modify SDLC Link Level 2 Test

Use the SDLC link level 2 (LL2) test to test the following:

- SDLC link between an NCP and a physical unit attached on a multipoint line
- SDLC link between two NCPs.
- VM VSE Link between VTAM and any secondary physical unit attached by communication adapter SDLC lines.

By running this test over an extended time, you can increase the possibility of repeating an intermittent error that is hard to re-create.

Note: If you just want to test connectivity to a physical unit, activate the physical unit instead of using a link level 2 test.

To test the link, VTAM sends test data over the link from the controlling NCP to the remote station (NCP or peripheral physical unit). The data is then echoed back to the sending NCP. This NCP then compares the data received with the data sent and forwards the results to VTAM.

Notes:

1. For communication-adapter-attached devices, VTAM sends test data over the communication adapter attachment to the physical unit. The data is then echoed back to VTAM.
2. VTAM may also receive an LL2 request from an communication-adapter-attached NCP, in which case VTAM echoes the data back to the NCP.

When VTAM receives the test results, it sends message IST549I to the initiating console to indicate how the link level 2 test ended: data returned without errors, data returned with errors, or no data returned because of an inoperative link or initialization error. See message IST549I in *VTAM Messages and Codes* for a complete message description.

When testing the link between an NCP and a peripheral physical unit, the physical unit name in the ID operand must be inactive ("dedicated" to the test). Other physical units on the same link, however, may remain active.

When testing the link between two NCPs, the primary link station is the one named in the LL2 command, and the secondary link station is the one responding to the test. You must start the test from the primary link station. The primary link station must be inactive, and the link must be active.

Start this test with the MODIFY LL2 command. This command causes the SSCP to send a test RU to the NCP to which the test terminal is connected. The NCP returns test results to the requesting SSCP in a Record Test Results RU.

The test results include:

- The number of test frames transmitted by *nodename*
- The number of test frames received by *nodename*

- The number of test frames received without error by *nodename* (that is, successfully returned by the test station).

If the above three numbers are not the same, that indicates an error in the link or a physical unit. Using a line trace might help you further isolate the problem.

See "MODIFY LL2 Command" in *VTAM Operation* for more information about the syntax of the MODIFY LL2 command.

Modify Tuning Statistics

To record tuning statistics for VTAM, use the TNSTAT start option and include SMF in the system during system operation. You can later stop and start the recording of tuning statistics with the MODIFY NOTNSTAT and MODIFY TNSTAT commands.

For detailed information about recording tuning statistics and using these statistics to improve communication between VTAM and a channel-attached communication controller or host processor, see "Gathering Tuning Statistics" in the *VTAM Network Implementation Guide*. For information on the syntax of the MODIFY TNSTAT and the MODIFY NOTNSTAT command see "MODIFY TNSTAT Command" and "MODIFY NOTNSTAT Command" in *VTAM Operation*. For information on the TNSTAT start option, see "TNSTAT" in *VTAM Resource Definition Reference*.

MODIFY VTAMOPTS Command for Changing Start Option Values

The MODIFY VTAMOPTS command enables you to change certain values that may have been specified on VTAM start options.

For a description of start options that can be changed using this command, see "MODIFY VTAMOPTS Command" in *VTAM Operation*.

Chapter 6. Using Dumps

This chapter covers the dumps that you can use for problem determination for the VTAM program. The included dumps are:

- MVS Dumps
  - "Abend Dump" on page 214
  - "FFST Dump" on page 214
  - "Stand-Alone Dump" on page 214
  - "SVC Dump" on page 214.
- VM Dumps
  - "FFST Dump" on page 215
  - "GDUMP Dump" on page 215
  - "SDUMP Dump" on page 215
  - "DUMP Command" on page 216
  - "VMDUMP Command" on page 216.
- VSE Dumps
  - "Program-Initiated Dump" on page 216
  - "SDAID Dump Facility" on page 216
  - "Stand-Alone Dump Utility" on page 216
  - "DUMP Command" on page 217.
- "First Failure Support Technology (FFST) for VTAM (MVS) (VM)" on page 217
- "Network Control Program (NCP) Dump" on page 227
- "Communication Scanner Processor (CSP) Dump (3720, 3725, and 3745 Only)" on page 230
- "Maintenance and Operator Subsystem (MOSS) Dump (3720, 3725, and 3745 Only)" on page 231.

"Formatting and Printing Dump Output (MVS) (VM)" on page 232 describes the service aids available for formatting and printing dump output.

Dumps on MVS Operating System

Several kinds of dumps can be produced in an MVS system, depending on the type of failure and operator action:

- "Abend Dump"
- "FFST Dump"
- "Stand-Alone Dump"
- "SVC Dump."

Abend Dump

If the appropriate DD card exists, an abend dump is produced when one of the following occurs:

- The operator enters a CANCEL command.
- An abend macroinstruction is issued.
- A job abnormally ends.

To get an abend dump, the input stream for VTAM must contain a DD statement with the ddname SYSUDUMP or SYSABEND. The resulting dump is written to the data set specified on the SYSUDUMP or SYSABEND DD card. The contents of the dump depend on user specifications. See Table 80 on page 1165 to determine what book has more information on the abend dump.

FFST Dump

For information on dumps generated by First Failure Support Technology (FFST), see "First Failure Support Technology (FFST) for VTAM (MVS) (VM)" on page 217.

Stand-Alone Dump

The stand-alone dump is produced when the operator invokes the stand-alone dump program. This program can be invoked when the operating system is in a disabled wait state or looping. The stand-alone dump may be a high-speed dump, which is not formatted, or a low-speed dump, which is formatted. The dump output is written to the tape or printer (low-speed only) specified on the output operands. The output for a high-speed dump can be formatted for viewing by IPCS. See Table 80 on page 1165 to determine what book contains more information on the stand-alone dump.

SVC Dump

SVC dumps are produced under these conditions:

- VTAM produces an SVC dump automatically when a program exception occurs. VTAM might be terminated as part of this process. The system log indicates the location of the dump output and whether the dump was successful.
- An operator can request a dump with the operating system DUMP command. This will not cause VTAM to terminate.
- An operator uses a SLIP command with ACTION=SVCD specified, and an event occurs that matches the trap indicated in the SLIP.
- A macroinstruction issues an abend, and there is a DD statement with ddname=SYSMDUMP.
- An SDUMP macroinstruction is issued.

- System recovery routines produce an SVC dump if VTAM causes an error, such as a program exception or abend.

An SVC dump is written to a SYS1.DUMPnn data set (if allocated), the SYSMDUMP output data set, or the data set specified on the DCB operand of the SDUMP macroinstruction. An SVC dump can be formatted for viewing by IPCS. To determine what book contains more information on SVC dump, see Table 80 on page 1165.

Dumps on VM Operating System

Several kinds of dumps can be produced in a VM system, depending on the type of failure and operator action:

- "FFST Dump"
- "GDUMP Dump"
- "SDUMP Dump"
- "DUMP Command" on page 216
- "VMDUMP Command" on page 216

See Table 80 on page 1165 to determine what book contains information on dumping CP storage.

"Formatting and Printing Dump Output (MVS) (VM)" on page 232 tells you how to format and print the output.

FFST Dump

For information on dumps generated by First Failure Support Technology (FFST), see "First Failure Support Technology (FFST) for VTAM (MVS) (VM)" on page 217.

GDUMP Dump

The GDUMP command can be entered by the VM system operator. The resulting dump can be formatted and printed with the PRTDUMP command of IPCS. The VTAM control blocks shown in "PRTDUMP Service Aid (VM)" on page 233 are formatted in a GDUMP.

Note: For a loop problem, do not use the GDUMP command. Instead, use the VMDUMP command.

SDUMP Dump

An SDUMP dump is produced automatically when a machine program check or a GCS internal error occurs.

Note: VTAM disables the VIT when it issues SDUMP to prevent the VIT table from being overwritten. The resulting dump can be formatted and printed with the PRTDUMP command of IPCS. The VTAM control blocks shown in "ABDUMP Service Aid (MVS)" on page 232 are formatted in an SDUMP.

DUMP Command

The DUMP command can be entered by the VM system operator. This command produces a formatted hard-copy dump.

VMDUMP Command

The VM system operator can enter the VMDUMP command in CP mode as follows:

```
#CP VMDUMP 0:END FORMAT GCS DSS
```

The resulting dump can be formatted and printed with the PRTDUMP command of IPCS. The VTAM control blocks shown in "PRTDUMP Service Aid (VM)" on page 233 are formatted in a VMDUMP.

CP should have control when you issue this command. That is, you should either press PA1 or enter #CP before issuing the VMDUMP command. See "PRTDUMP Service Aid (VM)" on page 233 for details on using PRTDUMP.

Dumps on VSE Operating System

Several kinds of dumps can be produced in a VSE environment, depending on the type of failure and operator action:

- "Program-Initiated Dump"
- "SDAID Dump Facility"
- "Stand-Alone Dump Utility"
- "DUMP Command" on page 217.

Note: Dumps produced by canceling VTAM are of limited diagnostic value. Use other dump facilities to document a problem before canceling VTAM.

Program-Initiated Dump

Customer programs can produce dumps by using system dump macroinstructions (PDUMP, JDUMP, DUMP). Other IBM licensed programs, such as CICS/VS, also have their own dumping facilities, which may be useful in isolating some VTAM problems.

SDAID Dump Facility

The dump facility of SDAID is useful when debugging requires a dump of one or more specific storage areas on the occurrence of a specific event. The output device can be a tape drive or a printer. SDAID output on tape can be printed with DOSVSDMP. See Table 80 on page 1165 to determine what VSE books contain more information on DOSVSDMP and SDAID.

Stand-Alone Dump Utility

A stand-alone dump created with the DOSVSDMP utility is useful when the severity of a system malfunction (such as a hard wait state) prevents using other methods. DOSVSDMP prints the output of a stand-alone dump.

DUMP Command

The DUMP command causes VSE to produce a dump of the storage area specified in the command. The operator can direct the dump either to tape or to a SYSLST device. To retrieve the dump information, you can do either of the following:

- Use DOSVSDMP to have the dump information printed on the printer assigned to SYSLST.
- Load the dumps recorded on the tape into your system's dump library, and use Info/Analysis to retrieve the dump information.

See Table 80 on page 1165 to determine what book describes retrieving dump information from tape for VSE.

First Failure Support Technology (FFST) for VTAM (MVS) (VM)

First Failure Support Technology is a licensed program that captures information about a potential problem when it occurs. See Appendix C, "First Failure Support Technology (FFST) Probes (MVS)(VM)" on page 845 for descriptions of the various FFST probes. See Table 80 on page 1165 to determine what book contains more information on FFST.

When a problem is detected, a software probe is triggered by VTAM. FFST then collects information about the problem and generates output to help solve the problem. Based on the options active for the probe you get a dump and a generic alert. See "The Generic Alert" on page 223 for information on generic alerts. You also get the FFST 'EPW' message group as shown in the "FFST Console" on page 224.

FFST Dumps

Each VTAM FFST probe can trip up to five times in five minutes before it is automatically turned off. Only one of the five dumps will be produced, thereby limiting the number of dumps that you get if a recurring problem triggers a probe.

You can get either an SDUMP (full dump) or a FFST minidump (partial dump) depending on the characteristics of the probe that is triggered.

Dump Data Set (MVS)

FFST saves the VTAM FFST minidump on a dynamically allocated sequential data set. The VTAM FFST full dump (SDUMP) is saved on SYS1.DUMPx data sets. You must specify the volume serial number and the UNIT identification information for this data set. Provide this information to FFST on a DD statement in the FFST installation procedure or in the FFST start-up command list installed at system installation. A start-up command list contains MVS commands to control FFST.

SDUMP

If the SDUMP option has been coded in the probe, FFST uses the operating system SDUMP macroinstruction to provide a full dump of the address space where the potential problem occurred.

Formatting an SDUMP: Use the VTAM dump formatter to view or print the full dump. If you try to use EPWDMPFM or EPWDMPFV to format a full dump, message `EPW9561E NOT A VALID FFST DUMP` will be issued.

FFST Minidump

If the SDUMP option has not been coded for the probe triggering the dump, a FFST minidump is written to the output data set. See Figure 29 on page 219 and Figure 30 on page 221.

The probe output data used for VTAM FFST minidumps are found in the data sets that were allocated when VTAM FFST was installed.

Formatting a FFST Minidump (MVS): Use the dump formatting CLIST, EPWDMPFM, to format your VTAM FFST minidump. EPWDMPFM formats your minidump and writes it to a data set that you can view online or print using the IEBPTPCH utility program. (FFST minidumps cannot be processed by the VTAM formatted dump tool.)

Formatting a FFST Minidump (VM): Use the VM dump formatter EPWDMPFV to format your VTAM minidump.

1. Use DUMPLOAD to download the dump from OPERATNS.
2. Run DUMPSCAN against the output from DUMPLOAD.
3. From the command line of DUMPSCAN, enter `EPWDMPFV` to format the dump.

Sample FFST Minidump: See Figure 29 on page 219 and Figure 30 on page 221 for a sample VTAM FFST minidump. Figure 29 on page 219 was produced when VTAM session services CP-CP (SSC) entered an unexpected state on a contention-winner session.

```
SCUNO - FAILURE ON CONWINNER SESSION
11/09/92

EPW9521I DUMP DATA SET NAME = FFSTDS.MVS42247.VTAM.DMP00033
EPW9522I TITLE FROM DUMP = SCUNO - FAILURE ON CONWINNER SESSION
EPW9523I DATE FROM DUMP = 10/28/92, TIME FROM DUMP = 12:00:06
EPW9501I PRODUCT NAME: VTAM
EPW9502I IBM PROGRAM
EPW9503I COMPONENT/PROGRAM ID: 569511701, LEVEL:  201
EPW9504I TYPE OF FAILURE: INCORROUT
EPW9505I PROBE PRIMARY SYMPTOM STRING:
PIDS/569511701 LVLS/201 PCSS/ISTSCC09 RIDS/ISTSCUNO
EPW9507I REGISTER SECONDARY SYMPTOM STRING:
REGS/GR13 VALU/H068E6098 REGS/GR14 VALU/H868D40CA REGS/GR15 VALU/H00000000
REGS/GR00 VALU/H868D40CA REGS/GR01 VALU/H068E60E0 REGS/GR02 VALU/H068E6200
REGS/GR03 VALU/H0751C000 REGS/GR04 VALU/H86BC5A14 REGS/GR05 VALU/H062D1B88
REGS/GR06 VALU/H00C97EB0 REGS/GR07 VALU/H00C171F8 REGS/GR08 VALU/H068E6078
REGS/GR09 VALU/H00000442 REGS/GR10 VALU/H068D7860 REGS/GR11 VALU/H00000001
REGS/GR12 VALU/H868D3CAA
0EPW9508I DATA COLLECTION WORK AREA:
0ASID(X'0012') ADDRESS(060B1000) KEY(00)
060B1000. 61004000 00000000 050060F8 01000002 |/. ........-8....|
060B1010. 068E6098 863A27A0 000054A0 06508ED8 |..-qf........&.Q|
060B1020. C5D7E6C4 E2E3E2D2 00000000 068E6020 |EPWDSTSK......-.|
060B1030. 60006200 8652136A 065204E0 00000001 |-...f......\....|
⋮
EPW9509I SPECIFIED DATA STRUCTURE TABLE:
ASID(X'0012') ADDRESS(06999BE8) KEY(00)
06999BE8.                   77001418 018000F8 |        .......8|
06999BF0. C9E2E3E2 C3C3C6E3 00F9F2F2 F5F64040 |ISTSCCFT.92256  |
06999C00. 0000125C 069711F8 EE0000CC C9D5C3D6 |...*.p.8....INCO|
06999C10. D9D9D6E4 E3404040 002803F8 00003401 |RROUT   ...8....|
⋮
EPW9509I DEFAULT DATA STRUCTURE TABLE:
ASID(X'0012') ADDRESS(063859E8) KEY(00)
063859E8.                   7700026C 014000F8 |        ...%..8.|
063859F0. C9E2E3D9 C1C3E9E9 E4F9F2F2 F5F64040 |ISTRACZZU92256..|
06385A00. 000001F4 00000000 EE000044 C9D5C3D6 |...4........INCO|
06385A10. D9D9D6E4 E3404040 002803F8 10001C00 |RROUT   ...8....|
⋮

EPW9510I AREA AROUND REGISTER 0 - 868D40CA:
ASID(X'0012') ADDRESS(068D3CCA) KEY(00)
068D3CCA.                      C8FA 1805140E |          H.....|
068D3CD0. 14E41F0E 55004004 47D0C046 58F0C906 |.U.... ..}{..0I.|
068D3CE0. 5810C902 41000080 89000018 16100A0D |..I.....i.......|
068D3CF0. 182D1838 1E391842 1F550E24 D207D000 |.............K.}.|
⋮
EPW9510I AREA AROUND REGISTER 15 - 00000000:
ASID(X'0012') ADDRESS(00000000) KEY(00) PREFIXED
00000000. 040C0000 8122F568 00000000 00000000 |....a.5.........|
00000010. 00FD3AC0 00000000 076C2000 826077B4 |...{.....%..b-..|
00000020. 070C1000 800198DE 070C6000 8639E0B2 |......q...-.f.\.|
00000030. 070C1000 8003D52A 070C0000 81E23422 |......N.....aS..|
⋮
VTAM COMMUNICATION VECTOR TABLE
EPW9520I DATA STRUCTURE DEFINED IN DEFAULT DST
ASID(X'0012') ADDRESS(00C171F8) KEY(00)
00C171F8.                   E5C5F4F2 40404040 |        VE42    |
00C17200. FFF901F4 05F6D000 00000000 0000FFD9 |.9.4.6}.......R.|
00C17210. 11280000 00000000 062FEE88 00000000 |.........h.....|
00C17220. 00000000 00000000 00C17524 00000000 |.........A......|
⋮
```

Figure 29 (Part 1 of 2). Sample FFST Minidump MVS

```
INTERNAL TRACE TABLE HEADER
EPW9520I DATA STRUCTURE DEFINED IN DEFAULT DST
ASID(X'0012') ADDRESS(05F6D000) KEY(00)
05F6D000. 00000000 00000000 A682A39A 14E4B800 |........wbt..U..|
05F6D010. A682A398 2717CF00 05F6F040 05F9EFE0 |wbtq.....60 .9.\|
INTERNAL TRACE TABLE
EPW9514I EXIT ROUTINE CALLED FOR DATA STRUCTURE
EPW9520I DATA STRUCTURE DEFINED IN DEFAULT DST
ASID(X'0012') ADDRESS(05F6F060) KEY(00)
05F6F060. C6C2D3D2 12001200 069DFF20 86387AFC |FBLK........f.:.|
05F6F070. 868BBD02 00000020 00000000 00000000 |f...............|
05F6F080. C7C2D3D2 12001200 06F46E78 06387B40 |GBLK.....4>...# |
05F6F090. 868BCE08 00000040 00000000 00000031 |f...... ........|
⋮
CONTINUATION OF INTERNAL TRACE TABLE
EPW9514I EXIT ROUTINE CALLED FOR DATA STRUCTURE
EPW9520I DATA STRUCTURE DEFINED IN DEFAULT DST
ASID(X'0012') ADDRESS(05F6D020) KEY(00)
05F6D020. D7C9E440 12990000 06368CF8 40000902 |PIU .r.....8 ...|
05F6D030. 280007FC 000001F4 00000136 1C000001 |.......4........|
05F6D040. D7C9E4F2 0306010A 00100380 00C4C1E3 |PIU2.........DAT|
05F6D050. C140C1F3 F1C4F9F2 C1F50000 00000000 |A A31D92A5......|
⋮

SCDAT - SSC PROCESS DATA
ASID(X'0012') ADDRESS(062D1B88) KEY(00)
062D1B88.                   E2C3C4E3 00000000 |        SCDT....|
062D1B90. C0000088 06A37020 06A3713B 06B259F8 |{..h.t...t.....8|
062D1BA0. 00041100 00000000 00000000 00000000 |................|
062D1BB0. 00000000 00000000 06543018 00000000 |.............0..|
062D1BC0. 00000000 0751C000 00000000 06A37020 |.....{......t.. |
062D1BD0. 60C3D7E2 E5C3D4C7 40000000 00000000 |-CPSVCMG ........|
062D1BE0. 000C12C1 00000000 F6BE0000 00000000 |...A....6.......|
062D1BF0 LENGTH(1040)==>All bytes contain X'00'
RUPE - REQUEST UNIT PROCESSING ELEMENT
ASID(X'0012') ADDRESS(0751C000) KEY(00)
0751C000. 54136004 00000000 00000000 00000000 |..-.............|
0751C010. 00000000 00000000 00000000 FDC3D7E2 |.............CPS|
0751C020. 00000000 D5C5E3C1 4BC1F8F1 D5404040 |....NETA.A81N    |
0751C030. 40404040 40404040 40404040 40000000 |                ...|
0751C040 LENGTH(16)==>All bytes contain X'00'
0751C050. 068E5E08 00000000 40060000 00004004 |..;..... ..... .|
0751C060. 00000000 00000000 00000000 86BC5A14 |............f.!.|
0751C070. 00000000 081C0002 00000000 00000000 |................|
0751C080 LENGTH(16)==>All bytes contain X'00'
0751C090. 00000000                          |....            |
IPS - INTER PROCESS SIGNAL
ASID(X'0012') ADDRESS(06BC5A14) KEY(00)
06BC5A14.          0440FDC3 D7E202D5 C5E3C14B |    . .CPS.NETA.|
06BC5A20. C1F8F1D5 40404040 40404040 00000000 |A81N        ....|
06BC5A30. 00090000 00000300 0005081C 00021400 |................|
06BC5A40 LENGTH(1040)==>All bytes contain X'00'
06BC5E50. 00000000 0000                     |......          |
ANDCB - ADJACENT NODE CONTROL BLOCK
0ASID(X'0012') ADDRESS(06A37020) KEY(00)
06A37020. C1D5C3C2 D5C5E3C1 4BC1F8F1 D5404040 |ANCBNETA.A81N    |
06A37030. 40404040 40000000 00000009 C3D7E2E5 |     .......CPSV|
06A37040. C3D4C740 00000000 00000000 00000000 |CMG .............|
06A37050 LENGTH(80)==>All bytes contain X'00'
06A370A0. 00000000 02000000 00000000 06B62138 |................|
06A370B0 LENGTH(16)==>All bytes contain X'00'
06A370C0. 0751BE20 E0000000 02000000 00000000 |....\...........|
06A370D0. 0751BEC0 00000000 00000000 80000000 |...{............|
06A370E0. 00000000 10F01002 02000000 0751C1E0 |.....0........A\|
06A370F0. 01000000 00000000 00000000 00000000 |................|
06A37100. 00000000 00000000 00000000 A0801010 |................|
06A37110. 00000000 00000000 06BB9608 00000000 |..........o.....|
06A37120. 00000000 00000000 00000000 00       |.............   |
```

Figure 29 (Part 2 of 2). Sample FFST Minidump MVS

```
EPW9522I TITLE FROM DUMP = VTAM/FFST MINI DUMP ACTIVE

EPW9523I DATE FROM DUMP = 09/02/94, TIME FROM DUMP = 09:57:57

EPW9501I PRODUCT NAME: VTAM VM/ESA V4R2

EPW9502I IBM PROGRAM

EPW9503I COMPONENT/PROGRAM ID: 565401001, LEVEL: 420

EPW9504I TYPE OF FAILURE: INCORROUT

EPW9505I PROBE PRIMARY SYMPTOM STRING:

PIDS/565401001 LVLS/420 PCSS/ISTTESTM PCSS/MINI

EPW9507I REGISTER SECONDARY SYMPTOM STRING:

REGS/GR13 VALU/H017BDA77 REGS/GR14 VALU/H817BD11A REGS/GR15 VALU/H00000000
REGS/GR00 VALU/H80000000 REGS/GR01 VALU/H017BD784 REGS/GR02 VALU/H01943800
REGS/GR03 VALU/H01608A90 REGS/GR04 VALU/H0000BFBD REGS/GR05 VALU/H0101BF00
REGS/GR06 VALU/H0000BFBD REGS/GR07 VALU/H00484330 REGS/GR08 VALU/H00484000
REGS/GR09 VALU/H00000032 REGS/GR10 VALU/H017BDB30 REGS/GR11 VALU/H017BCFFF
REGS/GR12 VALU/H817BC000

VTAM COMMUNICATION VECTOR TABLE

DISPLAY 00484000   F80
  00484000  E5C1F4F2  40404040  0000FFFD  004CE020 66 *VA42    .....<..*
  00484010  00000000  0000BFBD  11280000  00000000    *.................*
  00484020  01001750  00000000  80000000  80000000    *...&............*
  00484030  0048432C  00498010  13200000  00000010    *.................*
  00484040  11280000  00000000  01001750  00000000    *...........&....*
  00484050  00000000  00000000  004842D4  00000000    *...........M....*
  00484060  0C280000  00000010  00000000  814F5AA8    *............a|.y*
  00484070  11400000  00000000  01001750  00000000    *. .........&....*
  .
  .
  .
  00484F10  010018A0  815FFAD8  00000000  00000000    *....a¬.Q........*
  00484F20  00000000  00000000  00000000  00000000    *.................*
  00484F30  00000000  01410B18  00000000  00000000    *.................*
  00484F40  00000000  00000000  0069E18C  00000000    *.................*
  00484F50  00000000  00484F80  00000000  00000000    *......|.........*
  00484F60  00000000  00000000  00000000  00000000    *.................*
  00484F70  00000000  00000000  00000000  00000000    *.................*

INTERNAL TRACE TABLE HEADER

DISPLAY 004CE020   20
  004CE020  00000000  00000000  A9D1E3E2  CF0A8701 6E *........zJTS..g.*
  004CE030  A9D1DD8F  9F9D8800  004D3C20  00517FE0    *zJ....h..(....".*

INTERNAL TRACE TABLE

EPW9514I EXIT ROUTINE CALLED FOR DATA STRUCTURE

EPW9524I DATA STRUCTURE GREATER THAN 65504 BYTES. OUTPUT BEING SPLIT UP.


EPW9522I TITLE FROM DUMP = VTAM/FFST MINI DUMP ACTIVE

EPW9523I DATE FROM DUMP = 09/02/94, TIME FROM DUMP = 09:57:57

EPW9501I PRODUCT NAME: VTAM VM/ESA V4R2

EPW9502I IBM PROGRAM

EPW9503I COMPONENT/PROGRAM ID: 565401001, LEVEL: 420

EPW9504I TYPE OF FAILURE: INCORROUT

EPW9505I PROBE PRIMARY SYMPTOM STRING:

PIDS/565401001 LVLS/420 PCSS/ISTTESTM PCSS/MINI
```

Figure 30 (Part 1 of 3). Sample FFST Minidump VM

```
EPW9507I REGISTER SECONDARY SYMPTOM STRING:

REGS/GR13 VALU/H017BDA77 REGS/GR14 VALU/H817BD11A REGS/GR15 VALU/H00000000
REGS/GR00 VALU/H80000000 REGS/GR01 VALU/H017BD784 REGS/GR02 VALU/H01943800
REGS/GR03 VALU/H01608A90 REGS/GR04 VALU/H0000BFBD REGS/GR05 VALU/H0101BF00
REGS/GR06 VALU/H0000BFBD REGS/GR07 VALU/H00484330 REGS/GR08 VALU/H00484000
REGS/GR09 VALU/H00000032 REGS/GR10 VALU/H017BDB30 REGS/GR11 VALU/H017BCFFF
REGS/GR12 VALU/H817BC000

VTAM COMMUNICATION VECTOR TABLE

DISPLAY 00484000   F80
  00484000  E5C1F4F2  40404040  0000FFFD  004CE020 66 *VA42    .....<..*
  00484010  00000000  0000BFBD  11280000  00000000    *.............. *
  00484020  01001750  00000000  80000000  80000000    *...&...........*
  00484030  0048432C  00498010  13200000  00000010    *.............. *
  00484040  11280000  00000000  01001750  00000000    *..........&....*
  00484050  00000000  00000000  004842D4  00000000    *..........M....*
  00484060  0C280000  00000010  00000000  814F5AA8    *............a|.y*
  00484070  11400000  00000000  01001750  00000000    *. .........&....*
  .
  .
  .
  00484F10  010018A0  815FFAD8  00000000  00000000    *....a¬.Q........*
  00484F20  00000000  00000000  00000000  00000000    *.............. *
  00484F30  00000000  01410B18  00000000  00000000    *.............. *
  00484F40  00000000  00000000  0069E18C  00000000    *.............. *
  00484F50  00000000  00484F80  00000000  00000000    *......|.........*
  00484F60  00000000  00000000  00000000  00000000    *.............. *
  00484F70  00000000  00000000  00000000  00000000    *.............. *

INTERNAL TRACE TABLE HEADER

DISPLAY 004CE020   20
  004CE020  00000000  00000000  A9D1E3E2  CF0A8701 6E *........zJTS..g.*
  004CE030  A9D1DD8F  9F9D8800  004D3C20  00517FE0    *zJ....h..(....".*

INTERNAL TRACE TABLE

EPW9514I EXIT ROUTINE CALLED FOR DATA STRUCTURE

EPW9524I DATA STRUCTURE GREATER THAN 65504 BYTES. OUTPUT BEING SPLIT UP.

DISPLAY 004D3C40   FFE0
  004D3C40  E2C3C8C4  43202800  01001750  00484050 6E *SCHD.......&.. &*
  004D3C50  81410AAA  00000000  C9D5E3D4  00000000    *a.......INTM....*
  004D3C60  C1C4E2D7  43000000  01001750  00000000    *ADSP.......&....*
  004D3C70  00000000  00484050  80000000  00000000    *...... &........*
  004D3C80  D9C5D8E2  43170000  01001750  00498010    *REQS.......&....*
  004D3C90  806F9562  00010000  0101BD98  00000000    *.?n........q....*
  004D3CA0  C4E2D700  4300A800  01001750  00484050    *DSP...y....&.. &*
  004D3CB0  00000000  00000000  C9D5E3D4  00498010    *........INTM....*
  004D3CC0  C3C3C900  43588000  00498228  00000000    *CCI.......b.....*
  004D3CD0  81410A60  00000000  58588000  00000000    *a..-............*
  004D3CE0  D8E4C558  43482400  01001750  00484710    *QUE........&....*
  004D3CF0  806B141A  014126F0  E3E2E6E4  00498010    *.,.....0TSWU....*
  004D3D00  C5E7C9E3  43000010  01001750  00484050    *EXIT.......&.. &*
  004D3D10  81410858  80000000  C9D5E3D4  00498010    *a.......INTM....*
  004D3D20  D9C5D3E2  43170000  01001750  00498010    *RELS.......&....*
  004D3D30  806F95DA  00000000  0101BD98  00000000    *.?n........q....*
  .
  .
  .
```

Figure 30 (Part 2 of 3). Sample FFST Minidump VM

```
  004D3B80  C6C2D3D2  43001400  0195E028  010050E8    *FBLK.....n....&Y*
  004D3B90  8143B3CC  00000098  00000000  00000000    *a......q........*
  004D3BA0  E6C1C9E3  43002500  01001750  004847D0    *WAIT.......&....*
  004D3BB0  8143D842  80000000  C3C6D9F2  01003500    *a.Q.....CFR2....*
  004D3BC0  D9C5D8E2  43170000  01001750  00498010    *REQS.......&....*
  004D3BD0  806F9562  00010000  0101BD98  00000000    *.?n........q....*
  004D3BE0  C4E2D700  43602000  01001750  00484028    *DSP..-.....&.. .*
  004D3BF0  01943800  01943800  D9C1E3C9  00498010    *.m...m..RATI....*
  004D3C00  5C5C5C5C  40C9E2E3  E3C5E2E3  D440D4D6    ***** ISTTESTM MO*
  004D3C10  C4C57ED5  40404040  40404040  40404040    *DE=N           *
  004D3C20  5C5C5C5C  40E5C9E3  40C4C9E2  C1C2D3C5    ***** VIT DISABLE*
  004D3C30  C440C6D6  D940E5E3  C1D461C6  C6E2E340    *D FOR VTAM/FFST *

VTAM MODULE LIST

DISPLAY 00484F80   3000
  00484F80  00486DB0  00487F80  00000000  00000000 66 *.._...".........*
  00484F90  D6D9C3C7  C240F9F4  4BF2F2F7  00656C70    *ORCGB 94.227..%.*
  00484FA0  D6D9C3C6  C240F9F4  4BF0F7F3  006A3638    *ORCFB 94.073....*
  00484FB0  C1C9C3C1  D9E5E3E5  D4F3F0F1  0065C9C8    *AICARVTVM301..IH*
  00484FC0  C1C9C3C9  D6E5E3E5  D4F3F0F1  00657860    *AICIOVTVM301...-*
  00484FD0  C1C9C3C9  D940F9F4  4BF1F7F1  0066C980    *AICIR 94.171..I.*
  00484FE0  C1C9C3D9  E7E5E3E5  D4F3F0F1  0069B180    *AICRXVTVM301....*
  00484FF0  C1C9C3E4  C5E5E3E5  D4F3F0F1  0065D920    *AICUEVTVM301..R.*
  00485000  C1C9C3E5  C340F9F3  4BF2F3F6  006FA450 66 *AICVC 93.236.?u&*
  00485010  C1D7C3D2  E440F9F4  4BF2F0F6  0065E678    *APCKU 94.206..W.*
  00485020  C1D7C3E4  C5E5E3E5  D4F3F0F1  0068EF80    *APCUEVTVM301....*
  00485030  E3E2C3C3  D9E5E3E5  D4F3F0F1  0067E178    *TSCCRVTVM301....*
  00485040  E3E2C3C3  E2E5E3E5  D4F3F0F1  00669690    *TSCCSVTVM301..o.*
  00485050  E3E2C3C4  D9E5E3E5  D4F3F0F1  0068A218    *TSCDRVTVM301..s.*
  00485060  E3E2C3C6  C940F9F3  4BF2F3F5  006C4C88    *TSCFI 93.235.%<h*
  .
  .
  .
  00486D20  D9C1C3C3  C940F9F4  4BF2F0F6  006625F8    *RACCI 94.206...8*
  00486D30  C9C5C3E5  D9E5E3E5  D4F3F0F1  0066DB08    *IECVRVTVM301....*
  00486D40  E3E2C3D3  C240F9F3  4BF2F3F5  006C0170    *TSCLB 93.235.%..*
  00486D50  E3E2C3D3  C5E5E3E5  D4F3F0F1  00659008    *TSCLEVTVM301....*
  00486D60  E3E2C3D3  E240F9F3  4BF3F4F8  006BF7A8    *TSCLS 93.348.,7y*
  00486D70  E3E2C3F3  E240F9F3  4BF3F2F1  006B8EE8    *TSC3S 93.321.,.Y*
  00486D80  E3E2C3F3  D940F9F3  4BF2F4F5  006B9520    *TSC3R 93.245.,n.*
  00486D90  E3E2C3F3  C940F9F3  4BF2F3F5  006BA050    *TSC3I 93.235.,.&*
  00486DA0  E3E2C3F3  D640F9F3  4BF2F3F5  006B9E88    *TSC3O 93.235.,.h*
  00486DB0  00000000  00000000  00000000  00000000    *................*
  00486DC0  00000000  00000000  00000000  00000000    *................*
  .
  .
  .
  00486DD0  00000000  00000000  00000000  00000000    *................*
  00486DE0  00000000  00000000  00000000  00000000    *................*
  00486DF0  00000000  00000000  00000000  00000000    *................*
  00486E00  00000000  00000000  00000000  00000000    *................*
  .
  .
  .
  00487F50  00000000  00000000  00000000  00000000    *................*
  00487F60  00000000  00000000  00000000  00000000    *................*
  00487F70  00000000  00000000  00000000  00000000    *................*
```

Figure 30 (Part 3 of 3). Sample FFST Minidump VM

The Generic Alert

A software generic alert is built from the symptom record and routed to the NetView program if installed. The generic alert contains:

- The date and time that the probe was triggered
- The system name from the CVTSNAME field
- The product name (VTAM)
- The component identification and release number of the product triggering the probe
- The hardware identification information:
  - Machine type
  - Serial number

Model number
Plant code

- The dump data set and volume if a dump was taken
- The probe statement identifier
- The probe statement description
- The probe statement severity level.

The Symptom String

The primary symptom string contains the following data supplied by VTAM.

- PIDS/component ID: the VTAM component identifier
- LVLS/level: the VTAM specification for the product level
- PCSS/Probe ID: from the probe that was triggered
- PCSS/FULL or MINI: the type of dump taken
- RIDS: Module name from the probe that was triggered.

FFST Console

See Figure 31 and Figure 32 on page 225 for a sample console listing for FFST.
In Figure 31 the FFST program console message group 'EPW' shown informs you that a probe has been triggered and that data is being collected. The **EPW0404I** messages contain the primary symptom string for VTAM.

```
EPW0401I FFSTPROC: ERROR DETECTION INVOKED BY VTAM 287
EPW0406I DUMP DATASET IS: USER1.SP41D23.VTAM.DMP00002
EPW0407I FOUND ON VOLUME: CPDLB2
EPW0402I PRIMARY SYMPTOM STRING FOR VTAM FOLLOWS:
EPW0404I PIDS/569511701 LVLS/201 PCSS/ISTTSC01 PCSS/VR#HANG PCSS/MINI
EPW0404I RIDS/ISTTSCRI FLDS/TH4VRSSN VALU/H0000 FLDS/VRBSQRCV
EPW0404I VALU/H0000 FLDS/VRBDSTSA VALU/H00000000
EPW0701I END OF MESSAGE GROUP
F FFSTPROC,AP=FFST
EPW0610I FFSTPROC: DISPLAY FOR APPLID FFST FOLLOWS: 294
EPW0611I APPLID COUNTS    probe  DUMP SYMRC GENAL SYMST SUPDP
EPW0612I FFST   00002/00002  EN    EN   EN    EN    EN    EN
EPW0613I DUMPQUAL = USER1, DUMPVOL = CPDLB2
EPW0614I APPLID VENDOR
EPW0615I VTAM  IBM CORPORATION
EPW0701I END OF MESSAGE GROUP
```

Figure 31. Sample VTAM FFST Console Listing MVS

```
EPW0250I EPWPITSK: FFST INITIALIZATION FOR VTAM COMPLETE
FFST A=D,V=IBM,AP=VTAM
Ready;
EPW0610I FFST: DISPLAY FOR APPLID VTAM FOLLOWS:
EPW0611I APPLID     COUNTS    PROBE  DUMP SYMRC GENAL SYMST SUPDP
EPW0612I VTAM     00011/00008  EN     EN   EN    EN    EN    EN
EPW0613I
EPW0620I DUMP DESTINATION   GA EXIT   ALRT RCV  MSG LOG /STATUS
EPW0622I SYSTEM              *N/A*    NETVALRT  FFSTLOG1/ENABLED
EPW0613I
EPW0614I APPLID   PRODUCT NAME                    VENDOR
EPW0615I VTAM     VTAM VM/ESA V4R2                IBM CORPORATION
EPW0613I
EPW0617I THE FOLLOWING PROBEIDS HAVE TRIPPED AT LEAST ONCE:
EPW0618I ISTCSC01 ISTDRC01 ISTNAC01 ISTRAC02 ISTSCC23
EPW0618I ISTTESTF ISTTESTM ISTTSC08
EPW0701I END OF MESSAGE GROUP
FFST A=CL,AP=VTAM,PROBEID=ISTTESTM,V=IBM
Ready;
EPW0601I FFST: CLEAR COMMAND COMPLETE
FFST A=D,V=IBM,AP=VTAM
Ready;
FFST: DISPLAY FOR APPLID VTAM FOLLOWS:
APPLID     COUNTS    PROBE  DUMP SYMRC GENAL SYMST SUPDP
VTAM     00011/00007  EN     EN   EN    EN    EN    EN

DUMP DESTINATION   GA EXIT   ALRT RCV  MSG LOG /STATUS
SYSTEM              *N/A*    NETVALRT  FFSTLOG1/ENABLED

APPLID   PRODUCT NAME                    VENDOR
VTAM     VTAM VM/ESA V4R2                IBM CORPORATION

THE FOLLOWING PROBEIDS HAVE TRIPPED AT LEAST ONCE:
ISTCSC01 ISTDRC01 ISTNAC01 ISTRAC02 ISTSCC23
ISTTESTF ISTTSC08
EPW0701I END OF MESSAGE GROUP
EPW0401I FFST: EVENT DETECTION INVOKED BY VTAM
EPW0402I PRIMARY SYMPTOM STRING FOR PROBEID ISTTESTM FOLLOWS:
EPW0404I PIDS/565401001 LVLS/420 PCSS/ISTTESTM PCSS/MINI
EPW0403I EVENT DETECTION FAILED - GENERIC ALERT FAILURE
EPW0412I NetView SUBSYSTEM IS NOT ACTIVE
EPW0701I END OF MESSAGE GROUP
FFST A=D,V=IBM,AP=VTAM,PROBEID=ISTTESTM
Ready;
EPW0610I FFST: DISPLAY FOR PROBEID ISTTESTM FOLLOWS:
EPW0611I PROBEID    COUNTS    PROBE  DUMP SYMRC GENAL SYMST SUPDP
EPW0612I ISTTESTM 00001/00001  EN     EN   EN    EN    EN    EN
EPW0701I END OF MESSAGE GROUP
FFST A=D,V=IBM,AP=VTAM
Ready;
EPW0610I FFST: DISPLAY FOR APPLID VTAM FOLLOWS:
EPW0611I APPLID     COUNTS    PROBE  DUMP SYMRC GENAL SYMST SUPDP
EPW0612I VTAM     00012/00008  EN     EN   EN    EN    EN    EN
EPW0613I
EPW0620I DUMP DESTINATION   GA EXIT   ALRT RCV  MSG LOG /STATUS
EPW0622I SYSTEM              *N/A*    NETVALRT  FFSTLOG1/ENABLED
EPW0613I
EPW0614I APPLID   PRODUCT NAME                    VENDOR
EPW0615I VTAM     VTAM VM/ESA V4R2                IBM CORPORATION
EPW0613I
EPW0617I THE FOLLOWING PROBEIDS HAVE TRIPPED AT LEAST ONCE:
EPW0618I ISTCSC01 ISTDRC01 ISTNAC01 ISTRAC02 ISTSCC23
EPW0618I ISTTESTF ISTTESTM ISTTSC08
EPW0701I END OF MESSAGE GROUP
```

Figure 32. Sample VTAM FFST Console Listing VM

Using the Trap Module

The IBM-supplied trap module (ISTRACZT) contains pretested probes that you can use to capture data in places where a probe has not been installed inside of VTAM.

Follow these steps to install the trap program:

Step	Action
1	Add an instruction to check the ATCFFST field in the ISTATCVT control block. If ATCFFST is zero, VTAM FFST is not available.
2	Add an instruction to check the ATCRACZT field in the ISTATCVT control block. If ATCRACZT is zero, the trap module is not available.

3	Add an instruction to call the trap module.
	`BALR R14,R15 -------- 05EF`
	`DC  X'0001' -------- 0001 (default)`
	A two-byte field containing X'0001' or X'0002' follows the BALR instruction. A value of X'0001' issues the ISTRAC01 probe macro in VTAM with the SDUMP option. An index value of X'0002' issues the ISTRAC01 probe macroinstruction with the VTAM FFST minidump option. The ISTACVT and the VIT are included in the minidump output along with the VTAM module list which contains the five significant letters of a module name, its service level, and its address. Control is returned to VTAM at the address following the two-byte index.

Notes for Using the Trap Module:

1. The calling module must save GP register 7.
2. The module is reentrant.
3. AMODE is 24 or 31.
4. All registers except 7 are saved and restored.
5. The module is in LPALIB.

Note: If you are trying to invoke the trap module outside of the VTAM environment, you will need to:

- Follow the Notes for Using the Trap Module on page 226.
- Be in VTAM's key (6).
- Be authorized.

APPC Sense Code or RPL6 Slip Trap (VM)

The IBM-supplied slip traps are placed in the VTAM APPC component. These traps contain pretested probes that provide a full dump when a predetermined sense code or RPL6 (primary or secondary) return code is set. The sense code will trigger probe ISTRACZ3, and the RPL6RC will trigger probe ISTRACZ4. Only one slip trap can be active at a time.

Follow these steps to activate a slip trap:

Step	Action
1	Insure that the FFST program is operational.
2	Insure that the APPC VIT option is active.

Step	Action
3	Obtain the offset of the ATCASLIP field in the ATCVT. (This field is a full word.)
4	Using the CP TRACE STORE commmand, set ATCASLIP to the desired sense code or RPL6 return code.

Reporting a Problem

Some of the probes that are triggered are not VTAM problems. Analyze the probe output to determine if a VTAM problem exists before you contact an IBM Support Center representative. For more information on FFST probes, see Appendix C, "First Failure Support Technology (FFST) Probes (MVS)(VM)" on page 845.

Some probes might be triggered by VTAM problems that have already been fixed. To determine whether a problem has already been solved, take the following steps:

1. Search the RETAIN* database for occurrences of the symptom string you receive when the probe is triggered. (Also search the Information System (I/S) database or the database used by your organization.)
2. If you find an APAR that applies, apply the fix.
3. If you do not find an APAR and you cannot fix the problem, report it.

   For non-VTAM problems, call your IBM branch office. For suspected VTAM problems, do either of the following steps:

   - Access IBMLink and search for a similar problem by using the symptom string. If no matches are found, report the problem to IBM by using the electronic technical report (ETR) option on IBMLink.
   - Contact the IBM Support Center at 1-800-237-5511.

If you call the IBM Support Center, the Center must verify that the documentation collected is adequate to fix the problem and that the problem is a VTAM problem.

If it is a VTAM problem, the IBM Support Center opens an APAR against VTAM and includes the symptom string generated by the probe as part of the APAR text.

If it is a hardware, network definition, or user definition error, the IBM Support Center representative creates an ASKQ item for VTAM. The ASKQ item includes the symptom string and the solution for the problem and can be found in the problem determination database (PDDB).

Network Control Program (NCP) Dump

You can get either dynamic or static dumps of NCP storage in a communication controller.

You can use the DISPLAY NCPSTOR command (described on page 196) to dynamically dump up to 256 bytes of NCP storage. When you request a dynamic dump, VTAM sends repeated DISPSTOR RUs to the NCP until the entire contents of NCP storage have been transmitted. The NCP continues to operate during this period. As a result, the dump represents NCP storage over a period of time.

You can get a static dump with the VTAM dump facility (for a channel-attached or link-attached communication controller). You can also get a static dump with the independent dump utility (for a channel-attached communication controller only). Table 14 on page 229 summarizes the methods and requirements for dumping the NCP.

NCP dumps are not allowed on lines or data links (including multipoint subarea links) for which IPL=NO was specified in the NCP definition deck. Therefore:

- Code IPL=YES for any line or link which might be used to IPL or dump any attached type 4 physical unit.
- If IPL=YES was not specified for a line with one or more type 4 physical units attached, do not try to load or dump over that line.

Note: If you do not specify a value for IPL, the default is IPL=NO.

If you have a 3720 or 3745 Communication Controller with a hard disk in your network, you can transfer NCP, MOSS, and CSP dumps from the communication controller disk to the host. You can also transfer the NCP load module from the host to the disk. If you need to restart the NCP, you can load the NCP load module from the disk.

Specifically, the following tasks are allowed with the 3720 or 3745 Communication Controller with hard disk:

- You can save the NCP load module to the disk.
- You can load the NCP load module from the disk.
- You can transfer an NCP, MOSS, or CSP dump stored on the disk to the host.
- You can purge an NCP, MOSS, or CSP dump stored on the disk.
- You can control automatic loading and dumping of the NCP to or from the disk.
- You can display up to 256 bytes of NCP dump or state vector stored on the disk.

When to Use the NCP Dump

A dump of the NCP should be taken whenever the NCP abnormally terminates or when an error is suspected in the NCP. It may be possible to determine that a problem exists in the NCP by using the VTAM I/O trace to determine what PIUs are being sent to and received from the communication controller and by using the NCP line trace to determine what is happening on the lines between the communication controller and the link-attached logical unit.

Using the NCP option on MODIFY DUMP

To dump the NCP, use the MODIFY DUMP command with TYPE=NCP specified. For more information on this command, see "MODIFY DUMP Command" in *VTAM Operation*. You must execute a separate job to print the dump.

Note: A data host cannot load or dump an NCP.

You can use the DUMPDS operand of the MODIFY DUMP command to specify the file into which you want the dump transferred. If you omit this operand, VTAM uses the file specified on the PCCU definition statement for the NCP.

Note: If you use the same file on the MODIFY DUMP command as was named on the PCCU definition statement for the NCP, an earlier NCP dump may be overwritten.

If you omit the DUMPSTA operand, VTAM uses the link station specified on the VARY ACT command or the PCCU definition statement for the NCP, in that order.

MVS VM If you are using SNA network interconnection, and you want a gateway NCP to perform a dump, you can set its link station name on the DUMPSTA operand only if that NCP is in the same network as the host processor requesting the dump.

If you set AUTODUMP=YES on the PCCU definition statement, a dump is taken automatically if the NCP abnormally terminates. This dump is written to the dump file named on the DUMPDS option of the PCCU definition statement. If AUTOIPL=YES is specified on the PCCU definition, the NCP is automatically reloaded after the dump is taken. With the 3720 or 3745 Communication Controller with hard disk, you can specify DUMPLOAD=YES on the VARY ACT command, which automatically stores the dump on the disk and loads the NCP load module from the disk. (For more information about dump commands, see "MODIFY DUMP Command" in *VTAM Operation*.)

Note: For the 3705 only, the NCP is partially overwritten in the communication controller storage when the dump is taken. If the NCP is running with partitioned emulation programming (PEP), the emulation routines are included in the dump and are also partially overwritten. The 3705 must be reactivated after the dump is taken.

Table 14 (Page 1 of 2). Dumping the NCP

Method of Starting NCP Dump	Channel-or-Link Attached	Requirements	NCP Status	Printing Dump
MODIFY DUMP command, OPTION=DYNA	Both	PCCU definition requirements: DUMPDS must be specified.	Active throughout dump, no reactivation required.	Execute SSP dump formatting program IFLDUMP MVS VM or IFUDUMP VSE
MODIFY DUMP command, OPTION=STATIC	Both	PCCU definition requirements: DUMPDS must be specified.	Deactivated when dump is completed; operator must reactivate the NCP.	Execute SSP dump formatting program IFLDUMP MVS VM or IFUDUMP VSE
VTAM Error Recovery Procedures (automatic)	Both	PCCU definition requirements: DUMPDS must be specified. AUTODMP=YES must be specified. AUTOIPL may be specified.	Activated after dump is completed if AUTOIPL=YES and restart is successful, or if AUTOIPL=NO and operator requests communication controller IPL.	Execute SSP dump formatting program IFLDUMP MVS VM or IFUDUMP VSE

Table 14 (Page 2 of 2). Dumping the NCP

Method of Starting NCP Dump	Channel-or-Link Attached	Requirements	NCP Status	Printing Dump
VTAM Error Recovery Procedures (with operator intervention)	Both	PCCU definition requirements: DUMPDS must be specified. AUTODMP=NO must be specified. AUTOIPL may be specified.	Activated after dump is completed if AUTOIPL=YES and restart is successful, or if AUTOIPL=NO and operator requests communication controller IPL.	Execute SSP dump formatting program IFLDUMP MVS VM or IFUDUMP VSE
NCP dump utility: IFLREAD MVS VM or IFUREAD VSE	Channel-attached only	See Table 80 on page 1165 to determine what book describes the NCP dump utility.	Deallocate communication controller from VTAM and allocate to the independent dump utility.	IFLREAD MVS VM uses IFLDUMP to print the dump automatically. IFUREAD VSE uses IFUDUMP to print the dump automatically.
Controller-detected error (3720 or 3745 with disk only)	Both	VARY ACT requirements: DUMPLOAD=YES must be specified. The dump slot for this central control unit (CCU) on the controller's hard disk must be empty.	Activated after dump is completed if AUTOIPL=YES and restart is successful, or if AUTOIPL=NO and operator requests communication controller IPL.	Execute SSP dump formatting program IFLDUMP MVS VM or IFUDUMP VSE Transfer to host using MODIFY DUMP with ACTION=TRANSFER.

Using the Independent NCP Dump Utility (Channel-Attached Controller Only)

To use the independent NCP dump utility, the communication controller must be inactive. See Table 80 on page 1165 to determine what book describes the following:

- Job control language needed to invoke the independent NCP dump utility
- NCP data areas, registers, and codes found in an NCP storage dump.

Note: VSE To use this utility in VSE, a communication controller must be assigned to a programmable logical unit for the partition in which it is to run. When the dump is complete, you must take away the assignment to make the device available to VTAM again.

Communication Scanner Processor (CSP) Dump (3720, 3725, and 3745 Only)

The communication scanner processor (CSP) automatically dumps its contents when it detects an error. CSP stores the dump on the communication controller disk, or on the MOSS diskette, and the NCP sends an alert message to the host to

inform it of the error. You can use the MODIFY DUMP command to transfer this dump to a dump file in the host processor. After the dump has been transferred to the host, you must then run a separate job to print the dump.

If you have a 3720 or 3745 Communication Controller with hard disk in your network, you can also purge the CSP dump from the communication controller with the MODIFY DUMP command.

To transfer the contents of the CSP dump, use the MODIFY DUMP command with TYPE=CSP specified. You can use the DUMPDS operand of the MODIFY DUMP command to specify the file into which you want the dump transferred. If you omit this operand, the dump is put into one of two dump files:

- The file specified by the CDUMPDS operand of the PCCU definition statement for the NCP
- The file specified by the DUMPDS operand of the PCCU definition statement for the NCP (if CDUMPDS is not specified on the PCCU definition).

For more information on the MODIFY DUMP command, see "MODIFY DUMP Command" in *VTAM Operation*.

Maintenance and Operator Subsystem (MOSS) Dump (3720, 3725, and 3745 Only)

The maintenance and operator subsystem (MOSS) automatically dumps its contents when it detects an error. MOSS stores the dump on the communication controller disk, or on the MOSS diskette, and the NCP sends an alert message to the host to inform it of the error. You can use the MODIFY DUMP command to transfer this dump to a dump file in the host processor. After the dump has been transferred to the host, you must then run a separate job to print the dump.

If you have a 3720 or 3745 Communication Controller with hard disk in your network, you can also purge the MOSS dump from the communication controller with the MODIFY DUMP command.

To transfer the contents of the MOSS dump, use the MODIFY DUMP command with TYPE=MOSS specified. You can use the DUMPDS operand of the MODIFY DUMP command to specify the file into which you want the dump transferred. If you omit this operand, the dump is put into one of two dump files:

- The file specified by the MDUMPDS operand of the PCCU definition statement for the NCP
- The file specified by the DUMPDS operand of the PCCU definition statement for the NCP (if MDUMPDS is not specified on the PCCU definition).

For more information on the MODIFY DUMP command, see "MODIFY DUMP Command" in *VTAM Operation*.

Formatting and Printing Dump Output (MVS) (VM)

The service aids described in this section are available for formatting and printing dump output.

IPCS Service Aids (MVS)

IPCS processes SVC dumps and high-speed stand-alone dumps for online viewing. For information on using IPCS with VTAM, see Chapter 7, "Using VTAM Dump Analysis Tools (MVS)" on page 237.

To determine what book further describes IPCS commands, see Table 80 on page 1165.

ABDUMP Service Aid (MVS)

ABDUMP operates as part of the operating system's abnormal termination (abend) procedure. It automatically formats and prints abend dumps. (See "Abend Dump" on page 214.)

During ABDUMP processing, VTAM formats control blocks related to the abending task and prints them as part of the dump created by ABDUMP.

ABDUMP formats the control blocks listed below.

Note: This is an alphabetical list of the control blocks that might be in a dump. They might be in a different order in the dump.

Control Block	Description
ACDEB	VTAM data extent block for the abending task
APPCB	LU 6.2 control block
COPR	Control operator control block associated with the abending task
CRA	Component recovery area for the abending task
FMCB	Function management control block and extensions for the abending task
HSICB	Half session information control block for the abending task
LUCB	Logical unit control block associated with the abending task
MPST	Memory process scheduling table for the abending task
NSICB	Logical network services information control block for the abending task
NSSCB	Logical network services storage control block for the abending task
PST	Process scheduling table for the abending task
RAB	LU 6.2 resource allocation block for the abending task
RDTE	Resource definition table application program entry for the abending task
SAB	LU 6.2 logical-resource manager-session allocation block for the abending task

Following are formatted data areas described in *VTAM Data Areas for MVS/ESA*:

ACDEB
CRA
FMCB
LUCB
MPST
PST
RDTE.

The following information appears for each control block:

- A header line with the name and hexadecimal address of the beginning of the control block
- Under the header, the name of each selected field (as it appears in that control block's mapping DSECT) and the contents of the field (listed sequentially)
- After the formatted printout, a hexadecimal dump of the entire control block.

PRTDUMP Service Aid (VM)

PRTDUMP enables you to format dump output from various VM dumps including GDUMP, SDUMP, and VMDUMP

To format VTAM control blocks from GDUMP, SDUMP, and VMDUMP output, do the following:

1. Load the dump with DUMPLOAD, and use the PRTDUMP command to format or view it. (Figure 72 on page 455 shows an example of using PRTDUMP.)

   For GDUMP, SDUMP, and VMDUMP, PRTDUMP asks whether you want VTAM control blocks to be formatted.

2. Answer yes to format the dump.

   Note: Answering yes provides only the formatted portion of the dump.

   a. PRTDUMP then asks about VSCS formatting.

   b. Enter the formatting options you want for VSCS.

   c. VTAM then asks about VTAM formatting.

   d. Enter the formatting options you want for VTAM.

3. Answer no to print the unformatted dump.

If you request formatting, you can format the following VTAM control blocks from GDUMP or SDUMP output, depending on which virtual machine is dumped and on which options you select.

Note: This list is in alphabetical order; the control blocks may be ordered differently in a dump.

Control Block	Description
ACDEB	VTAM data extent block
APPCB	LU 6.2 control block
ATCVT	VTAM communication vector table
BPCB	Buffer pool control block
BPDTY	Buffer pool directory
CONFT	VTAM configuration table (chained to ATCVT)

COPR	Control operator control block
CRA	Component recovery area (active)
ERTE	Explicit route table entry
FMCB	Function management control block (chained to FMCB extensions)
FMCBE	FMCB extensions (queued to LUCB)
FMCBEXT	Function management control block extension
HSICB	Half session information control block
ITTRC	Internal trace table
LUCB	Logical unit control block (chained to ACDEB)
MPST	Memory process scheduling table
NCB	Node control blocks (pointed to by RDTE)
NSICB	Logical network services information control block
NSSCB	Logical network services storage control block
PST	Process scheduling table
PXB	Pool expansion blocks (chained to BPCB)
QAB	Queue anchor block for the RDT
RAB	LU 6.2 resource allocation block
RDTE	Resource definition table entry
SAB	LU 6.2 logical-resource-manager session-allocation block
SIB	Session information block
SIBIX	Session information block initiation extension
SIBRX	Session information block resource extension
SIBX	Session information block cross-network extension
SPANC	Storage pool anchor block
SPTAE	SPANC task-associated element
VRBLK	Virtual route block
WRE with EIDs	Waiting request element with event IDs.

Following are formatted data areas described in *VTAM Data Areas for VM/ESA*:

APPCB
COPR
HSICB
NSICB
NSSCB
RAB
SAB.

See "GDUMP Dump" on page 215 for more information on this dump.

The following information appears for each control block:

- A header line with the name and hexadecimal address of the beginning of the control block
- Under the header, the name of each selected field (as it appears in that control block's mapping DSECT) and the contents of the field (listed sequentially)
- After the formatted printout, a hexadecimal dump of the entire control block.

SADMP Service Aid (MVS)

SADMP formats and prints low-speed stand-alone dumps. During SADMP processing, VTAM formats selected control blocks and prints them as part of the dump created by SADMP.

See the diagnostic manuals for your operating system for more information on SADMP. See "Stand-Alone Dump" on page 214 for more information on this dump.

SDUMP Service Aid (VM)

This service aid formats VTAM control blocks in SDUMP output. The procedure is the same as that given in "PRTDUMP Service Aid (VM)" on page 233.

With SDUMP, the same VTAM control blocks as those listed in "ABDUMP Service Aid (MVS)" on page 232 may be formatted, depending on which virtual machine was dumped.

See "SDUMP Dump" on page 215 for more information on this dump.

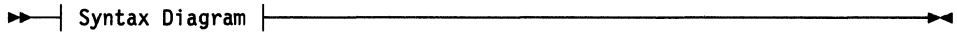
Chapter 7. Using VTAM Dump Analysis Tools (MVS)

This chapter covers the following:

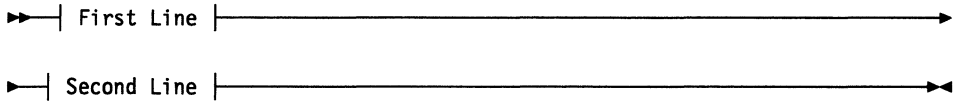
How to Read the Syntax Diagrams

This section describes how to read the syntax diagrams used in this book.

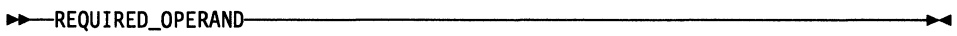
- Read the diagrams from left-to-right, top-to-bottom, following the main path line. Each diagram begins on the left with double arrowheads (►►) and ends on the right with two arrowheads facing each other (►◄).



- If a diagram is longer than one line, the first line ends with a single arrowhead (►) and the second line begins with a single arrowhead.

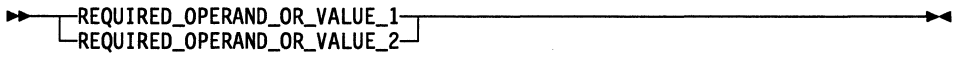


- Required operands and values appear on the main path line.

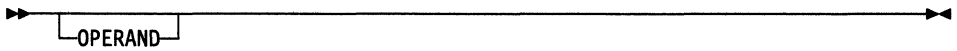


  You must code required operands and values.

  If there is more than one mutually exclusive required operand or value to choose from, they are stacked vertically in alphanumeric order.

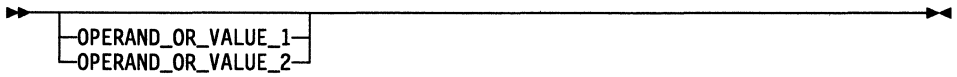


- Optional operands and values appear below the main path line.

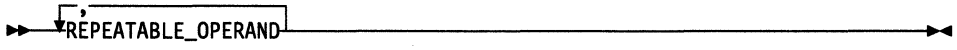


  You can choose not to code optional operands and values.

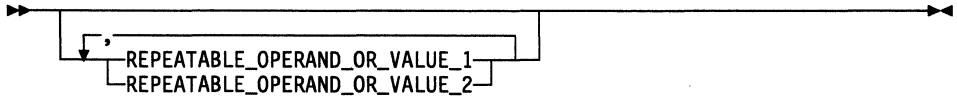
  If there is more than one mutually exclusive optional operand or value to choose from, they are stacked vertically in alphanumeric order below the main path line.



- An arrow returning to the left above an operand or value on the main path line means that the operand or value can be repeated. The comma means that each operand or value must be separated from the next by a comma.

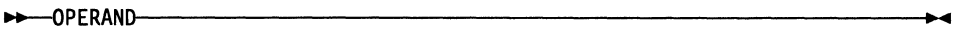


- An arrow returning to the left above a group of operands or values means more than one can be selected, or a single one can be repeated.



- A word in all uppercase is an operand or value you must spell exactly as shown. In this example, you must code ***OPERAND***.

  Note: VTAM commands are not case sensitive. You can code them in uppercase or lowercase.

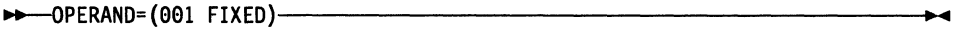
  

  If an operand or value can be abbreviated, the abbreviation is discussed in the text associated with the syntax diagram.

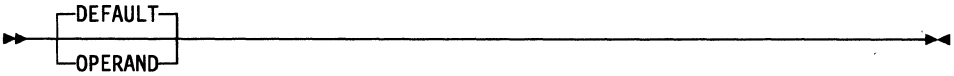
- If a diagram shows a character that is not alphanumeric (such as parentheses, periods, commas, and equal signs), you must code the character as part of the syntax. In this example, you must code ***OPERAND=(001,0.001)***.

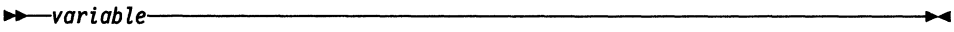
- If a diagram shows a blank space, you must code the blank space as part of the syntax. In this example, you must code ***OPERAND=(001 FIXED)***.

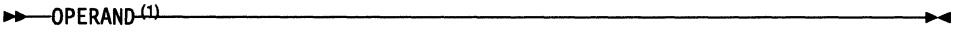
- Default operands and values appear above the main path line. VTAM uses the default if you omit the operand entirely.

- A word in all lowercase italics is a *variable*. Where you see a variable in the syntax, you must replace it with one of its allowable names or values, as defined in the text.

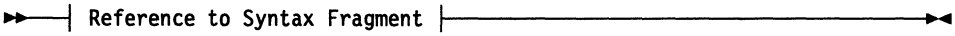
  

- References to syntax notes appear as numbers enclosed in parentheses above the line. Do not code the parentheses or the number.

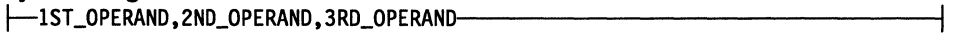
  

  Note:
  1 An example of a syntax note.

- Some diagrams contain *syntax fragments*, which serve to break up diagrams that are too long, too complex, or too repetitious. Syntax fragment names are in mixed case and are shown in the diagram and in the heading of the fragment. The fragment is placed below the main diagram.

  Syntax Fragment:

Enhanced VTAM Dump Analysis Tools

The VTAM dump analysis tools are enhancements to the IPCS subcommand VERBEXIT VTAMMAP. To use VERBEXIT VTAMMAP you can:

- Use the interactive panel interface
- Enter VERBEXIT VTAMMAP subcommands from the IPCS command line
- Create a batch job to issue the VERBEXIT VTAMMAP subcommands.

VTAMMAP will process SVC dumps, high-speed stand-alone dumps, or abend dumps.

"Sample VTAM Dump Analysis Functions" on page 242 shows how you can access the VTAM dump analysis tools.

If you experience problems that you suspect to be related to the VTAM dump analysis tools, see "VTAM Dump Analysis Tool Problems (MVS)" on page 46 for help.

For information on required target data sets for the tools, see "Installing the VTAM Dump Analysis Enhancements and the VIT Analysis Tool" in the *VTAM Network Implementation Guide*. If you want a customized interface to be active to select VTAM, see "Customizing the IPCS Interface" in the *VTAM Network Implementation Guide* for information on how to customize IPCS panel BLSPPRIM to activate option 7 on the IPCS Primary Option Menu, which is shown on page 242.

Operating Environment

- You cannot invoke multiple functions simultaneously using the IPCS command line interface to formatted dump. Batch jobs do allow this with multiple calls to VTAMMAP, but only one command can be entered on a line.
- The VTAM-supplied IPCS CLISTs must be used with VTAM V4R2.
- National Language Support (NLS) is not provided.
- IPCS for MVS/ESA is required for VTAM formatted dump.
- ISPF V3R2 or greater is required.
- To enable key lists for help panels APAR OY40082 and OY53592 must be applied to ISPF Version 3 Release 2.
- Many of the dump analysis tools described in this chapter analyze control blocks that reside in VTAM private storage. If the tool cannot access VTAM private storage, the tool will not run correctly.
- The CLISTs described in this chapter do **not** verify the accuracy of hexadecimal values such as storage addresses unless invoked from the ISPF panel.
- If you enter hexadecimal data either on the IPCS command line or using the batch option, you must enclose it in two sets of single quotation marks. (Do not use two sets if you are using the panel interface.)

  IPCS strips off the first set of quotation marks and the second set identifies hexadecimal data. For example, to be processed correctly the string '02C72020' must be entered as ''02C72020''. The following example shows how to enter hexadecimal data in a command on the IPCS command line or in a batch job:

```
VERBEXIT VTAMMAP 'SIBCHECK ADDR(X''01267B8'')'
```

Using VTAM Interactive Problem Control System (IPCS) CLISTs

For the VTAM dump analysis functions that are not part of the enhanced dump analysis tools, you can use IPCS CLISTs to issue commands to analyze dumps of VTAM storage. To start the CLIST, type the CLIST name on the IPCS command line.

For example, to start ISTVABND type

```
ISTVABND
```

on the command line.

You can also use the CLIST command interface provided by IPCS to group TSO and IPCS commands together if you want to automate dump analysis procedures. See Table 80 on page 1165 for a list of books that describe how to use IPCS.

Although these CLISTs are normally used online with IPCS, you can also issue them from the panel interface or run them in the background as batch jobs.

The IPCS CLISTs included in VTAM and described in this chapter are:

- "ISTVABND" on page 280
- "ISTVDUMP" on page 283
- "ISTVMAP" on page 286
- "ISTVSAVE" on page 288
- "ISTVSLIP" on page 291.

Obtaining Online Help for CLISTs

Note: This help operand applies only to the CLISTs; online help is invoked differently in the panel interface.

Each CLIST has online help information. To display it, enter the CLIST name followed by the HELP operand. Use no other operands, required or optional. For example, the following entry would display information on the ISTVDUMP CLIST.

```
ISTVDUMP HELP
```

After HELP information is displayed, you are prompted to either run the CLIST or exit the program. If you run the CLIST, you are prompted for each required operand.

Debugging CLIST Errors

Note: This DEBUG option does not apply to the panel interface.

When you suspect an error in the execution of a CLIST, use the DEBUG option to list each command within the CLIST before and after execution. Specify the DEBUG option after any required parameters when the CLIST is invoked, as shown in the following example.

```
ISTVSLIP DEBUG
```


Printing CLIST Output

The output from each CLIST is put into IPCSPRNT, the IPCS PRINT file. See Table 80 on page 1165 to determine the book that contains information on IPCSPRNT.

Sample VTAM Dump Analysis Functions

The following sample procedures provide examples of the ways in which you may access the VTAM dump analysis tools.

Using the Panel Interface

Follow these steps to access VTAM formatted dump using the panel interface:

Step	Action
1	Log on to TSO
2	Access IPCS
3	Select option 7 from the option list.

```
------------IPCS PRIMARY OPTION MENU----------------
OPTION  ===> _


0  DEFAULTS  - Specify default dump and options
1  BROWSE    - Browse dump data set
2  ANALYSIS  - Analyze dump contents
3  SUBMIT    - Submit problem analysis job to batch
4  COMMAND   - Enter IPCS subcommand or CLIST
5  UTILITY   - Perform utility functions
6  DUMPS     - Manage dump inventory
7  VTAM      - VTAM dump analysis
T  TUTORIAL  - Learn how to use the IPCS dialog
X  EXIT      - Terminate using log and list defaults


Enter END command to terminate IPCS dialog
```

If you want a customized interface to be active to select VTAM, see "Customizing the IPCS Interface" in the *VTAM Network Implementation Guide* for information on how to customize IPCS panel BLSPPRIM.

Step	Action
4	Select an option from the VTAMMAP Analysis Menu

```
ISTD0001                VTAMMAP Analysis Menu

Select one of the following.  Then press Enter.

__  1.  APPC . . - APPLCONV, PARTNRLU, APPLMODE, APPMODAL
    2.  APPN . . - APPNBASE, FNDADJCP, FNDANDCB, FNDCOS, FNDDECB, etc
    3.  General. - HOST, VTAM, VTBASIC, VTFNDMOD, VTMODS, VITAL, etc
    4.  Queues . - PABSCAN, VTCVTPAB, VTREADYQ
    5.  Resource - RDTCHECK, RDTFULL, RDTHIER, RDTSUM, VTNODE
    6.  Session. - FINDDSIB, FINDSIB, SES, SIBCHECK
    7.  Search . - SRTFIND
    8.  Storage. - SPANC, STORAGE, VTBUF, VTRPH
    9.  Waits. . - VTWRE
    10. ERs/VRs. - ROUTES, VTVRBLK
    11. CLISTs. .- ISTVABND, ISTVDUMP, ISTVMAP, ISTVSAVE, ISTVSLIP




(C) Copyright IBM Corporation 1993. All rights reserved.
Command ===>
 F1=Help    F2=Split   F3=Exit    F9=Swap    F12=Cancel
```

Follow the screen prompts to process your dump.

Using the IPCS Command Line

Follow these steps to access VTAM formatted dump using the IPCS command line interface.

Step	Action
1	Log on to TSO
2	Access IPCS
3	Select option 4 from the option list.

```
------------IPCS PRIMARY OPTION MENU----------------
OPTION  ===> _


0  DEFAULTS  - Specify default dump and options
1  BROWSE    - Browse dump data set
2  ANALYSIS  - Analyze dump contents
3  SUBMIT    - Submit problem analysis job to batch
4  COMMAND   - Enter IPCS subcommand or CLIST
5  UTILITY   - Perform utility functions
6  DUMPS     - Manage dump inventory
7  VTAM      - VTAM dump analysis
T  TUTORIAL  - Learn how to use the IPCS dialog
X  EXIT      - Terminate using log and list defaults


Enter END command to terminate IPCS dialog
```

4	Enter a VTAMMAP command on the IPCS command line. For example: `VERBEXIT VTAMMAP 'SIBCHECK ADDR(X''01267B8'')'`

Using the Batch Option

Follow these steps to access VTAM formatted dump using the batch processing interface.

Step	Action
1	Prepare the JCL data set. See Table 80 on page 1165 to determine what book describes IPCS.

Examples:	Sample command (single command):
	`VERBEXIT VTAMMAP 'RDTFULL'`
	Sample command (multiple commands):
	`VERBEXIT VTAMMAP 'RDTFULL'`
	`VERBEXIT VTAMMAP 'SIBCHECK ADDR(X''01267B8'')'`

VTAM Formatted Dump Procedures

This section contains an alphabetical list of the VTAM formatted dump analysis tools and IPCS CLISTs available with VTAM.

The descriptions for each tool include:

- Procedure name
- Description
- Operands
- Syntax
- Sample output.

ALL

Use ALL to invoke the following functions:

VTBASIC
RDTFULL
VTAM
STORAGE
ROUTES
SES.

Note: The ALL function is not displayed on the main formatted dump panel with other General functions, but is available on the General panel.

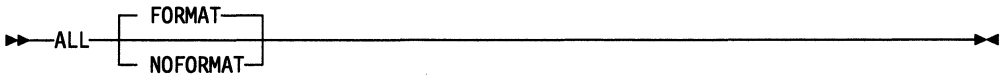
Operands

Trace output

Enter **Format** to format the VIT and **No format** to display the VIT in hexadecimal format. **Format** is the default.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for ALL

ALL

```
                              ALL Analysis

 VTAM INTERNAL TRACE TABLE    02825000

 PRESENT WRAP A50F9F0D034AA600  LAST WRAP A50F9A0C70FE24F15
 CURRENT ENTRY 0284BCC0         LAST ENTRY  028ECFE
    C4E2D740 12582410 02915E88 00CC4908   02A275F8 02A275F8 E3E2E6E4 02929010
    D3D2E2C8 12000100 00CC4C70 00000000   82A95442 00000000 00000000 02929010
    E4D5D3D2 12000100 00CC4C70 00000100   82A9546C 00000000 01000000 02929010
    D8E4C558 12482810 02915E88 00CC4248   82A954F8 02A275F8 C9D5E3D4 02929010
⋮
 ATCVT: 00CC41F8
    ATCRDT... 02955740  ATCSRT... 02C35008  ATCCONFT. 00CC18E8
    ATCBPDA.. 02953000  ATCACTRM. 0000      ATCVTLOD. 02A27650
    GWSSCP =  YES
 DATA: 00CC41F8
    +0000  E5C5F4F2  40404040  FFF900C8  02825000  | VE42    .9.H.b&. |
    +0010  00000000  0000FFF9  11280000  00000000  | .......9........ |
    +0020  02915E88  00000000  00000000  00000000  | .j;h............ |
    +0030  00CC4524  00000000  13201000  00000010  | .ö.............. |
    +0040  11280000  00000000  02915E88  00000000  | .........j;h.... |
⋮
 RDTE: 02CE0CEC
    RPRNAME.. APPCAP09  RPRENTRY. 55        RPRBITAN. 09000810  01
    RPRDEVCH. C06D0000  00800000
 DATA: 02CE0CEC
    +0000  C1D7D7C3  C1D7F0F9  80000000  00550200  | APPCAP09........ |
    +0010  00000000  00010095  000A0000  02CE0DE8  | .......n.....".Y |
    +0020  02CE0D54  0000015C  0000015C  00000000  | ."....*...*.... |
    +0030  02CE0008  00000000  00000000  02000200  | .".............. |
    +0040  00090008  10010010  00000000  00000000  | ................ |
⋮
No SIBs on the ATCSIBQ chain
```

APPLCONV

Use APPLCONV to display all conversations for an APPC application. APPLCONV formats and displays the APPCB control block, and the COPR control block if present. It will also format and display the APPC resource allocation block (RAB) and each session control block (SAB) associated with the RAB.

Operands

APPC Application Name

The APPC application name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks.

The APPC application name is required.

Use the following syntax as an alternative to the panel interface.

Sample Output for APPLCONV

```
APPLCONV APPLNAME(APPCAP05)

                      APPLCONV Analysis
 APPCB: 0290E6B8
    APPLUCB.. 0291A100  APPTSKID. 02818588  APPACB... 00CB4820
    APPLUN... APPCAP05
    APPSPTAE  02906530 0290D898 02906620 029065D0 02906580
 DATA: 0290E6B8
    +0000  62C1D7D7  0291A100  02818588  00000000  | .APP.j ..aeh.... |
    +0010  00000000  00000000  024100B0  00000000  | ...........[.... |
    +0020  31094000  00000010  00000000  00000000  | .. ............ |
    +0030  0101001B  00000000  00CB4820  0290B088  | ............[h |
    +0040  00000000  00000000  00000000  00000000  | ................ |
:
 No session limit negotiations were in progress

                Current conversation(s) for APPCAP05
 RAB: 02804028
    RABCONID. 01000003  RABCRPLA. 00000000  RABPSFSM. 01000000  04800000
    RABSABPT. 028030C0  RABNETID. NETA      RABLUNAM. APPCAP06
    RABMODEN. BATCH
    Conversation State SEND
 DATA: 02804028
    +0000  62D9C1C2  00000000  02818588  00000000  | .RAB.....aeh.... |
    +0010  00000000  01000003  01000004  D5C5E3C1  | ............NETA |
    +0020  40404040  C1D7D7C3  C1D7F0F6  C2C1E3C3  |     APPCAP06BATC |
    +0030  C8404040  A50F95D0  028030C0  00000000  | H   v.n}...{.... |
    +0040  00000000  00000000  024100B4  00000000  | ...........fl.... |
:
 SAB: 028030C0
    SABSHARE. C0        SABLRMFL. 00        SABFSM... 30
    SABSENSE. 00000000  SABNSFG1. 03        SABNSFG2. 60
 DATA: 028030C0
    +0000  62E2C1C2  00000000  00000000  01000004  | .SAB............ |
    +0010  02804028  D5C5E3C1  40404040  C1D7D7C3  | .. .NETA    APPC |
    +0020  C1D7F0F6  C2C1E3C3  C8404040  C0003000  | AP06BATCH   {... |
    +0030  00000000  00000000  02803028  00000000  | ................ |
    +0040  00000000  00ABEEC3  CE09D9CC  00000000  | .....¿.Cô.Rö.... |
    +0050  08000360  00000000  00000000  00000000  | ...-............ |
    +0060  00000000  00000000  00000000  00000000  | ................ |
    +0070  00000000  00000000  00000000  00000000  | ................ |
    +0080  00000000  00000000  00000000            | ............     |
```

APPLMODE

Use APPLMODE to display all logon modes in the logon mode table for conversations between an application program and a particular partner LU. APPLMODE will process the LU entries searching for all LU entries that match the specified partner LU name and the optional partner LU network identifier. If entries are found, APPLMODE processes the chain of modes and determines the settings for:

- Current session limits(x,y,z), where:

```
x - Session limit
y - Minimum number of contention winner sessions for local LU
z - Minimum number of contention winner sessions for remote LU
```

- Current session count(x,y,z), where:

```
x - Active session count
y - Active contention winners at local LU
z - Active contention winners at remote LU
```

- Pending session counts(x,y,z), where:

```
x - Count of pending sessions
y - Count of pending contention winners
z - Count of pending contention losers
```

- Pending session termination counts(x,y,z), where:

```
 x - Pending termination contention winners
+ Pending termination contention losers
 y - Pending termination contention winners
 z - Pending termination contention losers
```

- Defined session limits(x,y,z), where:

```
x - Defined session limit
y - Defined minimum number of contention winner sessions for local LU
z - Defined minimum number of contention winner sessions for remote LU
```

APPLMODE formats and displays the APPCB control block, and the COPR control block if present.

Operands

APPC Application Name

The APPC application name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks.

The APPC application name is required.

Partner LU Name

The partner LU name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks.

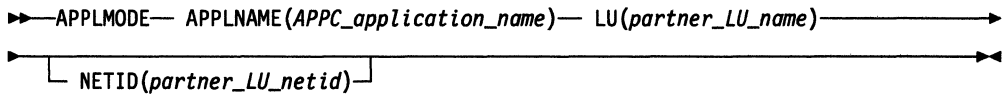
The partner LU name is required.

Partner LU NetID

The partner LU NetID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for APPLMODE

APPLMODE APPLNAME(APPCAP05) LU(APPCAP06)

```
                        APPLMODE Analysis
 APPCB: 0290E6B8
    APPLUCB.. 0291A100  APPTSKID. 02818588  APPACB... 00CB4820
    APPLUN... APPCAP05
    APPSPTAE  02906530 0290D898 02906620 029065D0 02906580
 DATA: 0290E6B8
    +0000  62C1D7D7  0291A100  02818588  00000000  | .APP.j ..aeh.... |
    +0010  00000000  00000000  024100B0  00000000  | ...........[.... |
    +0020  31094000  00000010  00000000  00000000  | .. ............. |
    +0030  0101001B  00000000  00CB4820  0290B088  | ............[h |
    +0040  00000000  00000000  00000000  00000000  | ................ |
    +0050  00000000  00000000  00000000  00000000  | ................ |
    +0060  C1D7D7C3  C1D7F0F5  00000000  00000000  | APPCAP05........ |
    +0070  00000000  00000000  028030C0  02804028  | ...........{.. . |
    +0080  00000000  0292A2FC  00000000  00000000  | .....ks......... |
    +0090  00000000  0290DA98  00000000  00000000  | .......q........ |
    +00A0  00000000  00000000  023FA0A8  00000000  | ..........ffy.... |
    +00B0  36200000  000000A0  02906530  0290D898  | .......ff......Qq |
    +00C0  02906620  029065D0  02906580  00000000  | .......}........ |
    +00D0  00000000  00000000  00000000  00000000  | ................ |
    +00E0  00000000  00000000  00000000  00000000  | ................ |
    +00F0  00000000  00000000  00000000  00000000  | ................ |
    +0100  00000000  00000000                      | ........         |

 No session limit negotiations were in progress

    Modes between application APPCAP05 and partner LU APPCAP06

 Mode name SNASVCMG

 Current session limits             (X'0002',X'0001',X'0001')
 Current session counts             (X'0001',X'0001',X'0000')
 Pending session counts             (X'0000',X'0000',X'0000')
 Pending session termination counts (X'00000000',X'0000',X'0000')
 Define session counts              (X'0004',X'0002',X'0002')

 Mode name BATCH

 Current session limits             (X'0004',X'0002',X'0002')
 Current session counts             (X'0001',X'0001',X'0000')
 Pending session counts             (X'0000',X'0000',X'0000')
 Pending session termination counts (X'00000000',X'0000',X'0000')
 Define session counts              (X'0004',X'0002',X'0002')
```

APPMODAL

Use APPMODAL to display all information about a particular logon mode for a conversation between an application and a partner LU. APPMODAL will process the LU entries searching for all LU entries that match the specified partner LU name and the optional partner LU NetID. If LU entries are found, the chain of modes is searched for a match to the specified logon mode name. If a matching logon mode is found, APPMODAL determines the settings for:

- Current session limits(x,y,z), where:

  ```
  x - Session limit
  y - Minimum number of contention winner sessions for local LU
  z - Minimum number of contention winner sessions for remote LU
  ```

- Current session count(x,y,z), where:

  ```
  x - Active session count
  y - Active contention winners at local LU
  z - Active contention winners at remote LU
  ```

- Pending session counts(x,y,z), where:

  ```
  x - Count of pending sessions
  y - Count of pending contention winners
  z - Count of pending contention losers
  ```

- Pending session termination counts(x,y,z), where:

  ```
  x - Pending termination contention winners + losers
  y - Pending termination contention winners
  z - Pending termination contention losers
  ```

- Defined session limits(x,y,z), where:

  ```
  x - Defined session limit
  y - Defined minimum number of contention winner sessions for local LU
  z - Defined minimum number of contention winner sessions for remote LU
  ```

APPMODAL displays:

- Active conversations between the two applications on the logon mode by running the chain of RABs
- Waiting requests off the logon mode (requests for conversations that have not been serviced)
- Free sessions on the logon mode (SABs that represent sessions that are not currently assigned to a conversation).
- Pending active sessions on the logon mode (SABs that represent sessions that are in the process of being activated on the logon mode).

APPMODAL formats and displays the APPCB, COPR, LME (selected fields), RAB, and SAB control blocks.

Operands

APPC Application Name

The APPC application name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks.

The APPC application name is required.

Partner LU Name
The partner LU name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks.

The partner LU name is required.

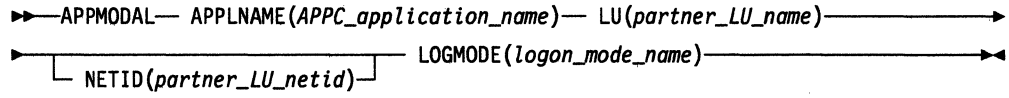
Partner LU NetID
The partner LU NetID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks.

Logmode Name
The logon mode name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks.

The logon mode name is required.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for APPMODAL

```
APPMODAL APPLNAME(APPCAP05) LU(APPCAP06) LOGMODE(BATCH)

                        APPMODAL Analysis

  APPCB: 0290E6B8
     APPLUCB.. 0291A100  APPTSKID. 02818588  APPACB... 00CB4820
     APPLUN... APPCAP05
     APPSPTAE  02906530 0290D898 02906620 029065D0 02906580
  DATA: 0290E6B8
     +0000  62C1D7D7  0291A100  02818588  00000000  | .APP.j ..aeh.... |
     +0010  00000000  00000000  02410080  00000000  | ...........[.... |
     +0020  31094000  00000010  00000000  00000000  | .. ............. |
     +0030  0101001B  00000000  00CB4820  0290B088  | ............[h |
     +0040  00000000  00000000  00000000  00000000  | ................ |
:
  No session limit negotiations were in progress

  LME: 0290B148
     LMENETID. NETA      LMENM.... APPCAP06  LMEFSM... C2

 Mode name BATCH
      .
 Current session limits              (X'0004',X'0002',X'0002')
 Current session counts              (X'0001',X'0001',X'0000')
 Pending session counts              (X'0000',X'0000',X'0000')
 Pending session termination counts  (X'00000000',X'0000',X'0000')
 Define session counts               (X'0004',X'0002',X'0002')

                        Current conversation(s)

  RAB: 02804028
     RABCONID. 01000003  RABCRPLA. 00000000  RABPSFSM. 01000000  04800000
     RABSABPT. 028030C0  RABNETID. NETA      RABLUNAM. APPCAP06
     RABMODEN. BATCH
     Conversation State SEND
  DATA: 02804028
     +0000  62D9C1C2  00000000  02818588  00000000  | .RAB.....aeh.... |
     +0010  00000000  01000003  01000004  D5C5E3C1  | ............NETA |
     +0020  40404040  C1D7D7C3  C1D7F0F6  C2C1E3C3  |     APPCAP06BATC |
     +0030  C8404040  A50F95D0  028030C0  00000000  | H   v.n}...{.... |
     +0040  00000000  00000000  024100B4  00000000  | ...........fl.... |
:
  No conversations found awaiting BID response

  No free sessions found

  No pending active sessions found
```

APPNBASE

Use APPNBASE to format the global APPN control blocks:

- ACMDT
- APNVT
- DRDAT
- MTDAT
- SCDAT
- SLGDT
- TRDAT.

The control block addresses and the hexadecimal data from each control block are provided, to help you diagnose APPN problems.

Operands

None.

Use the following syntax as an alternative to the panel interface.

Syntax

Sample Output for APPNBASE

APPNBASE

```
                        APPNBASE Analysis

  APNVT: 00C1BD40
     +0000  C1D7D7D5  00000000  062FEE88  00C1C118  | APPN.......h.AA. |
     +0010  0652A948  00000000  00000000  00000000  | ..z............. |
     +0020  00000000  00000000  00000000  00000000  | ................ |
     +0030  00000000  00000000  00000000  00000000  | ................ |
     +0040  00000000  00000000  068D8E08  00000000  | .........-...... |
     +0050  062EBE68  00000000  00C1BDC4  00000000  | ..".....A"D.... |
:
  ACMDT: 00C1C118
     +0000  C1C3D4C4  0004C1F0  F1D50000  00000004  | ACMD..A01N...... |
     +0010  D5C5E3C1  00000000  09D5C5E3  C14BC1F0  | NETA.....NETA.A0 |
     +0020  F1D54040  40404040  40408080  00000000  | 1N        ...... |
     +0030  00000000  00000000  80380100  00000000  | ................ |
     +0040  000A8C00  00000000  00000024  00006C60  | ............%-   |
     +0050  00000064  00000000  00000000  00000000  | ................ |
:
  DRDAT: 062EB1A0
     +0000  C4D9C4E3  0000C800  C0000018  069EF200  | DRDT..H.{.....2. |
     +0010  069EF110  069EF2F0  00041000  C0000018  | ..1...20....{... |
     +0020  069EF098  069EF098  069EF098  00041000  | ..0q..0q..0q.... |
     +0030  8000002C  069EF110  00480400  00000000  | ......1......... |
     +0040  00000000  00000000  00000000  00000000  | ................ |
     +0050  00000000  824B5D76  8686D5D0  824B4E9A  | ....b.).ffN}b.+. |
:
  MTDAT: 062EBD70
     +0000  D4E3C4E3  00000000  00000000  00000000  | MTDT............ |
     +0010  00000000  068DFFA8  068DFF48  067EDEF8  | .....-.y.-...=.8 |
     +0020  067EDE70  067EFFD0  067EFF98  068DE008  | .=...=.}.=.q.-\. |
     +0030  00000000  00000000  00000000  00000000  | ................ |
     +0040  00000000  00000000  0000003C  00000000  | ................ |
     +0050  10300000  00000000  00000000  80000004  | ................ |
:
  SCDAT: 062D1B88
     +0000  E2C3C4E3  00000000  C0000088  069C6B10  | SCDT....{..h..,. |
     +0010  069C6138  069C67C8  00041100  00000000  | ../....H........ |
     +0020  00000000  00000000  00000000  00000000  | ................ |
     +0030  068E9C18  00000000  00000000  00000000  | ................ |
     +0040  00000000  069C6480  60C3D7E2  E5C3D4C7  | ........-CPSVCMG |
     +0050  40000000  00000000  000C12C1  00000000  |  ..........A.... |
:
  SLGDT: 062EB080
     +0000  E2D3C4E3  00000000  8000002C  06CCBA20  | SLDT............ |
     +0010  00082400  00000000  00000000  40000024  | ............ ...  |
     +0020  06ACBAA0  06C10020  40000004  00000000  | .....A.. ....... |
     +0030  00000000  00000000  00000000  00000000  | ................ |
     +0040  824B5D76  868BDC10  824B4E9A  06318010  | b.).f...b.+..... |
     +0050  00C1BED0  06B6A100  062FEE88  00000C60  | .A"}.......h...- |
:
  TRDAT: 062D1848
     +0000  E3D9C4E3  068F0008  067ACF90  40000000  | TRDT.....:.. ...  |
     +0010  00000000  00000000  40000000  00000000  | ........ ....... |
     +0020  00000000  00000000  4000001C  00000000  | ........ ....... |
     +0030  00000000  40000008  00000000  00000000  | .... ........... |
     +0040  10000000  00000000  40000008  00000000  | ........ ....... |
     +0050  00000000  00000000  00000000  00000000  | ................ |
:
```

FINDDSIB

Use FINDDSIB to scan the ATCVT DSSIB queue for DSSIBs that meet specified selection criteria. The following are displayed for each DSSIB selected:

- DSSIB address
- Procedure correlation identifier (PCID) of the request
- Owning SSCP name
- Real name of the destination logical unit
- Real network ID of the destination logical unit
- Alias name of the destination logical unit
- Alias network ID of the destination logical unit
- Adjacent SSCP in the originating direction.

FINDDSIB has no required selection operands. If you enter no value for all selection operands, all DSSIBs are eligible for selection.

To select specific DSSIBs, you may enter a value for any of the selection operands below. All entered values must be present in the correct position within a DSSIB for it to be selected. For example, if you specify both a real name and an alias name, only DSSIBs with the specified real name in the RNAME position and the specified alias name in the ALIAS position are eligible for selection.

If you enter no value for a selection operand, DSSIBs with any value in that position are eligible for selection.

Use the Process positional operand to set the number of eligible DSSIBs which will actually be selected and displayed (all of them or just the first one encountered).

Operands

PCID

Specify 2–16 hexadecimal digits in the form X'x...' for the PCID associated with the DSRLST request. Specify an even number of digits, otherwise the high-order 4 bits are assumed to be zero. If the PCID entered is fewer than 16 digits, then it is right-justified, and a match occurs with all DSSIBs with PCIDs whose rightmost digits match the specified digits. The specified PCID is not padded with any characters.

Owning SSCP

The owning SSCP name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Real Name

The real name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Real Network ID

The real network ID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Alias Name

The alias name of the DLU resource should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Alias Network ID
The alias network ID of the DLU resource should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

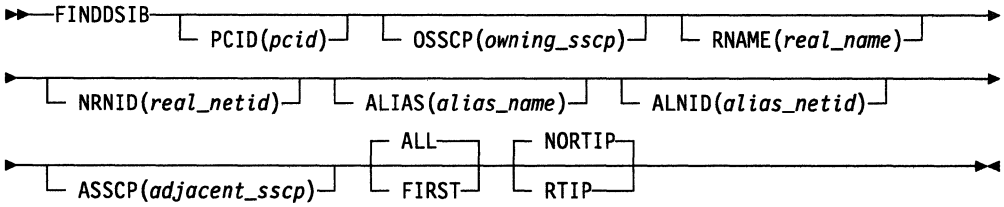
Adjacent SSCP
The adjacent SSCP in the originating direction should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Process
Use **First** to display the first DSSIB that meets the selection criteria. Otherwise, all DSSIBs that meet the selection criteria are displayed.

Routing in Progress
Use **Check RTIP** to display only DSSIBs that indicate "routing in progress" ("routing in progress" is indicated when bit DSSRTIP is on). Otherwise, FINDDSIB does not check for "routing in progress" (that is, the DSSRTIP bit is ignored).

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for FINDDSIB

FINDDSIB

```
                        FINDDSIB Analysis
 DSSIB         PCID         OSSCP    RNAME    NRNID    ALIAS    ALNID    ASSCP
-------- ---------------- -------- -------- -------- -------- -------- -------
05A13498 F0871BD0A7E3DF8C XYZSCP05 TSO105   XYZNET   TSO105   XYZNET   ABCNET
DSSIBs processed:                        1
DSSIBs matching search criteria:         1
```

FINDSIB

Use FINDSIB to scan a queue of SIBs for those that meet specified selection criteria. The following are displayed for each SIB selected:

- SIB address
- Initiation finite state machine (SIBFSMIN)
- Termination finite state machine (SIBFSMTM)
- PLU NetID name
- PLU name
- SLU NetID name
- SLU name
- PLU network address
- SLU network address
- Procedure correlation identifier (PCID).

FINDSIB has no required operands. If you enter no values for all selection operands, all SIBs on the ATCVT SIB queue are eligible for selection.

To scan an SIB queue other than the ATCVT SIB queue (such as the primary or secondary SIB queue off of an RDTE), you must specify a primary or secondary SIB queue. Specify only one queue. If both a primary queue and a secondary queue are specified, only the secondary queue will be used.

To select specific SIBs, enter a value for any of the selection operands below. All values entered must be present in the correct position within the SIB for it to be selected. If you select both PLU Name and PCID, only SIBs with the specified PLU Name in the PLUNAME position and the specified PCID in the PCID position are eligible for selection.

Note: You might need to find SIBs for a resource but do not know whether the resource is the PLU or SLU. In this special case, you can specify the resource name for both the PLU name and the SLU name, and if the resource name is found in either one, a match occurs. The SES function can also be used to find all sessions for a given resource name.

If you enter no value for a selection operand, SIBs with any value in that position are eligible for selection from the specified SIB queue.

Use the Process operand to set the number of eligible SIBs that will actually be selected and displayed (all of them or just the first one encountered).

Operands

PLU Name
The PLU name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

PLU NetID
The PLU network ID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

SLU Name
The SLU name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

SLU NetID
The SLU network ID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters the leftmost characters are compared.

PCID
Specify 2–16 hexadecimal digits in the form X'x...' for the PCID. Specify an even number of digits, otherwise the high-order 4 bits are assumed to be zero. If the PCID entered is fewer than 16 digits, then it is right-justified, and a match occurs with all SIBs with PCIDs whose rightmost digits match the specified digits. The specified PCID is not padded with any characters.

PLU Network Address
Specify 1–12 hexadecimal digits in the form X'x...' for the network address of the PLU. If you specify fewer than 12 digits, FINDSIB selects a network address whose rightmost digits match the specified digits.

```
Example: Subarea       12  Input = 1204BC  Dump Data = 0000001204BC
         Element     04BC
```

SLU Network Address
Specify 1–12 hexadecimal digits in the form X'x...' for the network address of the SLU. If you specify fewer than 12 digits, FINDSIB selects a network address whose rightmost digits match the specified digits.

```
Example: Subarea        A  Input = A1123  Dump Data = 0000000A0123
         Element     0123
```

Note: The following three operands, Displacement 1, Value 1, and Value 1 Type, must be specified together. They allow any field in an SIB to be checked for a user-specified value.

Displacement 1
Enter the displacement into the SIB where Value 1 is to be found. The maximum decimal displacement is 4095, and the maximum hexadecimal displacement is X'FFF'.

Value 1
Only SIBs containing this character, hex, or binary value at the displacement specified in Displacement 1 are selected.

Value may contain character or hexadecimal data of 1–8 bytes in length. Hexadecimal data should contain an even number of up to 16 hexadecimal digits in the form X'xx...', otherwise the high order 4 bits are assumed to be zero.

Binary data can be used to look at a particular bit within a byte. You may specify one byte of binary data in the form X'xx'. Only 1 bit within the byte may be selected. Therefore, you can specify only the following hexadecimal values: 01, 02, 04, 08, 10, 20, 40, and 80. A value with more than 1 bit set (for example, 82) will not be accepted. If you want to test 2 bits within the same byte, you must use Displacement 2, Value 2, and Value 2 Type, as well as Displacement 1, Value 1, and Value 1 Type.

Value 1 Type
Enter B for binary, C for character, or X for hexadecimal to indicate the type of data entered for Value 1.

Note: The following three operands, Displacement 2, Value 2, and Value 2 Type, are used together.

Displacement 2
Same as **Displacement 1**.

Value 2
Same as **Value 1**.

Value 2 Type
Same as **Value 1 Type**.

Note: If both (Displacement 1,Value 1,Value 1 Type) and (Displacement 2,Value 2,Value 2 Type) are specified, both sets of conditions must be met for a SIB to be selected.

You may specify only one queue, Primary SIB or Secondary SIB.

Primary SIB
Enter the address of a SIB on the primary SIB queue off of an RDTE. The address must be 1–8 hexadecimal digits in the form X'x...'. If the address specified is fewer than eight digits, it is padded on the left with zeros.

Secondary SIB
Enter the address of a SIB on the secondary SIB queue off of an RDTE. The address must be 1–8 hexadecimal digits in the form X'x...'. If the address specified is fewer than eight digits, it is padded on the left with zeros.

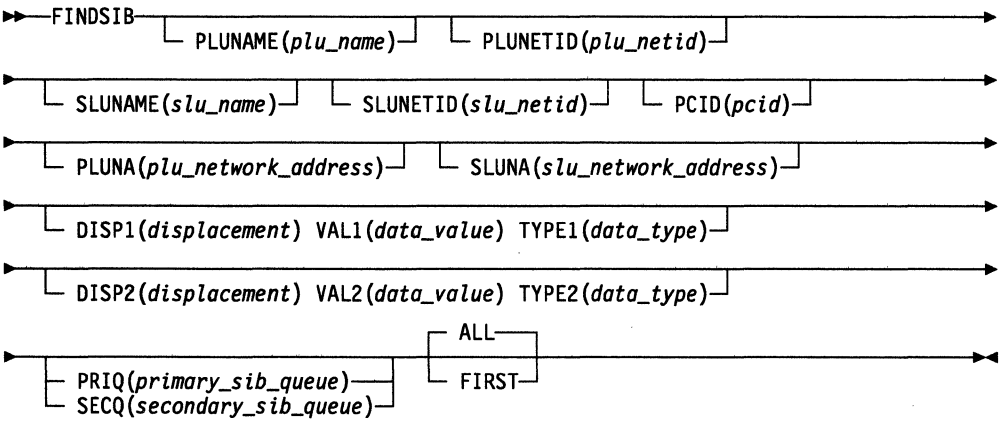
Note: If All is specified for Primary SIB or Secondary SIB, all elements from the first match are considered regardless of the address.

Process
Use **First** to display the first SIB that meets the selection criteria. Otherwise, all SIBs that meet the selection criteria are displayed.

Note: Scanning the entire SIB queue can take a long time.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for FINDSIB

```
FINDSIB PLUNETID(NETB) PLUNAME(ECHOB1B) SLUNETID(NETC) SLUNAME(C01D0067)

                          FINDSIB Analysis

SIB ADDR FSMS PLUNETID PLUNAME  SLUNETID SLUNAME     PLUNA        SLUNA        PCID
-------- ---- -------- -------- -------- --------- ------------ ------------ ----------------
069AB830 3C00 NETB     ECHOB1B  NETC     C01D0067 000000000000 000000000000 ECC39EEE2A54E5D9

SIBs processed:                    1095
SIBs matching search criteria:        1
```

FNDADJCP

FNDADJCP scans all of the partner nodes that have CP-CP sessions with this host for the given resource. If a resource is not provided, all partner nodes are displayed.

FNDADJCP has no required operands. If you do not enter a resource name, all ACPCB control blocks are formatted.

Operands

Network ID

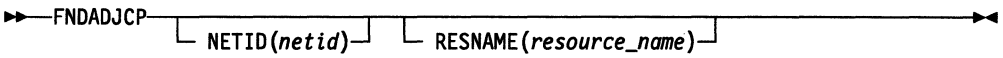
The network ID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Resource Name

The resource name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for FNDADJCP

FNDADJCP

```
                          FNDADJCP Analysis

  ACPCB: 069EF200
     +0000  40C1C3D7  D5C5E3E9  40404040  C1F0F4D7  |  ACPNETZ    A04P |
     +0010  F8F8F3C1  00040008  00000000  069EF4D0  | 883A.........4} |
     +0020  00000000  00000000  00000000  00000000  | ................ |
     +0030  00000000  00000000  00000000  00000000  | ................ |
     +0040  00000000  00000000  00000000  4000000C  | ............ ... |
     +0050  00000000  00000000  4000000C  00000000  | ........ ....... |
     +0060  00000000  00000000  10000000            | ............     |
:
  ACPCB: 069EF4D0
     +0000  40C1C3D7  D5C5E3C1  40404040  C1F0F4D7  |  ACPNETA    A04P |
     +0010  F8F8F7C1  00040008  069EF200  069EF458  | 887A......2...4. |
     +0020  00000000  00000000  00000000  00000000  | ................ |
     +0030  00000000  00000000  00000000  00000000  | ................ |
     +0040  00000000  00000000  00000000  4000000C  | ............ ... |
     +0050  00000000  00000000  4000000C  00000000  | ........ ....... |
     +0060  00000000  00000000  10000000            | ............     |
```

FNDANDCB

Use FNDANDCB to help diagnose problems with CP-CP sessions between this host and adjacent nodes. For a particular resource, FNDANDCB finds and formats the adjacent node control blocks.

Operands

Resource Name

The resource name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

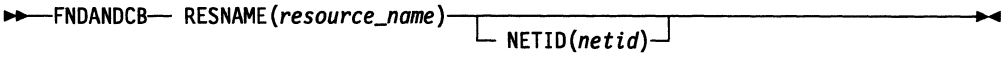
The resource name is required.

Network ID

The network ID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for FNDANDCB

FNDANDCB RESNAME(A04P887A)

```
                          FNDANDCB Analysis

ANDCB: 069C6B10
   +0000  C1D5C3C2  D5C5E3C1  4BC1F0F4  D7F8F8F7  | ANCBNETA.A04P887 |
   +0010  C1404040  40000000  0000000D  C3D7E2E5  | A    .......CPSV |
   +0020  C3D4C740  00000000  00000000  58588000  | CMG ............ |
   +0030  069C6A24  00000000  00000000  00C1BE88  | ..]..........A"h |
   +0040  04000000  00000000  00000000  00000000  | ................ |
   +0050  00000000  00000000  00000000  00000000  | ................ |
   +0060  00000000  00000000  00000000  00000000  | ................ |
   +0070  00000000  00000000  00000000  A6C1C094  | ............wA{m |
   +0080  33E74905  02000000  00000000  069C69F8  | .X.............8 |
   +0090  00000000  00000000  00000000  00000000  | ................ |
   +00A0  06B72EE0  89400000  02000000  00000000  | ...\i .......... |
   +00B0  00000000  00000000  00000000  80000000  | ................ |
   +00C0  00000001  10F01002  02000000  06B6DD00  | .....0.......... |
   +00D0  01000000  00000000  00000000  00000000  | ................ |
   +00E0  00000000  80000000  00000000  A0801010  | ................ |
   +00F0  00000000  00000000  0679ABE8  00000000  | ......... .Y.... |
   +0100  00000000  00000000  00000000  00        | .............    |

CPCAP: 0679ABE8
   +0000  C3D7C3C1  000C12C1  00000000  80800000  | CPCA...A........ |
```

FNDCOS

Use FNDCOS to format mode tables, mode table entries, and class-of-service entries.

FNDCOS formats and displays the following control blocks:

- ISTCSTRU
- ISTMCOSS
- ISTMDTAB
- ISTNDWED
- ISTTGWGT.

In order to reduce repetitious output, the control blocks ISTCSTRU, ISTNDWED, and ISTTGWGT will not display for consecutive, identical class of service names.

FNDCOS has no required operands.

Operands

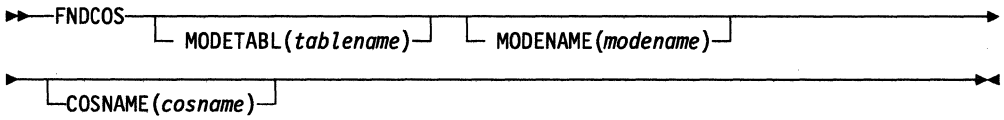
Mode Table
The name of the mode table should be 1–8 alphanumeric characters.

Mode Name
The name of the entry in the mode table should be 1–8 alphanumeric characters.

Class of Service Name
The name of the entry in the APPN class of service table should be 1–8 alphanumeric characters.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for FNDCOS

FNDCOS

```
                              COS Analysis

  MDTAB: 0695A460
     +0000  C5D4E2D4  D6C4C540  00000015  C0000014  | EMSMODE ....{... |
     +0010  0695A9A0  0695A4A0  0695A4A0  00000800  | .nz..nu..nu..... |
     +0020  00000000  0695A0E0                       | .....n.\         |

  MCOSS: 0695A9A0
     +0000  C4E8D5C1  D4C9C340  7BC3D6D5  D5C5C3E3  | DYNAMIC #CONNECT |
     +0010  0693A720  00000000  0695A960  00000000  | .lx......nz-.... |
     +0020  00000000  00000000                       | ........         |

  CSTRU: 0693A720
     +0000  7BC3D6D5  D5C5C3E3  0693A2E0  0693AB60  | #CONNECT.ls\.l.- |
     +0010  00000008  40000000  0693A760  0693A920  | .... ....lx-.lz. |
     +0020  40000000  0693A960  0693AB20  06A4E080  |  ....lz-.l...u\. |
     +0030  40                                       |                  |

  TGWGT: 0693A760
     +0000  00000000  0693A7A0  00000000  01C0004C  | ....lx......{.< |
     +0010  75FF00FF  00FF00FF  1E                   | .........        |
:
                .                 .
  TGWGT: 0693A920
     +0000  0693A8E0  00000000  00FF00FF  01C000FF  | .ly\.........{.. |
     +0010  00FF00FF  00FF00FF  F0                   | ........0        |

  NDWED: 0693A960
     +0000  00000000  0693A9A0  00001F05           | .....lz.....     |
:
                .                 .
  NDWED: 0693AB20
     +0000  0693AAE0  00000000  4000FFA0           | .l.\.... ...     |
```

FNDDECB

Use FNDDECB to format a directory entry and its parent directory entries.

FNDDECB formats and displays the ISTDECB control block.

Operands

Network ID

The network ID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Note: If you do not specify a network ID, the host network ID will be used to form a fully qualified network name.

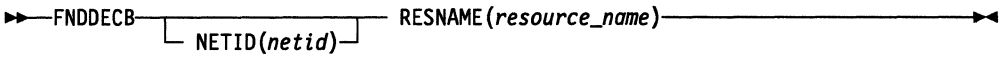
Resource Name

The resource name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

The resource name is required.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for FNDDECB

FNDDECB RESNAME(A44N)

```
                          DECB Analysis

 DECB: 068E9158
    +0000  C4C5C3C2  44000004  06BCB7D8  068E9498  | DECB.......Q..mq |
    +0010  00010000  C1F4F4D5  40404040  D5C5E3C1  | ....A44N    NETA |
    +0020  40404040  000400F4  07041548  067D7968  |     ...4.....' . |
    +0030  06A4CFC0  A682A447  068E90F0  40000048  | .u.{wbu....0 ... |
    +0040  06A38BE8  06A63978  06A38B80  06A62D88  | .t.Y.w...t...w.h |
    +0050  06A63978  06DFCF90  00550000            | .w.........      |

 Parent DECB chain

 DECB: 068E90F0
    +0000  C4C5C3C2  44000004  06A64E58  06A63B80  | DECB.....w+..w.. |
    +0010  00000000  C1F0F2D5  40404040  D5C5E3C1  | ....A02N    NETA |
    +0020  40404040  000400F6  00000000  06CEE788  |     ...6......Xh |
    +0030  06A49890  A682A461  00000000  40000048  | .uq.wbu/.... ... |
    +0040  06A38B80  06EE8B80  00000000  00000000  | .t.............. |
    +0050  06C908A8  06C90AB0  00000000            | .I.y.I......     |
```

FNDENDEL

Use FNDENDEL to help diagnose problems with adjacent end nodes. For a particular resource, FNDENDEL provides the associated ENDEL control block.

FNDENDEL has no required operands. If you do not enter a resource name, all ENDEL control blocks are formatted.

Operands

Network ID

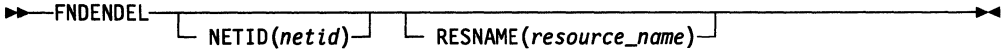
The network ID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Resource Name

The resource name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for FNDENDEL

```
FNDENDEL RESNAME(A04P208A)

                        FNDENDEL Analysis

ENDEL: 06A41020
   +0000  C5D5C4D3  00000000  06A41048  000DD5C5  | ENDL.....u....NE |
   +0010  E3C14BC1  F0F4D7F2  F0F8C140  40404000  | TA.A04P208A    . |
```

FNDLCB

Use FNDLCB to help diagnose problems with directory search requests. For a particular procedure correlation identifier (PCID), FNDLCB finds and formats the following control blocks:

- LCB
- LCB extension
- OSCB
- Original and best reply PLOCBs from the queue of LCB control blocks.

FNDLCB has no required operands. If you do not enter a PCID, all control blocks from the previous list are formatted.

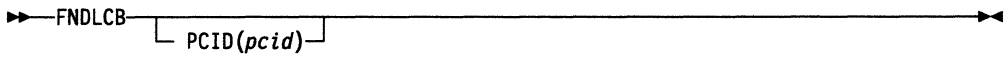
Operands

PCID

Specify 1–16 hexadecimal digits in the form X'x...' for the PCID. Specify an even number of digits, otherwise the high-order 4 bits are assumed to be zero. If the PCID entered is fewer than 16 digits, then it is right-justified, and a match occurs with all LCBs with PCIDs whose rightmost digits match the specified digits. The specified PCID is not padded with any characters.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for FNDLCB

FNDLCB

```
                        FNDLCB Analysis

  LCB: 06BE4E90
     +0000  40D3C3C2  00000000  06BE4170  00000000  |  LCB.....”.....  |
     +0010  00000000  00000000  00000000  00000000  |  ................  |
     +0020  D3A3D286  D58FA88F  000DD5C5  E3C34BC3  |  LtKfN.y...NETC.C  |
     +0030  F0F4D7F2  F0F8C100  00000000  00000000  |  04P208A.........  |
     +0040  00000002  11100000  00000000  00000000  |  ................  |
     +0050  21110000  00000002  11000000  00000000  |  ................  |
⋮
 Best reply
  PLOCB: 06CE3B00
     +0000  40D7D3D6  00000000  00000000  00000000  |  PLO............  |
     +0010  00000000  00000000  00000000  00000000  |  ................  |
     +0020  00000000  00808000  00000000  00000000  |  ................  |
     +0030  00000000  00000000  00000000  00000000  |  ................  |
     +0040  00000000  A4000040  00000000  D3A3D286  |  ....u.. ....LtKf  |
⋮
 Original reply
  PLOCB: 06BBC2D8
     +0000  40D7D3D6  00000000  00000000  00000000  |  PLO............  |
     +0010  00000000  00000000  00000000  00000000  |  ................  |
     +0020  00000000  40C4A000  000008B6  00000000  |  .... D..........  |
     +0030  0004C1F0  F1D50000  00000004  D5C5E3C1  |  ..A01N......NETA  |
     +0040  00000000  A0000080  00002110  D3A3D286  |  ............LtKf  |
     +0050  D58FA88F  000DD5C5  E3C34BC3  F0F4D7F2  |  N.y...NETC.C04P2  |
⋮
  LCBEXT: 06C6E178
     +0000  D3C3C2C5  10000000  0004D5C5  E3C10000  |  LCBE......NETA..  |
     +0010  00000000  0008C1F3  F1C9F4F8  F9F20000  |  ......A31I4892..  |
     +0020  0004D5C5  E3C10000  00000000  0008C1F3  |  ..NETA........A3  |
     +0030  F1D7F4F8  F9C10000  0004D5C5  E3C10000  |  1P489A....NETA..  |
     +0040  00000000  0004C1F0  F2D50000  00000000  |  ......A02N......  |
     +0050  00000000  00C00000  068F8A38  00000000  |  .....{.........  |
⋮
  LCB: 06BE4170
     +0000  40D3C3C2  06BE4E90  06BE4560  00000000  |  LCB.”+..”.-....  |
     +0010  00000000  00000000  00000000  00000000  |  ................  |
     +0020  CFA9CD86  D295A38D  000DD5C5  E3C34BC3  |  .z.fKnt-..NETC.C  |
     +0030  F0F4D7F1  F6F5C100  00000000  00000000  |  04P165A.........  |
     +0040  00000002  01100000  00000000  00000000  |  ................  |
     +0050  20110000  00000002  01000000  00000000  |  ................  |
⋮
```

FNDNDREC

FNDNDREC scans the topology and route selection database for node records matching the given resource and formats the NDREC control block.

Use FNDNDREC to help diagnose topology and routing problems.

Operands

Resource Name

The resource name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

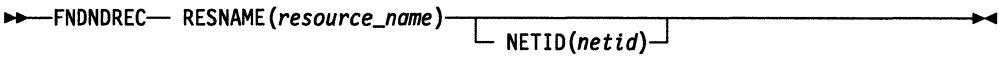
The resource name is required.

Network ID

The network ID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for FNDNDREC

```
FNDNDREC RESNAME(A04P883A) NETID(NETZ)

                          FNDNDREC Analysis

NDREC: 068F3A48
   +0000  D5C4D9C3  0000002B  000DD5C5  E3E94BC1  | NDRC......NETZ.A |
   +0010  F0F4D7F8  F8F3C140  40404000  40000004  | 04P883A    . ...  |
   +0020  00000000  00000000  40000004  00000000  | ........ .......  |
   +0030  00000000  00000000  00000000  00000000  | ................  |
   +0040  00000000  00000000  00000000  00000000  | ................  |
   +0050  00000F00  00000001  00000000  00000000  | ................  |
   +0060  00000000  00000000  00000000  00000000  | ................  |
   +0070  00000000  00000000  12440DD5  C5E3E94B  | ...........NETZ.  |
   +0080  C1F0F4D7  F8F8F3C1  00000000  00000000  | A04P883A........  |
   +0090  00000000  00000000  00000000  00000C45  | ................  |
   +00A0  0A800000  0000FF60  13000000  00000000  | .......-........  |
   +00B0  00000000  00000000  00000000  0000      | ..............    |
```

FNDNODE

Use FNDNODE to format one or more APPN adjacent end nodes and/or adjacent network nodes.

FNDNODE formats and displays the ISTAENCB and ISTANNCB control blocks.

FNDNODE has no required operands.

Operands

Network ID

The network ID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Note: If you specify a resource name but do not specify a network ID, the host network ID will be used to form a fully qualified network name.

Resource Name

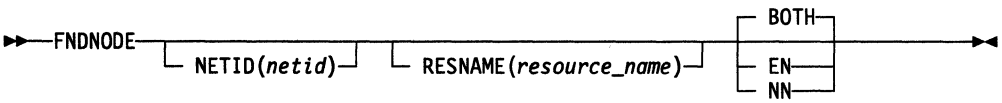
The resource name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Type

Enter BOTH to format both end nodes and network nodes. BOTH is the default. Enter EN to format only end nodes. Enter NN to format only network nodes.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for FNDNODE

FNDNODE

```
                                FNDNODE Analysis

                                 End Node List

  AENCB: 06B24708
     +0000  C5D5C3C2  00000000  06B246A0  00000000  | ENCB.......... |
     +0010  00000000  00000000  000DD5C5  E3C24BC2  | .........NETB.B |
     +0020  F0F1D7F0  F0F9C140  40404000  01000074  | 01P009A    ..... |
     +0030  0100006E  40000000  00000000  00000000  | ...> .......... |
     +0040  00000000  00000000  00000000  00000000  | ................ |
     +0050  06B094D0  00000000  00000000  00000000  | ..m}............ |
:
  AENCB: 06A4A088
     +0000  C5D5C3C2  06A4A0F0  00000000  06A4A5D0  | ENCB.u.0.....uv} |
     +0010  06A4A6A0  00010000  000DD5C5  E3E84BC1  | .uw.......NETY.A |
     +0020  F0F3D7F8  F8F2C140  40404000  00000000  | 03P882A    ..... |
     +0030  00000000  40000000  00000000  00000000  | .... .......... |
     +0040  00000000  00000000  00000000  00000000  | ................ |
     +0050  068F34D0  00000000  00000000  00000000  | ...}............ |

                                Network Node List

  ANNCB: 06A4A020
     +0000  D5D5C3C2  00000000  06A4A2F8  00000000  | NNCB.....us8.... |
     +0010  00000000  00000000  0009D5C5  E3C14BC1  | ..........NETA.A |
     +0020  F8F1D540  40404040  40404040  05000005  | 81N          .... |
     +0030  03000006  40000000  00000000  00000000  | .... .......... |
     +0040  00000000  00000000  00000027  000000A0  | ................ |
     +0050  068F30E8  00000000  00000000  00000000  | ...Y............ |

  ANNCB: 06A4A2F8
     +0000  D5D5C3C2  06A4A020  00000000  00000000  | NNCB.u.......... |
     +0010  06A4A020  FFFF0000  0009D5C5  E3C14BC1  | .u........NETA.A |
     +0020  F0F2D540  40404040  40404040  01000011  | 02N          .... |
     +0030  01000012  40000000  00000000  00000000  | .... .......... |
     +0040  00000000  00000000  00000095  0000009C  | ...........n.... |
     +0050  068F3278  00000000  00000000  00000000  | ................ |
```

FNDREREC

Use FNDREREC to format one or more routing node entries including siblings and children.

FNDREREC formats and displays the ISTREREC control block.

FNDREREC has no required operands.

Operands

Network ID

The network ID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Note: If you specify a resource name but do not specify a network ID, the host network ID will be used to form a fully qualified network name.

Resource Name

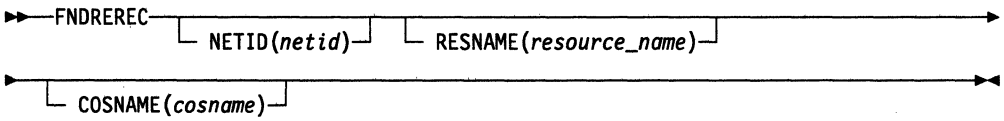
The resource name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Class of Service Name

The name of the entry in the APPN class of service table should be 1–8 alphanumeric characters.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for FNDREREC

```
FNDREREC

                              REREC Analysis

 Parent: NETA.A81N  COSNAME: #CONNECT

  REREC: 06A4F5F8
     +0000  E3D9C5C3  00000000  068F30E8  06A4D0B0  | TREC.......Y.u}. |
     +0010  00000000  06A4F440  00000000  00000000  | .....u4 ........ |
     +0020  00000000  00010000  06A4F440  00000000  | .........u4 .... |
     +0030  00000000  00000000  00000000  00000000  | ................ |
     +0040  00000000  00000000  00000000  00000000  | ................ |

 Sibling chain

 NAME: NETA.A500N

  REREC: 06A4F440
     +0000  E3D9C5C3  06A4F5F8  068F3020  06A4D050  | TREC.u58.....u}& |
     +0010  00000000  00000000  06A3C480  00D20005  | .........tD..K.. |
     +0020  000000D7  00020000  00000000  00000000  | ...P............ |
     +0030  00000000  00000000  00000000  00000000  | ................ |
     +0040  00000000  00000000  00000000  00000000  | ................ |

 No children
```

FNDSCCB

Use FNDSCCB to format all ISTLCBs for a specific search concentration control block.

FNDSCCB formats and displays the ISTLCB control block.

Operands

Network ID

The network ID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Note: If you do not specify a network ID, the host network ID will be used to form a fully qualified network name.

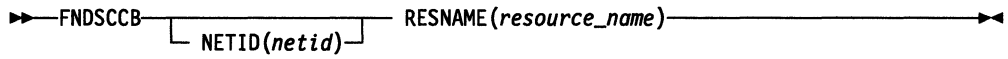
Resource Name

The resource name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

The resource name is required.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for FNDSCCB

```
FNDSCCB RESNAME(B01N)
```

```
                              FNDSCCB Analysis

LCB: 075E2D40
   +0000  40D3C3C2  075BB6B0  075E22C0  07258020  | LCB.$...;.{.... |
   +0010  075E22C0  00000000  0751C640  07521000  | .;.{......F .... |
   +0020  E7F3A765  0E1691B0  0009D5C5  E3C24BC2  | X3x...j...NETB.B |
   +0030  F0F1D500  00000000  00000000  00000000  | 01N............. |
   +0040  00020001  20000000  00000000  00000000  | ................ |
   +0050  00020002  00020000  20000000  00000000  | ................ |
   +0060  00000000  00020000  00020001  20000000  | ................ |
   +0070  00000000  00000000  01000067  0004C2F0  | .............B0 |
   +0080  F1D50000  00000004  D5C5E3C2  00000000  | 1N......NETB.... |
   +0090  80000000  D6E2C3C2  00000000  00000000  | ....OSCB........ |
   +00A0  00000000  00000000  075E2D40  1C000000  | .........;. .... |
   +00B0  0005C1F5  F0F0D500  00000004  D5C5E3C1  | ..A500N.....NETA |
   +00C0  00000000  00000000  00000000  00000000  | ................ |
   +00D0  13280000  00000000  071BAB00  00000000  | ................ |
   +00E0  00000000  00000000  00000000  00000000  | ................ |
   +00F0  00000000  00000000  00000000  00000000  | ................ |
```

FNDSITCB

Use FNDSITCB to help diagnose problems with the session services for LU-LU sessions. For a particular procedure correlation identifier (PCID), PLU name or network identifier, or SLU name or network identifier, FNDSITCB provides the associated SITCB control block.

FNDSITCB has no required operands. If you enter no values for all selection operands, all SITCB control blocks on the queue are eligible for selection.

Operands

PLU Name
The PLU name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

PLU Network ID
The name representing the network ID of another network outside the host network where a resource resides should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

SLU Name
The SLU name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

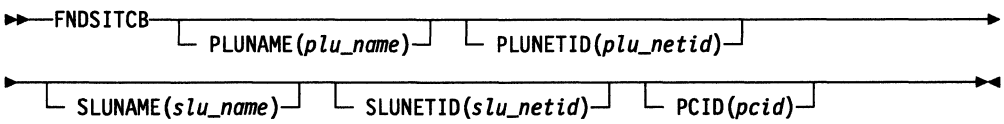
SLU Network ID
The name representing the network ID of another network outside the host network where a resource resides, should be 1–8 alphanumeric characters. If it contains fewer than 8 characters the leftmost characters are compared.

PCID
Specify 1–16 hexadecimal digits in the form X'x...' for the PCID. Specify an even number of digits, otherwise the high-order 4 bits are assumed to be zero. If the PCID entered is fewer than 16 digits, then it is right-justified, and a match occurs with all SITCBs with PCIDs whose rightmost digits match the specified digits. The specified PCID is not padded with any characters.

Note: You might need to find SITCBs for a resource but do not know whether the resource is the PLU or SLU. In this case, you can specify the resource name for both the PLU name and the SLU name, and if the resource name is found in either one, a match occurs.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for FNDSITCB

FNDSITCB

```
                        FNDSITCB Analysis

  SITCB: 06ACBAA0
     +0000  E2C9C3C2  00000000  00D7D5E2  04000000  | SICB.....PNS.... |
     +0010  00000000  00000000  86881220  00000000  | ........fh......  |
     +0020  00000000  00000000  06ACB020  FCD9C3C6  | ............RCF |
     +0030  FCE2D9D8  FCE2D9D8  00000000  FCE2D9D8  | .SRQ.SRQ.....SRQ |
     +0040  00000000  00000000  00000000  00000000  | ................  |
     +0050  00000000  00000000  00000000  00000000  | ................  |
:
  SITCB: 06ACB020
     +0000  E2C9C3C2  00000000  00D7D5E2  04000000  | SICB.....PNS.... |
     +0010  00000000  00000000  86881220  00000000  | ........fh......  |
     +0020  00000000  06ACBAA0  06C24AA0  FCD9C3C6  | .........B¢..RCF |
     +0030  FCE2D9D8  FCE2D9D8  00000000  FCE2D9D8  | .SRQ.SRQ.....SRQ |
     +0040  00000000  00000000  00000000  00000000  | ................  |
     +0050  00000000  00000000  00000000  00000000  | ................  |
:
```

FNDTGREC

Use FNDTGREC to help diagnose topology and routing problems. For an origin control point (CP), FNDTGREC formats the NDREC and TGREC control blocks linked between it and the destination control block.

The name of the destination CP is in the TGREC control block. The name of the origin CP is in the NDREC control block and is a required operand for FNDTGREC. The CP name is in the form of a network identifier and a resource name.

Operands

Origin Resource Name

The origin resource name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

The origin resource name is required.

Origin Network ID

The origin network ID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Destination Network ID

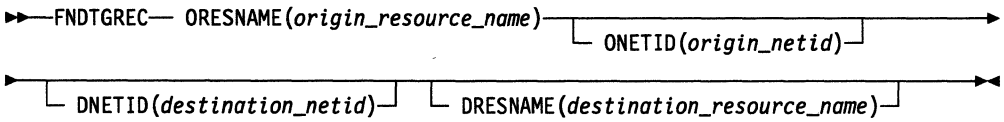
The destination network ID should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Destination Resource Name

The destination resource name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, the leftmost characters are compared.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for FNDTGREC

FNDTGREC ORESNAME(N317408) ONETID(NETA)

```
                              FNDTGREC Analysis

 NDREC: 068F37F0
    +0000  D5C4D9C3  00000024  000CD5C5  E3C14BD5  | NDRC.....NETA.N |
    +0010  F3F1F7F4  F0F84040  40404000  40000004  | 317408     . ... |
    +0020  06A08560  06A08800  40000004  00000000  | ..e-..h. ....... |
    +0030  00000000  00000000  00000000  00000000  | ................ |
    +0040  00000000  00000000  00000000  00000000  | ................ |
    +0050  00000F40  00000001  00000000  00000000  | ... ............ |
    +0060  00000000  00000000  00000000  00000000  | ................ |
    +0070  00000000  00000000  11440CD5  C5E3C14B  | ...........NETA. |
    +0080  D5F3F1F7  F4F0F800  00000000  00000000  | N317408......... |
    +0090  00000000  00000000  00000000  00000C45  | ................ |
    +00A0  0A800000  00468000  23000000  00000000  | ................ |
    +00B0  00000000  00000000  00000000  0000      | ..............   |

 TGREC: 06A08560
    +0000  E3C7D9C3  00000000  06A08640  0F700000  | TGRC......f .... |
    +0010  40000004  00000000  00000000  00000024  |  ............... |
    +0020  068F37F0  068F3020  069E6D40  00000001  | ...0......_ .... |
    +0030  00000000  00000000  00000000  00000000  | ................ |
    +0040  00000000  00000000  00000000  00000000  | ................ |
    +0050  00000000  00000000  16470000  00348076  | ................ |
    +0060  00000000  00000000  204C0000  00000000  | .........<...... |
    +0070  00000000  00000000  00000000  00000000  | ................ |
    +0080  00000000  00001446  12800109  D5C5E3C1  | ............NETA |
    +0090  4BC1F0F1  D5000000  00040000  00000000  | .A01N........... |
    +00A0  00000000  00000000  00000000  00000000  | ................ |
    +00B0  00000000  00000000  00000000  00000000  | ................ |
    +00C0  00000000  00000000  00000000  00000000  | ................ |
    +00D0  00000000  00000000                      | ........         |

 TGREC: 06A08640
    +0000  E3C7D9C3  06A08560  06A08720  0F700000  | TGRC..e-..g..... |
    +0010  40000004  00000000  00000000  00000024  |  ............... |
    +0020  068F37F0  068F30E8  06A07020  00000001  | ...0...Y........ |
    +0030  00000000  00000000  00000000  00000000  | ................ |
    +0040  00000000  00000000  00000000  00000000  | ................ |
    +0050  00000000  00000000  16470000  00260076  | ................ |
    +0060  00000000  00000000  204C0000  00000000  | .........<...... |
    +0070  00000000  00000000  00000000  00000000  | ................ |
    +0080  00000000  00001446  12800109  D5C5E3C1  | ............NETA |
    +0090  4BC1F0F2  D5000000  01360000  00000000  | .A02N........... |
    +00A0  00000000  00000000  00000000  00000000  | ................ |
    +00B0  00000000  00000000  00000000  00000000  | ................ |
    +00C0  00000000  00000000  00000000  00000000  | ................ |
    +00D0  00000000  00000000                      | ........         |
⋮
```

HOST

Use HOST to determine the following for the VTAM host from which a dump was taken:

- Address space ID (ASID)
- CP name
- CP network address
- Host CDRM name
- Network ID
- Node type
- SSCP network address
- Whether the CDRM supports CDRSCs
- Whether the SSCP is gateway-capable.

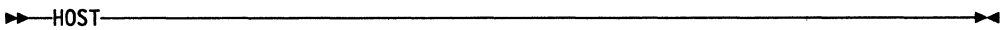
This information can provide a valuable point of reference for network problem diagnosis.

Operands

None.

Use the following syntax as an alternative to the panel interface.

Syntax



Equated Symbol

Symbol	Description
ATCVT	VTAM's ATCVT.
ISTSRTDcdrmname	Derived from the CDRM RDTE for the host (for example, for cdrmname SSCP1A, the ISTSRTDcdrmname symbol would be ISTSRTDSSCP1A).

Sample Output for HOST

```
HOST

                              HOST Analysis

  NetID                NETA
  ASID (Hex)           001D
  ASID (Dec)             29
  Subarea (Hex)    0000000D   Element  0001
  Subarea (Dec)          13   Element     1
  CDRM Name            A13N

  This SSCP is gateway capable
  This CDRM supports dynamic CDRSCs

  CP Network address  0000000D 0006
  CP Name             NETA.A13N
  This is a pure network node
```

ISTVABND

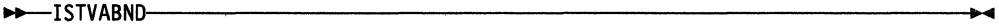
ISTVABND determines the following in an MVS dump of a VTAM abend:

- System completion code
- Program interrupt code
- Instruction length code
- Translation exception address
- PSW
- Abending module name, displacement, PTF level
- Failing instruction
- Registers at time of abend
- VTAM save area chain (forward and backward)
- Symptom string information.

IPCS symbols for each register and the PSW address are created. After ISTVABND executes, storage pointed to by the registers and PSW can be accessed by using these IPCS symbols.

The ISTVABND command can also be issued from the panel interface.

Syntax



Equated Symbols

Symbol(s)	Description
R0, REG0	Register 0
R1, REG1	Register 1
R2, REG2	Register 2
R3, REG3	Register 3
R4, REG4	Register 4
R5, REG5	Register 5
R6, REG6	Register 6
R7, REG7	Register 7
R8, REG8	Register 8
R9, REG9	Register 9
R10, REG10, RA, REGA	Register 10
R11, REG11, RB, REGB	Register 11
R12, REG12, RC, REGC	Register 12
R13, REG13, RD, REGD	Register 13
R14, REG14, RE, REGE	Register 14
R15, REG15, RF, REGF	Register 15
PSW	PSW address
MODULE NAME	Module that called ISTSSCZZ

Note: If the abend was the result of an ABEND0A9 issued by module ISTSSCZZ, the registers at the time of the call to ISTSSCZZ (rather than the registers when ISTSSCZZ issued the ABEND0A9) are used to create the symbols listed above. Also, the module name, displacement, and PTF level of the module that called ISTSSCZZ are displayed.

Additional Information

An abend can occur in SRB mode or in TCB mode.

For an MVS dump of an abend in TCB mode, ISTVABND locates the abending TCB and the RTM2WA. The completion code (system or user) in the TCB is analyzed and displayed. From the RTM2WA, the program interrupt code (PIC), instruction length code (ILC), and translation exception address (TEA) if valid, are analyzed and displayed.

For an abend in SRB mode, there is no RTM2WA. The PIC, ILC, TEA, registers, and PSW are taken from the SDWA which is found in the MVS FRR stack.

The PSW address is used to determine the abending module name, displacement, and PTF level. Register 13 (which usually contains a pointer to the abending module's save area) is used to trace the save area chain forward and backward (by calling ISTVSAVE) to show module linkage.

Symptom string information is obtained from the variable recording area (VRA) of the SDWA.

Sample Output from ISTVABND

The following is a sample of the output from ISTVABND for an ABEND0C4 in module ISTTRTLR.

ISTVABND

```
CLIST ISTVABND STARTED AT 09:06:51.

(ISTVABND) THIS DUMP WAS THE RESULT OF AN ABEND IN SRB MODE

SYSTEM COMPLETION CODE = 0C4
PROGRAM INTERRUPT CODE = 0010    INSTRUCTION LENGTH CODE = 0004

PSW AT TIME OF ABEND: 076C2000 82DE96FA
TRANSLATION EXCEPTION ADDRESS = 30580038
THE FAILING INSTRUCTION IS: 43603001
ISTAPCFR-VTAM FRR DUMP

 VTAMMAP input data
 VTFNDMOD SYMBOL(PSW) NNTERNAL

Module name:              ISTTRTLR
Compile date:             92.224
Address entered:          02DE96FA
Module entry point:       02DE9558
                          --------
Displacement into module:      1A2

First '40'X bytes of module:
 DATA: 02DE9558
    +0000  47F0F014  0FC9E2E3  E3D9E3D3  D940F9F2  | .00..ISTTRTLR 92 |
    +0010  4BF2F2F4  90ECD00C  05C018FD  5860F000  | .224.Ö}..{...-0. |
    +0020  58D06000  1F99BF97  C40F58A0  C40A187D  | .}-..r†pD..ffD..' |
    +0030  1E791E7A  50706000  5880C3CE  18B714B8  | . .:&.-...Có.¼.½ |

Storage around address entered:
 DATA: 02DE96E6
    +0000  5840806C  41500002  1E544130  50304190  | . .%.&......&... |
    +0010  00021F66  43603001  1E69D200  30013000  | .....-....K..... |
    +0020  42603000  4130504F  1F664360  30011E69  | .-....&|...-.... |
    +0030  D2003001  30004260  30004110  D0681E94  | K......-....}..m |

REGISTERS AT TIME OF ABEND:
REG0 = 82D9FE00   REG1 = 028F8010   REG2 = 00000000   REG3 = 3058003A
REG4 = B0580008   REG5 = B058000A   REG6 = 00000000   REG7 = 00C49D40
REG8 = 00000002   REG9 = 00000002   REGA = 00000008   REGB = 00000110
REGC = 82DE9572   REGD = 02EC9EA0   REGE = 82DE974C   REGF = 00000000

SAVE AREA CHAIN (STARTING WITH SAVE AREA AT 02EC9EA0):
ACRT -> SSTM -> ISTT -> CPNQ
CURRENT SAVE AREA = ACRT

SYMPTOM STRING:
AB/S00C4 LVLS/410 RIDS/ISTAPCFR#R PIDS/5695-11701 ADRS/000001A2
RIDS/ISTTRTLR
LVLS/92.224 REGS/0C188

CLIST ISTVABND ENDED AT 09:07:16. RETURN CODE = 0.
```

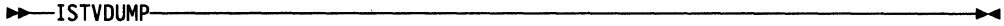
ISTVDUMP

ISTVDUMP determines the SDATA options in effect when an MVS dump occurs. The SDATA options determine which MVS storage areas are requested to be dumped when the dump is taken by VTAM (SDUMP) or requested by the operator (console dump). ISTVDUMP can thus help you determine why a specific address is not in an MVS dump.

Note: ISTVDUMP shows what areas were requested for a dump. However, because the area was requested does not guarantee that information is in the dump. If an area is missing from your dump, it can be due to other reasons (for example, data is lost transferring the dump from the dump data set to tape, or the dump data set is too small, causing a partial dump to be taken).

The ISTVDUMP command can also be issued from the panel interface.

Syntax



Additional Information

When you are working with an MVS dump of VTAM, the following information may be useful:

- The PSA must have been dumped to access low-core address hexadecimal 408.
- CSA must have been dumped to access the ATCVT.
- VTAM private storage must have been dumped (RGN parameter specified when the dump is taken) to access most VTAM modules and control blocks.

If neither CSA nor RGN is requested for a dump, ISTVDUMP issues a warning message. See the sample output in "Sample Output from ISTVDUMP" on page 285 for an example. To resolve most VTAM problems, you must have the VTAM private region and CSA.

When the dump is taken, ISTVDUMP analyzes the RTM recovery termination control table (RTCT) and the SDUMP parameter list (SDUMP) to determine what was requested on the SDATA operand.

For a stand-alone dump obtained by AMDSADMP, the pointer to the RTCT is zero. If you run ISTVDUMP against a stand-alone dump, a message is issued indicating that the RTCT pointer is zero and the CLIST stops processing. See Table 80 on page 1165 to determine the book that contains information on SDATA options, RTCT, and SDUMP.

The following list shows all of the possible settings of the SDATA flags in the SDUMP parameter list. Note that there is no specific indication for extended areas (above 16MB). When an area is requested (for example, RGN), it is dumped, as is the extended area if present.

SDUMP Parameter List

Flag	Description
SDUALPSA	Dump all PSAs in the system.
SDUPSA	Dump the current PSA.
SDUNUC	Dump the nucleus.
SDUSQA	Dump SQA.
SDULSQA	Dump LSQA.
SDURGN	Dump region (private area).
SDULPA	Dump active LPA module for RGN.
SDUTRT	Dump trace table and GTF buffers.
SDUCSA	Dump CSA.
SDUSWA	Dump SWA for region.
SDUSMDMP	Summary dump requested.
SDUNSMDP	Do not dump summary dump.
SDUNSPSA	Do not dump all PSA.
SDUNASQA	Do not dump SQA.
SDUALNUC	Dump all nucleus areas.

Sample Output from ISTVDUMP

In this sample, note that CSA and RGN (SDATA option SDURGN) were both requested.

ISTVDUMP

```
CLIST ISTVDUMP STARTED AT 13:36:44.

SDATA OPTIONS REQUESTED FOR THIS DUMP:

  SDUALPSA - DUMP ALL PSA'S IN THE SYSTEM
  SDUPSA   - DUMP THE CURRENT PSA
  SDUNUC   - DUMP THE NUCLEUS
  SDUSQA   - DUMP SQA
  SDULSQA  - DUMP LSQA
  SDURGN   - DUMP REGION (PRIVATE AREA)
  SDULPA   - DUMP ACTIVE LPA MODULE FOR RGN
  SDUTRT   - DUMP TRACE TABLE / GTF BUFFERS
  SDUCSA   - DUMP CSA
  SDUSWA   - DUMP SWA FOR REGION
  SDUSMDMP - SUMMARY DUMP REQUESTED
  SDUALNUC - DUMP ALL NUCLEUS AREAS

CLIST ISTVDUMP ENDED AT 13:36:44. RETURN CODE = 0.
```

In this sample, RGN and CSA were not requested when the dump was taken.

ISTVDUMP

```
CLIST ISTVDUMP STARTED AT 15:32:17.

SDATA OPTIONS REQUESTED FOR THIS DUMP:

  SDUALPSA - DUMP ALL PSA'S IN THE SYSTEM
  SDUNUC   - DUMP THE NUCLEUS
  SDUSQA   - DUMP SQA
  SDULSQA  - DUMP LSQA
  SDULPA   - DUMP ACTIVE LPA MODULE FOR RGN
  SDUTRT   - DUMP TRACE TABLE / GTF BUFFERS
  SDUSWA   - DUMP SWA FOR REGION
  SDUSMDMP - SUMMARY DUMP REQUESTED

*********************************
* PRIVATE REGION WAS NOT DUMPED *
*********************************
**********************
* CSA WAS NOT DUMPED *
**********************

CLIST ISTVDUMP ENDED AT 15:32:18. RETURN CODE = 0.
```

ISTVMAP

Use ISTVMAP to determine the starting and ending addresses and area size of the following major MVS/ESA storage areas in a dump:

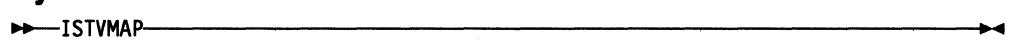
- Extended maximum possible region
- Extended private region
- Extended CSA
- Extended MLPA
- Extended FLPA
- Extended PLPA
- Extended SQA
- Extended read/write nucleus
- Extended read/only nucleus
- CSA
- MLPA
- FLPA
- PLPA
- SQA
- Read/only nucleus
- Read/write nucleus
- Maximum possible region
- Private region
- Low storage.

When you cannot find an address in a dump, the starting and ending addresses of major MVS/ESA storage areas in the dump will help you determine if and where that address is in the dump.

Also use ISTVMAP when areas of storage needed to diagnose a VTAM problem do not appear to be in the dump. Knowing which storage area a given address represents and what was actually dumped can be helpful in determining why a specific storage address is not in a dump. See ISTVDUMP to determine which storage areas were requested to be dumped when the dump was taken.

The ISTVMAP command can also be issued from the panel interface.

Syntax



Additional Information

The storage in an MVS/ESA system is mapped by the CVT virtual storage address extension, the global data area (GDA), and the local data area (LDA). ISTVMAP uses these areas to produce a map of storage that is unique to the dump being processed. The map does not apply to any other dump.

Sample Output from ISTVMAP

ISTVMAP

```
CLIST ISTVMAP STARTED AT 21:37:40.

 START      END       SIZE          VIRTUAL STORAGE AREA
 --------   --------  --------  ------------------------------
 06500000  7FFFFFFF  79B00000  EXTENDED MAXIMUM POSSIBLE REGION
 06500000  084FFFFF  02000000  EXTENDED PRIVATE REGION
 02624000  064FFFFF  03EDC000  EXTENDED CSA
 025FA000  02623FFF  0002A000  EXTENDED MLPA
 025F7000  025F9FFF  00003000  EXTENDED FLPA
 01C13000  025F6FFF  009E4000  EXTENDED PLPA
 01311000  01C12FFF  00902000  EXTENDED SQA
 012A9000  01310FFF  00068000  EXTENDED READ/WRITE NUCLEUS
 01000000  012A8E4F  002A8E50  EXTENDED READ/ONLY NUCLEUS
 --------------------16M LINE--------------------
 00FDE000  00FFFFFF  00022000  READ/ONLY NUCLEUS
 00F3D000  00FDDABF  000A0AC0  READ/WRITE NUCLEUS
 00E6D000  00F3CFFF  000D0000  SQA
 00C98000  00E6CFFF  001D5000  PLPA
 00C97000  00C97FFF  00001000  FLPA
 00C92000  00C96FFF  00005000  MLPA
 00800000  00C91FFF  00492000  CSA
 00005000  007FFFFF  007FB000  MAXIMUM POSSIBLE REGION
 00005000  00414FFF  00410000  PRIVATE REGION
 00000000  00004FFF  00005000  LOW STORAGE

CLIST ISTVMAP ENDED AT 21:37:43. RETURN CODE = 0.
```

ISTVSAVE

ISTVSAVE follows a VTAM save area (VWA) chain forward and backward, starting from the specified save area address.

Using the save area address that was entered, ISTVSAVE follows the forward save area chain until it encounters a zero or not valid forward chain pointer. Then, starting again at the original save area address that was entered when ISTVSAVE was invoked, ISTVSAVE follows the backward save area chain until it encounters a zero or not valid backward chain pointer. If an error occurs during the attempt to access storage, the save area chain in that direction (forward or backward) is assumed to end.

The VWA eye-catchers are displayed, separated by arrows (->) indicating the order of VTAM module linkage. If ISTVSAVE encounters a save area with no (or a not valid) eye-catcher, the output for that eye-catcher may contain unprintable characters or periods (for example, SSUW -> SSZZ -> ....). The "...." means that the first word of the save area does not contain a standard VTAM save area eye-catcher.

Operands

You can specify an address or symbol pointing to any valid VWA. You must enter at least one of the following when you invoke ISTVSAVE:

address
The address of a VTAM module save area should be 1–8 hexadecimal digits.

symbol
A previously equated IPCS symbol that points to a VTAM save area.

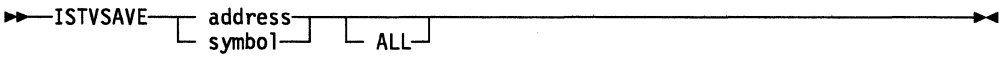
X If the current address being displayed points to a VTAM save area, the IPCS symbol **X** can be used to represent it.

Optionally, you can specify:

ALL
To list the register save area for each save area on the chain.

The ISTVSAVE command can also be issued from the panel interface.

Syntax



Additional Information

Most VTAM modules use standard register-save-area linkage. The first word of a register save area is optional, and some VTAM modules store a 4-character identifier there. These identifiers are the VWA eye-catchers, which are displayed by ISTVSAVE. In most cases, VTAM MVS module names have the form ISTxxCxx, and the identifier consists of the 4th, 5th, 7th, and 8th characters of the name of the VTAM module that owns the save area.

```
Example:   VTAM Module Name = ISTACCRT  VWA eye-catcher = C'ACRT'
           VTAM Module Name = ISTSSCTM  VWA eye-catcher = C'SSTM'
```

If the save area does not follow these rules, it should follow the enhanced save area chaining convention. If it follows neither, the results of this CLIST are unpredictable. For the enhanced save the chain, the first 3 characters are IST followed by the module name. The address pointed to by register 13 will always point back to the save area chain. For a discussion of both methods, see "Using Save Area Module Linkage Conventions—Subarea" on page 552.

Sample Output from ISTVSAVE

The following is a sample of the output from ISTVSAVE for an ABEND0A9 in module ISTSSCZZ.

ISTSSCZZ is the SSABEND macro processor. The purpose of this module is to issue the ABEND0A9. It is necessary to know the caller of ISTSSCZZ to diagnose the ABEND0A9 properly. In this example, the caller of ISTSSCZZ was ISTSSCUW.

The "current save area" is the save area pointed to by the address entered when ISTVSAVE was invoked (that is, the save area for module ISTSSCUW is at address 0B2B0480, which is the address where IPCS was positioned when ISTVSAVE was invoked using the symbol X).

ISTVSAVE R13 ALL

```
SAVE AREA CHAIN (STARTING WITH SAVE AREA AT 0B2B0480):

ACRT -> SSTM -> SSTP -> SSKT -> SSU3 -> SSUW -> SSZZ -> ....

CURRENT SAVE AREA = SSUW
```

The following is a sample of the output from ISTVSAVE for an ABEND0C4 in module ISTDECH2. This is a particularly long save area chain.

ISTVSAVE 981DE88

```
SAVE AREA CHAIN (STARTING WITH SAVE AREA AT 0981DE88):

ACRT -> ACRR -> DEST -> DESD -> DEIS -> DEP2 -> DESA -> DESF -> DEQR ->
DESF -> DESJ -> DEP2 -> DEVP -> DEK2 -> DESG -> DESB -> DESC -> DEIS ->
DEK2 -> DEVP -> DEG2 -> DEVP -> DER3 -> DEH2 -> . J. -> .00.

CURRENT SAVE AREA = DEH2
```

The following is sample output from ISTVSAVE with R13 specified as the *symbol* operand. R13 represents address 04FB53C8.

ISTVSAVE R13 ALL

```
(ISTVSAVE) CLIST WAS INVOKED WITH ADDRESS/SYMBOL 'R13'
(ISTVSAVE) SYMBOL 'R13' REPRESENTS ADDRESS 04FB53C8

CURRENT SAVEAREA:

SAVE AREA FOR SSNP
04FB53C8.                   E2E2D5D7 04FB5360 |        SSNP...-|
04FB53D0. 04FB5448 00000000 00000000 00000000 |................|
04FB53E0 LENGTH(48)==>All bytes contain X'00'
04FB5410. 84A3968E                            |dto.            |

SAVEAREA(S) ENCOUNTERED FOLLOWING THE FORWARD SAVEAREA CHAIN:

SAVE AREA FOR ....
04FB5448 LENGTH(76)==>All bytes contain X'00'

SAVEAREA(S) ENCOUNTERED FOLLOWING THE BACKWARD SAVEAREA CHAIN:

SAVE AREA FOR SSTD
04FB5360. E2E2E3C4 04FB52F0 04FB53C8 84A75C4C |SSTD...0...Hdx*<|
04FB5370. FFA395C8 00000000 04A99CD8 04FB5224 |.tnH.....z.Q....|
04FB5380. 04AACF30 04A521F8 00000015 04A395C8 |.....v.8.....tnH|
04FB5390. 04A521F8 04FB5348 04AACF30 04A99CD8 |.v.8.........z.Q|
04FB53A0. 04AACF08 84A759FA 84A75AAE          |....dx..dx!.    |

SAVE AREA FOR SSTV
04FB52F0. E2E2E3E5 04FB5288 04FB5360 84A76856 |SSTV...h...-dx..|
04FB5300. FFA759E0 00000000 04A99CD8 04FB5224 |.x.\.....z.Q....|
04FB5310. 04FB5224 04AACF30 04FB520C 04FB5348 |................|
04FB5320. 05003EEE 04FB5360 04A51598 04A99CD8 |.......-.v.q.z.Q|
04FB5330. 04AACF08 84A766F2 84A767AE          |....dx.2dx..    |

SAVE AREA FOR SSTD
04FB5288.                   E2E2E3C4 04FB5170 |        SSTD....|
04FB5290. 04FB52F0 84A75D6A FFA766D8 00000000 |...0dx)|.x.Q....|
04FB52A0. 04A99CD8 04FB5224 04AACF30 04FB520C |.z.Q............|
04FB52B0. 00000015 04AACF30 04FB520C 00000015 |................|
04FB52C0. 04A51598 04A99CD8 04AACF48 84A759FA |.v.q.z.Q....dx..|
04FB52D0. 00000000                            |....            |

SAVE AREA FOR SSTM
04FB5170. E2E2E3D4 04FB5010 04FB5288 84A41A8A |SSTM..&....hdu..|
04FB5180. FFA759E0 04AACF48 04A99CD8 04A51598 |.x.\.....z.Q.v.q|
04FB5190. 00000000 04A51598 0000001D 04A51598 |.....v.q.....v.q|
04FB51A0. 04AACF30 05003EE6 04FB5224 04A99CD8 |.......W.....z.Q|
04FB51B0. 04AACF48 84A41192 84A4127A          |....du.kdu.:    |

SAVE AREA FOR ACRT
04FB5010. C1C3D9E3 00000000 04FB5170 8498E214 |ACRT........dqS.|
04FB5020. FFA41170 04A99CD8 04A99CD8 00C35558 |.u...z.Q.z.Q.C..|
04FB5030. 84A0D988 04FB5010 00000000 84FB5008 |d.Rh..&.....d.&.|
04FB5040. 04A99228 04A99228 04A99228 04A99CD8 |.zk..zk..zk..z.Q|
04FB5050. 8498CD9E 0498DD9D 04A99228          |dq...q...zk.    |

SAVE AREA CHAIN (STARTING WITH SAVE AREA AT 04FB53C8):

ACRT - SSTM - SSTD - SSTV - SSTD - SSNP - ....
CURRENT SAVE AREA = SSNP
```

ISTVSLIP

Use ISTVSLIP to display the registers and PSW that were current at the time of an SLIP dump. The registers and PSW are extracted from the SDUMP buffer pointed to by the CVT.

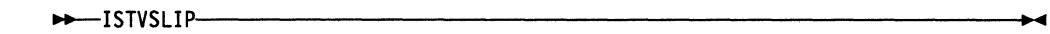
All 16 general registers and the PSW are displayed, along with the module name and displacement that the address portion of the PSW represents (the module in control at the time the SLIP trap occurred). The module name and displacement that register 14 represents (usually the calling module or within the current module) are also displayed. If the dump was not taken as a result of an SLIP trap, a message to that effect is displayed.

IPCS symbols are created for each register and the address portion of the PSW. After ISTVSLIP has executed, storage locations pointed to by these registers (or PSW) can be displayed using these symbols in the IPCS LIST command. For example, **L R4** will display the storage pointed to by register 4 at the time the SLIP trap occurred.

Note: ISTVSLIP does not support stand-alone dumps taken after the SLIP ACTION=WAIT MVS system command is issued.

The ISTVSLIP command can also be issued from the panel interface.

Syntax



Equated Symbols

Symbol(s)	Description
R0, REG0	Register 0
R1, REG1	Register 1
R2, REG2	Register 2
R3, REG3	Register 3
R4, REG4	Register 4
R5, REG5	Register 5
R6, REG6	Register 6
R7, REG7	Register 7
R8, REG8	Register 8
R9, REG9	Register 9
R10, REG10, RA, REGA	Register 10
R11, REG11, RB, REGB	Register 11
R12, REG12, RC, REGC	Register 12
R13, REG13, RD, REGD	Register 13
R14, REG14, RE, REGE	Register 14
R15, REG15, RF, REGF	Register 15
PSW	PSW address

Additional Information

The dump data set name is displayed to verify that the correct dump is being processed. The title of the dump is displayed for additional verification that the dump being processed is, in fact, an ISTVSLIP dump.

To determine the module name, displacement, and PTF level for the PSW and register 14, VTFNDMOD is called. The PSW address and register 14 are used as input to VTFNDMOD.

Sample Output from ISTVSLIP

ISTVSLIP

```
CLIST ISTVSLIP STARTED AT 17:42:19.

DUMP DATASET NAME: IPCS.P620527.DUMPA
TITLE FROM DUMP:   SLIP DUMP ID=0001
THE ADDRESS OF THE SDUMP BUFFER IN THE CVT IS 00C95000.
PRIMARY ASID AT THE TIME OF ENTRY TO SLIP IS X'000A'

PSW AT ENTRY TO RTM: 00000000 01D03790

REGISTERS WHEN SLIP TRAP MATCHED:
REG0 = 08000000   REG1 = 080A9000   REG2 = 00000012   REG3 = 00000012
REG4 = 007FF158   REG5 = 80C1EFC8   REG6 = 00003000   REG7 = 00000004
REG8 = 0000000B   REG9 = 81E6190C   REGA = 0650CE40   REGB = 81E51A98
REGC = 81D0347A   REGD = 0650CE40   REGE = 01F2A714   REGF = 010E35E0

(ISTVSLIP) PROCESSING OF PSW FOLLOWS:

 VTAMMAP input data
 VTFNDMOD ADDR(X'01D03790') NNTERNAL

Module name:               ISTCFF3D
Compile date:              92.318
Address entered:           01D03790
Module entry point:        01D03462
                           --------
Displacement into module:       32E

First '40'X bytes of module:
 DATA: 01D03462
    +0000  47F0F016  10C9E2E3  C3C6C6F3  C44040F9  | .00..ISTCFF3D  9 |
    +0010  F24BF3F1  F80005C0  185F1861  187041F0  | 2.318..{.¬./...0 |
    +0020  00005800  CB7E47F0  C0160000  020089F0  | .....=.0{.....i0 |
    +0030  0008BFFD  C0121B11  0A7818A1  18161807  | ..†.{...... .... |

Storage around address entered:

 DATA: 01D0377C
    +0000  89000018  16101823  58F00010  58F0F034  | i........0...00. |
    +0010  58F0F020  05EF47F0  C5445850  04081255  | .00....0E..&.... |
    +0020  4780C34A  91405488  4780C34A  58205B50  | ..C¢j .h..C¢..$& |
    +0030  12224780  C34A9102  54884710  C34A5850  | ....C¢j..h..C¢.& |

Address is in extended pageable LPA (above the 16M line).
Extended pageable LPA starting address: 01C13000
Extended pageable LPA ending address:   025F8FFF

(ISTVSLIP) PROCESSING OF REG 14 FOLLOWS:

VTAMMAP input data

VTFNDMOD ADDR(X'1F2A714')

Module name:               ISTDSCGD
Compile date:              92.256

Address entered:           01F2A714
Module entry point:        01F2A6A0
                           --------
Displacement into module:       74
```

```
First '40'X bytes of module:

DATA: 01F2A6A0
    +0000  47F0F014  0FC9E2E3  C4E2C3C7  C440F9F2  | .00..ISTDSCGD 92 |
    +0010  4BF2F5F6  90ECD00C  18BF41F0  00005800  | .256..}....0.... |
    +0020  B95C0700  47F0B02C  00000200  89F00008  | .*...0......i0.. |
    +0030  BFFDB028  1B110A78  18C150D0  C00450C0  | .........A&}{.&{ |

Storage around address entered:

 DATA: 01F2A700
    +0000  B93E58A0  ACE45630  B8E858F0  A0005820  | .....U...Y.0.... |
    +0010  B8F805EF  980E1028  58201020  12FF4770  | .8..q........... |
    +0020  B094D203  2000B92C  58E04000  50E02024  | .mK......\ .&\.. |
    +0030  47F0B098  41F00008  12FF4770  B52041A0  | .0.q.0.......... |

Address is in extended pageable LPA (above the 16M line).
Extended pageable LPA starting address: 01C13000
Extended pageable LPA ending address:   025F8FFF

(ISTVSLIP) THE REGISTER SAVE AREA CHAIN COULD NOT BE ANALYZED.
           REG 13 IS NOT VALID OR STORAGE IS NOT AVAILABLE.

CLIST ISTVSLIP ENDED AT 17:43:30. RETURN CODE = 0.
```

PABSCAN

A request/response unit processing element (RUPE) represents the unit of work VTAM must perform for a given request or response received from the network. VTAM queues RUPEs to a process anchor block (PAB). Knowing what type of work is queued to a PAB may be important in resolving storage and performance problems. Use PABSCAN to scan a chain of RUPEs queued to a PAB and obtain a summary of the RUPEs by RU type.

You may scan all work elements on the PAB, or limit the search to RUPEs containing a specific value in one or more of the following fields:

- Destination address field (DAF)
- Origin address field (OAF)
- CPCB operation code (CPCBOPC)
- Request/Response unit (RUPERQD,RUPERSD)
- A user-specified location within the RUPE.

For each work element that is selected, the RU is extracted and counted. After all of the selected work elements have been counted, a summary showing the number of work elements containing each RU type is displayed.

Operands

You must specify one address or one symbol to represent the first RUPE in the chain of RUPEs to be analyzed.

Address
Enter 1–8 hexadecimal digits in the form X'x...' for the address of the chain of RUPEs to be analyzed. If the address is fewer than 8 digits, it is padded on the left with zeros.

IPCS Symbol
Enter an IPCS symbol name that is 1–31 alphanumeric characters. The symbol name represents the beginning of a chain of RUPEs. Do not include a period.

Under IPCS, the symbol X represents the address currently being displayed. If the current address is pointing to a chain of RUPEs, this symbol may be used.

If you specify no other selection operands, the first 100 RUPEs are analyzed and a summary of the RUPEs by RU type is displayed.

If you use more than one of the following operands, all of the selection criteria must be met for a RUPE to be selected.

Destination Address Field
Only RUPEs containing this destination address field (RUPEDAF) are eligible for selection. Specify 1–12 hexadecimal digits in the form X'x...'. If the address is fewer than 12 digits, the rightmost digits are compared.

Origin Address Field
Only RUPEs containing this origin address field (RUPEOAF) are eligible for selection. Specify 1–12 hexadecimal digits in the form X'x...'. If the address is fewer than 12 digits, the rightmost digits are compared.

Control Op Code
Only RUPEs containing this CPCB op code are eligible for selection. The Control Op Code must be 1–8 hexadecimal digits in the form X'x...'. If the op code is fewer than 8 digits, it is left-justified and compared with the leftmost digits in the dump.

Request/Response Unit
Only RUPEs that contain this RU are eligible for selection. The leftmost digits of field RUPERSD are compared (if RUPERSP is nonzero) to the value entered, otherwise the leftmost digits of RUPERQD (if RUPERQP is nonzero) are compared to the value entered. The length used for the comparison is the length of the value entered.

Detail
The default is N. Specify Y to have the following fields extracted and displayed for each RUPE meeting the selection criteria:

- Position of RUPE on the PAB
- RUPE address
- CPCB op code (CPK)
- Origin address field (RUPEOAF)
- Destination address field (RUPEDAF)
- First 4 bytes of request/response unit (if present)
- User-data at a specified displacement (if Displacement and Value are specified).

One line of output per RUPE is produced.

Max
Specify the maximum number of RUPEs to be processed. The default for MAX is 100. If MAX is not specified, only the first 100 RUPEs on the PAB are analyzed. The maximum value for MAX is 99999. If the maximum number of RUPEs are processed and more remain on the PAB, PABSCAN will report the number of unprocessed elements remaining on the PAB.

Displacement
Enter the displacement into the RUPE where Value is to be found. The maximum decimal displacement is 4095 and the maximum hexadecimal displacement is X'FFF'.

Length
Enter a value of 1–8 for the number of bytes you want displayed, starting at the displacement specified in Displacement.

Length must be used with the Displacement operand. Together, they display any portion of a RUPE. The Length operand cannot be used with the Value and Value Type operands.

Note: The following two operands, Value and Value Type, must be used together with the Displacement operand. They allow any field in a RUPE to be checked for a user-specified value. The Value and Value Type operands cannot be used with the Length operand.

Value
Only RUPEs containing this data at the displacement specified in Displacement are eligible for selection.

Value may contain character or hexadecimal data of 1–8 bytes in length. Character data should consist of alphanumeric characters. Hexadecimal data should contain an even number of up to 16 hexadecimal digits in the form X'xx...'; otherwise, the high order half-byte is assumed to be zero.

Binary data can be used to look at a particular bit within a byte. You may specify one byte of binary data in the form X'xx'. Only 1 bit within the byte may be selected. Therefore, you can specify only the following hexadecimal values: 01, 02, 04, 08, 10, 20, 40, and 80. A value with more than 1 bit set (for example, 82) will not be processed.

Value Type
Enter B for binary, C for character, or X for hexadecimal to indicate the type of data entered for Value.

Additional Information

To determine whether an RU is a request or a response, PABSCAN first checks field RUPERSP.

If RUPERSP is nonzero, the RU is considered to be a response. The designation RSP, along with the contents of field RUPERSD, are used to represent the response.

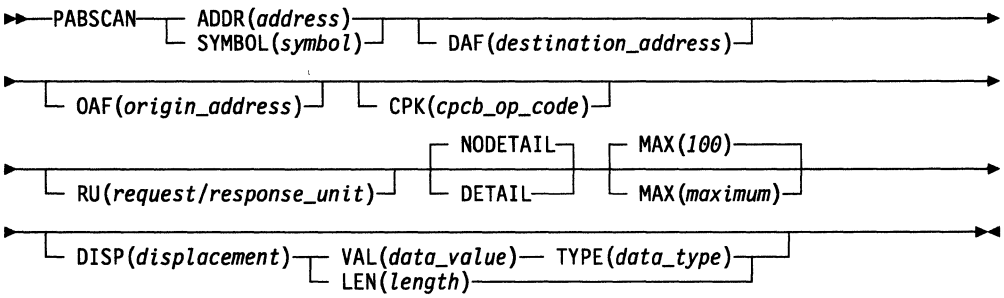
If RUPERSP is zero, field RUPERQP is then checked.

If RUPERQP is nonzero, the RU is considered to be a request. The designation REQ, along with the first 4 bytes of the RU from field RUPERQD, are used to represent the request (using 4 bytes allows for the largest of the RU headers and also picks up the format byte for RU headers which are 3 bytes long).

If both RUPERQP and RUPERSP are zero, the designation NORU, along with the contents of field CPCBOPC, are used to represent the RU. The designation NORU notes the fact that no RU (neither request nor response) pointer existed in the RUPE or the length of the RU was zero.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for PABSCAN

PABSCAN SYMBOL(X)

```
                              PABSCAN Analysis
ELEM# RUPEADDR CPCBOPC    RUPEOAF      RUPEDAF       RU        USERDATA
----- -------- -------- ------------ ------------ -------- ----------------
    1 062E3028 00000000 000000010003 000000010003 **NORU**
00000000 REQ  occurred          1
RUPEs left on the chain         0
Elements processed              1
```

PARTNRLU

Use PARTNRLU to display all partner LUs for an APPC application. PARTNRLU formats and displays the APPCB control block, the COPR control block if present, and the LME.

Operands

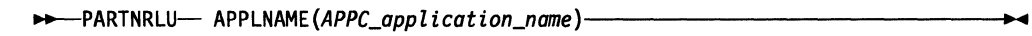
APPC Application Name

The APPC application name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks.

The APPC application name is required.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for PARTNRLU

```
PARTNRLU APPLNAME(APPCAP05)

                        PARTNRLU Analysis
APPCB: 0290E6B8
   APPLUCB.. 0291A100  APPTSKID. 02818588  APPACB... 00CB4820
   APPLUN... APPCAP05
   APPSPTAE  02906530 0290D898 02906620 029065D0 02906580
DATA: 0290E6B8
   +0000  62C1D7D7  0291A100  02818588  00000000  | .APP.j ..aeh.... |
   +0010  00000000  00000000  024100B0  00000000  | ...........[.... |
   +0020  31094000  00000010  00000000  00000000  | .. ............. |
   +0030  0101001B  00000000  00CB4820  0290B088  | ..............[h |
   +0040  00000000  00000000  00000000  00000000  | ................ |
   +0050  00000000  00000000  00000000  00000000  | ................ |
   +0060  C1D7D7C3  C1D7F0F5  00000000  00000000  | APPCAP05........ |
   +0070  00000000  00000000  028030C0  02804028  | ...........{.. . |
   +0080  00000000  0292A2FC  00000000  00000000  | .....ks......... |
   +0090  00000000  0290DA98  00000000  00000000  | .......q........ |
   +00A0  00000000  00000000  023FA0A8  00000000  | ..........ffy.... |
   +00B0  36200000  000000A0  02906530  0290D898  | .......ff......Qq |
   +00C0  02906620  029065D0  02906580  00000000  | .......}........ |
   +00D0  00000000  00000000  00000000  00000000  | ................ |
   +00E0  00000000  00000000  00000000  00000000  | ................ |
   +00F0  00000000  00000000  00000000  00000000  | ................ |
   +0100  00000000  00000000                      | ........         |

No session limit negotiations were in progress

                  Current Partner LU(s) for APPCAP05
LME: 0290B148
   LMENETID. NETA      LMENM.... APPCAP06  LMEFSM... C2
```

RDTCHECK

RDTCHECK displays the RDTE name, RDTE address, RDTE entry type, RDTE header type, network address, and the current and desired state of an RDTE. In addition, RDTCHECK displays pertinent flag bits from the following control blocks if available:

- Resource definition table application entry (RAP)
- Resource definition table physical unit entry (RCC)
- Resource definition table cross-domain resource manager (RCDRM)
- Resource definition table cross-domain resource entry (RCDRS)
- Resource definition table allocation entry prefix (RCPRE)
- Resource definition table line entry (RLN)
- Resource definition table logical unit entry (RLU)
- Common physical unit prefix (RPU)
- Resource definition table NCP entry (RRN).

Operands

You must specify one address or one symbol.

Address

Enter 1–8 hexadecimal digits in the form X'x...' for the address of the RDTE to be analyzed. If the address is fewer than 8 digits, it is padded on the left with zeros.

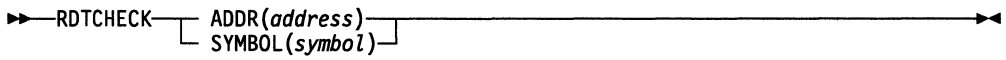
IPCS Symbol

Enter 1–31 alphanumeric characters for an IPCS symbol name that has been previously equated with the address of the RDTE to be analyzed. Do not include a period.

Under IPCS, the symbol X represents the address currently being displayed. If the current address is an RDTE, this symbol may be used to refer to it.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for RDTCHECK

RDTCHECK ADDR(X'02CE0776')

```
                              RDTCHECK Analysis
 RDTE: 02CE077C
    RPRNAME.. APPCAP05  RPRENTRY. 55        RPRBITAN. 01000910  01
    RPRDEVCH. C06D0000  00800000
 DATA: 02CE077C
    +0000  C1D7D7C3  C1D7F0F5  80000000  00550200  | APPCAP05........ |
    +0010  40040000  0001008D  00060000  02CE0878  |  ......-....."..  |
    +0020  02CE07E4  0000015C  0000015C  00000000  | ."..U...*...*....  |
    +0030  02CE0008  00000000  00000000  05050505  | ."..............  |
    +0040  00010009  10010010  00000000  00000000  | ................  |
    +0050  00000001  00000000  C06D0000  00800000  | ........{_......  |
    +0060  00000000  00000000  02CE11A8  00000000  | ........."..y....  |
    +0070  02CE1020  00000000  00000000  00000000  | ."..............  |
    +0080  00000000  00000000  00000000  00000002  | ................  |
    +0090  00000000  00000000  00700033  38E40000  | .............U..  |
    +00A0  00000000  00000000                      | ........          |
RPRENTRY X'55' indicates an application
RPRHDTYP X'02' indicates an application header
Network address         X'00000001008D'
Current state of RDTE   X'0505'
Desired state of RDTE   X'0505'
--------------------------- RPRE STATUS BIT FLAGS ---------------------------
RPRAOPN  = 1  Supports LU to LU sessions
RPRDOM   = 1  LU is in this domain
RPRDINUS = 1  This node has been activated at least once
RPRGIST  = 1  Initial status from system definition
RPRDAFAD = 1  RDT added by config services
--------------------------- RCPRE STATUS BIT FLAGS --------------------------
Non-Backup Session Count  X'00000002'
Session Limit (Zero Means No Limit) X'0000'
Backup Session Count      X'0000'
LOGAPPL
RCPRRECD = 1  Record ok
RCPCROSS = 1  Supports cross domain sessions
RCPPRIM  = 1  LU is primary capable
RCPCYMOD = 00  Operator modifiable feature -None
RCPCYSET = 00  SYSDEF defined feature -None
RCPCSM   = 0001  Unstable
RCPSEC   = 1  LU is secondary capable
RCPLVL   = 1  Level of VTAM >= 4.1 for an LU in an NCP segment
RCPUNRCV = 1  Receipt of unrecognized control vector on CINIT supported
RCPSLUSS = 1  Session started is sent by resource when acting as SLU
RCPT21NS = 1  T2.1 nodes and extended BIND supported
---------------------------- RAP STATUS BIT FLAGS ---------------------------
RAPASLGI = 1  Application first time logon issued
RAPPARS  = 1  Parsess(yes) was coded
RAPAPASS = 1  CLSDST pass authorized
RAPAACQ  = 1  Acquire authorized
RAPAPPC  = 1  APPC=yes was coded
RAPSRBX  = 1  Schedule exits in SRB mode (OS/VS only)
```

RDTFULL

Use RDTFULL to display all resource definition table entries (RDTEs) and node control blocks (NCBs) or a selected RDTE.

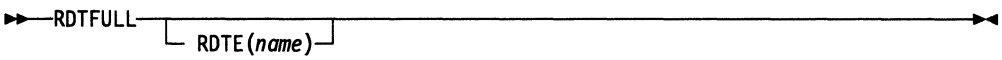
Operands

RDTE Name

The RDTE name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks.

Use the following syntax as an alternative to the panel interface.

Syntax



Note: Using this function without specifying a name may produce large amounts of data.

Sample Output for RDTFULL

RDTFULL

```
                        RDTFULL Analysis
 QAB: 02955740
    +0000  D9C4E340  000D0000  00CC1AB0  02CE0008  | RDT .....ö.[."..  |
    +0010  00100000  00700074                      | ........          |
 RDT: 00CC1AB0
    RPRNAME.. VTAMSEG   RPRENTRY. 02        RPRBITAN. 02000100  00
    RPRDEVCH. 00000000  00000000
 DATA: 00CC1AB0
    +0000  E5E3C1D4  E2C5C740  80000000  00020200  | VTAMSEG ........  |
    +0010  00000000  00000000  00000000  00CC1B18  | ..............ö..  |
    +0020  00000000  000000B0  00000000  00000000  | .......[........  |
    +0030  00000000  00000000  00000000  05050505  | ................  |
    +0040  00020001  00000000  00000000  00000000  | ................  |
:
 RDTE: 00CC1B60
    RPRNAME.. SSCP1A    RPRENTRY. 11        RPRBITAN. 00000940  00
    RPRDEVCH. C06D0000  00800000
 DATA: 00CC1B60
    +0000  E2E2C3D7  F1C14040  80000000  00110200  | SSCP1A  ........  |
    +0010  00000000  00010001  00020006  00CC1C5C  | ..............ö.*  |
    +0020  00CC1BC8  00000210  000000B0  00000000  | .ö.H.......[....  |
    +0030  00CC1AB0  00000000  00000000  05050505  | .ö.[............  |
    +0040  00000009  40000100  00000000  00000000  | .... ...........  |
:
 RDTE: 02CE0CEC
    RPRNAME.. APPCAP09  RPRENTRY. 55        RPRBITAN. 09000810  01
    RPRDEVCH. C06D0000  00800000
 DATA: 02CE0CEC
    +0000  C1D7D7C3  C1D7F0F9  80000000  00550200  | APPCAP09........  |
    +0010  00000000  00010095  000A0000  02CE0DE8  | .......n.....".Y  |
    +0020  02CE0D54  0000015C  0000015C  00000000  | ."....*...*....  |
    +0030  02CE0008  00000000  00000000  02000200  | ."..............  |
    +0040  00090008  10010010  00000000  00000000  | ................  |
:
    +0130  00000000  00000000  00000000  00000000  | ................  |
    +0140  00000000  00020001  00010000  00130000  | ................  |
    +0150  00000000  00000000  00000000            | ............      |
```

RDTHIER

If an RDTE name is specified, the specified RDTE and all RDTEs below it in the RDTE hierarchy are displayed. If no RDTE name is specified, RDTHIER is identical to RDTFULL.

Operands

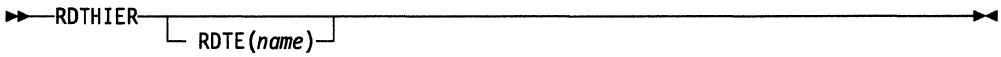
RDTE Name

The RDTE name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks.

If the specified RDTE occurs more than once in a dump, the hierarchies for each RDTE are displayed.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for RDTHIER

```
RDTHIER RDTE(VTAMSEG)

                    RDTHIER Analysis
 RDT: 00CC1AB0
    RPRNAME.. VTAMSEG   RPRENTRY. 02        RPRBITAN. 02000100  00
    RPRDEVCH. 00000000  00000000
 DATA: 00CC1AB0
    +0000  E5E3C1D4  E2C5C740  80000000  00020200  | VTAMSEG ........ |
    +0010  00000000  00000000  00000000  00CC1B18  | .............ö.. |
    +0020  00000000  000000B0  00000000  00000000  | .......[........ |
    +0030  00000000  00000000  00000000  05050505  | ................ |
    +0040  00020001  00000000  00000000  00000000  | ................ |
⋮
 RDTE: 00CC1B60
    RPRNAME.. SSCP1A    RPRENTRY. 11        RPRBITAN. 00000940  00
    RPRDEVCH. C06D0000  00800000
 DATA: 00CC1B60
    +0000  E2E2C3D7  F1C14040  80000000  00110200  | SSCP1A  ........ |
    +0010  00000000  00010001  00020006  00CC1C5C  | .............ö.* |
    +0020  00CC1BC8  00000210  000000B0  00000000  | .ö.H.......[.... |
    +0030  00CC1AB0  00000000  00000000  05050505  | .ö.[............ |
    +0040  00000009  40000100  00000000  00000000  | .... ........... |
⋮
 RDTE: 02CE0CEC
    RPRNAME.. APPCAP09  RPRENTRY. 55        RPRBITAN. 09000810  01
    RPRDEVCH. C06D0000  00800000
 DATA: 02CE0CEC
    +0000  C1D7D7C3  C1D7F0F9  80000000  00550200  | APPCAP09........ |
    +0010  00000000  00010095  000A0000  02CE0DE8  | .......n.....".Y |
    +0020  02CE0D54  0000015C  0000015C  00000000  | ."....*...*.... |
    +0030  02CE0008  00000000  00000000  02000200  | .".............. |
    +0040  00090008  10010010  00000000  00000000  | ................ |
⋮
    +0130  00000000  00000000  00000000  00000000  | ................ |
    +0140  00000000  00020001  00010000  00130000  | ................ |
    +0150  00000000  00000000  00000000            | ............     |
```

RDTSUM

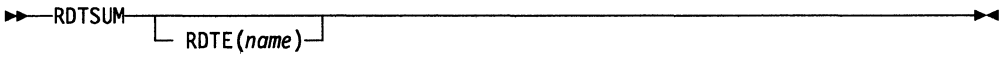
Use RDTSUM to display a summary for all RDTEs or for a selected RDTE.

Operands

RDTE Name

The RDTE name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks.

Syntax



Sample Output for RDTSUM

RDTSUM

```
                        RDTSUM Analysis
 QAB: 02955740
    +0000  D9C4E340  000D0000  00CC1AB0  02CE0008  | RDT .....ö.[."..  |
    +0010  00100000  00700074                      | ........          |
VTAMSEG     ADDRESS 00CC1AB0  RPRHDTYP 02   RPRENTRY 02  APPLICATION HDR
            RPRDAF  000000000000  RPRCURST  0505         ACTIV
SSCP1A      ADDRESS 00CC1B60  RPRHDTYP 02   RPRENTRY 11  CDRM
            RPRDAF  000000010001  RPRCURST  0505         ACTIV
ISTATA00    ADDRESS 00CC1D70  RPRHDTYP 02   RPRENTRY 55  APPL
            RPRDAF  000000010002  RPRCURST  0200         CONCT
ISTNOP      ADDRESS 00CC1ED0  RPRHDTYP 02   RPRENTRY 55  APPL
            RPRDAF  000000010003  RPRCURST  0505         ACTIV
ISTPDCLU    ADDRESS 00CC2030  RPRHDTYP 02   RPRENTRY 55  APPL
            RPRDAF  000000010005  RPRCURST  0505         ACTIV
ISTAPNCP    ADDRESS 00CC2190  RPRHDTYP 06   RPRENTRY 11  CDRM
            RPRDAF  000000010006  RPRCURST  0505         ACTIV
SSCP1A      ADDRESS 00CC23A0  RPRHDTYP 02   RPRENTRY 55  APPL
            RPRDAF  000000010008  RPRCURST  0505         ACTIV
ISTPUS      ADDRESS 00CC2598  RPRHDTYP 01   RPRENTRY 01  PU_T4/5
            RPRDAF  000000010000  RPRCURST  0505         ACTIV
ISTGROUP    ADDRESS 00CC2740  RPRHDTYP 01   RPRENTRY 30  GROUP
            RPRDAF  000000000000  RPRCURST  0505         ACTIV
ISTPDILU    ADDRESS 00CC27C8  RPRHDTYP 07   RPRENTRY 07  CDRSC SEGMENT
            RPRDAF  000000000000  RPRCURST  0505         ACTIV
ISTADJCP    ADDRESS 00CC2870  RPRHDTYP 0F   RPRENTRY 0F  ADJCP MAJ NODE
            RPRDAF  000000000000  RPRCURST  0505         ACTIV
ISTCDRDY    ADDRESS 02CBDF58  RPRHDTYP 07   RPRENTRY 07  CDRSC SEGMENT
            RPRDAF  000000000000  RPRCURST  0505         ACTIV
ISTDSWMN    ADDRESS 02CBEF40  RPRHDTYP 04   RPRENTRY 04  SW SNA MAJ NODE
            RPRDAF  000000000000  RPRCURST  0505         ACTIV
⋮
```

ROUTES

Use ROUTES to display explicit route table entries (ERTEs) and virtual route blocks (VRBLKs).

Operands

None.

Use the following syntax as an alternative to the panel interface.

Syntax

Sample Output for ROUTES

ROUTES

```
                              ROUTES Analysis

 ERTE: 067C7020
    ERTPTR... 00000000  ERTERN... 00        ERTFLG... 00        ERTTGN... 01
    ERTADJSA. 00000001  ERTDSA... 00000001
 DATA: 067C7020
    +0000  14280000  00000000  00C70000  00008000  | .........G...... |
    +0010  00000000  00000001  00000001  00000001  | ................ |
    +0020  00000000  00000000                      | ........         |

 ERTE: 067C70E0
    ERTPTR... 067C70B0  ERTERN... 05        ERTFLG... 00        ERTTGN... 01
    ERTADJSA. 00000004  ERTDSA... 00000002
 DATA: 067C70E0
    +0000  14280000  067C70B0  05830000  00004000  | .....@.[.c.... . |
    +0010  00000000  00000001  00000004  00000002  | ................ |
    +0020  00000000  00000000                      | ........         |

 ERTE: 067C70B0
    ERTPTR... 067C7080  ERTERN... 02        ERTFLG... 00        ERTTGN... 01
    ERTADJSA. 00000004  ERTDSA... 00000002
 DATA: 067C70B0
    +0000  14280000  067C7080  02830000  00008000  | .....@...c...... |
    +0010  00000000  00000001  00000004  00000002  | ................ |
    +0020  00000000  00000000                      | ........         |
⋮
```

```
VRB: 062FE580
   VRBADJSA. 00000001  VRBFXCHN. 00000000  VRBDSTSA. 00000001  VRBVRN... 00
DATA: 062FE580
   +0000  05E80000  00000000  00000001  00000000  | .Y.............. |
   +0010  11280000  00000000  062FEE88  00000000  | ...........h.... |
   +0020  00000000  00000000  00000000  00000000  | ................ |
   +0030  00C1C610  00000000  1B081000  00000018  | .AF............. |
   +0040  00000000  00000000  00000000  00000000  | ................ |
   +0050  00000000  00000000  05000000  062E7100  | ................ |
   +0060  00000000  00000000  00360000  00000000  | ................ |
   +0070  00040000  00000000  00000000  00000000  | ................ |
   +0080  05000100  00000000  00000000  00000000  | ................ |
   +0090  00000000  00000000  00000000  00000000  | ................ |
   +00A0  00000000  00000000  05000200  00000000  | ................ |
   +00B0  00000000  00000000  00000000  00000000  | ................ |
   +00C0  00000000  00000000  00000000  00000000  | ................ |
   +00D0  00000000  00000001  00000000  00000000  | ................ |
   +00E0  00000000  00000001                      | ........         |

VRB: 062EE268
   VRBADJSA. 00000004  VRBFXCHN. 062EE360  VRBDSTSA. 00000002  VRBVRN... 03
DATA: 062EE268
   +0000  05E80300  062EE360  00000004  00000000  | .Y....T-........ |
   +0010  11280000  00000000  062FEE88  00000000  | ...........h.... |
   +0020  00000000  00000000  00000000  00000000  | ................ |
   +0030  00C1C610  00000000  1B081000  00000018  | .AF............. |
   +0040  00000000  00000000  00000000  00000000  | ................ |
   +0050  00000000  00000000  01000000  00000000  | ................ |
   +0060  00000000  00000000  00000000  00000000  | ................ |
   +0070  00000000  00000C18  00000000  00000000  | ................ |
   +0080  01000100  00000000  00000000  00000000  | ................ |
   +0090  00000000  00000000  00000000  00000C18  | ................ |
   +00A0  00000000  00000000  01000200  00000000  | ................ |
   +00B0  00000000  00000000  00000000  00000000  | ................ |
   +00C0  00000000  00000C18  00000000  00000000  | ................ |
   +00D0  03000000  00000002  00000000  00000000  | ................ |
   +00E0  00000000  00000001                      | ........         |
⋮
```

SES

Use SES to format the RDTE specified by *name*, and all SIBs and RDTEs in session with *name*. The specified *name* can be any session endpoint, such as a logical unit, terminal, or application program. It also formats:

- ACDEBs
- APPCBs
- COPRs
- FMCBs
- FMCBEXTs
- HSICBs
- LUCBs
- NSICBs
- NSSCBs
- RABs
- SABs
- SIBIXs
- SIBRXs
- SIBXs.

If *name* is not specified, SES formats all SIBs, RDTEs, ACDEBs, APPCBs, COPRs, FMCBs, FMCBEXTs, HSICBs, LUCBs, MPSTs, NSICBs, NSSCBs, PSTs, RABs, SABs, SIBIXs, SIBRXs, and SIBXs.

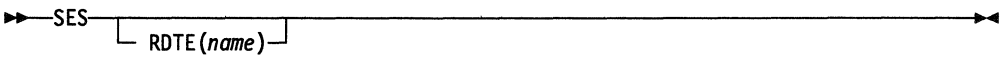
Operands

RDTE Name

The RDTE name should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks. The specified RDTE can be any session endpoint, such as a logical unit, terminal, or application program.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for SES:

```
SES

                          SES Analysis
MPST: 00B14D38
   MPSCHAIN. 00B14DB8  MPSPSTQ.. 02B298E8
DATA: 00B14D38
   +0000  D4D7E2E3  00000000  00B14DB8  00700012  | MPST.....£(½....  |
   +0010  00F74600  028298E8  029137D0  008D85E8  | .7...bqY.j.}.-eY  |
   +0020  7FFD6400  02921830  02829BE8  029137D0  | "....k...bqY.j.}  |
   +0030  029218D0  008D8658  00000000  029137D0  | .k.}.-f......j.}  |
   +0040  000260CA  00B14D38  812633D8  00B14D50  | ..-..£(.a..Q.£(&  |
   +0050  00FECE80  00000000  00000000  00000000  | ..ô.............  |
   +0060  00000000  02825240  00000000  00000000  | .....b. ........  |
PST: 028298E8
   +0000  61000480  00000000  028298E8  00B14D38  | /........bqY.£(.  |
   +0010  00000000  00000000  00000000  00000000  | ................  |
   +0020  00000000  000000EA  00BD7C50  00000000  | .........-@&....  |
   +0030  00000000  00000000  00B1A844  00000000  | .........£y.....  |
   +0040  01011000  00000030  00000000  00000000  | ................  |
   +0050  00000000  00000000  00B1A848  00000000  | .........£y.....  |
⋮
ACDEB: 0280E140
   ACDTCB... 008F1D18  ACDCHN... 02913E60  ACDRDTE.. 02CFFE20
DATA: 0280E140
   +0000  0F100000  02913E60  028298E8  00000000  | .....j.-.bqY....  |
   +0010  00000000  00000000  00B1A590  00000000  | .........£v.....  |
   +0020  2D010000  00000010  00000000  00000000  | ................  |
   +0030  008B4938  02913E60  00000000  0000000B  | .¤...j.-........  |
   +0040  00000000  00000000  00000000  02825240  | ..............b.  |
   +0050  008F1D18  02913E60  00000000  00000000  | .¶...j.-........  |
   +0060  00000000  00000000  0291F3D0  00000000  | .........j3}....  |
   +0070  00000000  00000000  00000006  00000000  | ................  |
⋮
LUCB: 0291F3D0
   +0000  5268004F  00000000  028298E8  00000000  | ...|.....bqY....  |
   +0010  00000000  80000000  0280C018  01FD0006  | ..........{.....  |
   +0020  00000000  02CFFE20  027E9B50  028F7798  | .........=Œ&.¶.q  |
   +0030  0280E140  0280E258  0000000D  027E9B50  | ... ..S......=Œ&  |
   +0040  00000000  00000000  00000000  00000000  | ................  |
   +0050  00B1A588  00000000  19011000  00000048  | .£vh............  |
   +0060  00000000  00000000                      | ........          |

APPCB  0280E258
APPLUCB  0291F3D0    APPTSKID 028298E8    APPACB   008B4938    APPLUN   Z2345678
APPSPTAE 02912980 02912AF0 02912AA0 02912A50 02912A00
000000 62C1D7D7 0291F3D0 028298E8 00000000    00000000 00000000 024110B0 00000000    *.APP..3....Y...................*
000020 31094000 00000010 00000000 00000000    0101001B 00000000 008B4938 02910208    *.. ............................*
000040 00000000 00000000 00000000 00000000    00000000 00000000 00000000 00000000    *................................*
000060 E9F2F3F4 F5F6F7F8 00000000 00000000    00000000 00000000 027E8748 027E1228    *Z2345678.....................*
000080 00000000 02930AFC 00000000 00000000    00000000 02825240 00000000 00000000    *...................... ........*
0000A0 00000000 00000000 023FB0A8 00000000    36200000 000000A0 02912980 02912AF0    *.............................0*
0000C0 02912AA0 02912A50 02912A00 00000000    00000000 00000000 00000000 00000000    *................................*
0000E0 00000000 00000000 00000000 00000000    00000000 00000000 00000000 00000000    *................................*
000100 00000000 00000000                                                             *........                      *

FMCBX: 027E9B50
   +0000  00000007  005A7C13  01000028  028FBAA8  | .....!@.....¶.y  |
   +0010  00000007  005A0000  00070050  027E9820  | .....!.....&.=q.  |
   +0020  00000000  028FB970  028FBBE0  00000000  | .....¶½..¶'\....  |
   +0030  30000000  C1404040  40404040  D5C5E3C2  | ....A       NETB  |
   +0040  40404040  00000000  00000000  00000000  |     ............  |
   +0050  00000000                               | ....              |

FMCB: 028FBAA8
   +0000  032CB000  027E9B50  028298E8  00000000  | .....=Œ&.bqY....  |
   +0010  0281D998  00000000  00000000  00000000  | .aRq............  |
   +0020  02909270  00000000  1C016000  00000018  | ..k.......-.....  |
   +0030  00000000  00000000  00000000  00000000  | ................  |
   +0040  00000000  00000000  02909698  00000000  | ..........oq....  |
   +0050  35086000  00000040  00000000  00000000  | ..-.... ........  |
   +0060  0101001B  00000000  00000000  00000000  | ................  |
   +0070  00000000  00000000  00000000  0100002B  | ................  |
⋮
```

```
HSICB  027E6248
HSISENSE 00000000    HSICONID 01000028    HSIBIUIN 0BA10115    HSISENDQ 00000000
HSIHLDPS 00000000    HSIFMHST 00000000    HSIPACQ  00000000

000000 62C8E2C9 00000000 01000028 00000000    00000000 00000000 0BA10115 98800001    *.HSI.........................*
000020 00000000 00000000 00000000 00000000    00000000 00000000 00000000 80000000    *.............................*
000040 80000000 00018003 00800140 00000000    00000000 00000000 00000000 00000000    *.......... ..................*
000060 0000D000 00000000 00000000 00000000    00000004 00000000 00000000 00000000    *.............................*

SAB    027E8748

SABSHARE 80          SABLRMFL 00          SABFSM   30          SABSENSE 00000000
SABNSFG1 03          SABNSFG2 60

000000 62E2C1C2 00000000 00000000 0100002B    027E1228 D5C5E3C2 40404040 C1404040    *.SAB.............NETB    A  *
000020 40404040 C1F2F3F4 F5F6F7F8 80003000    00000000 00000000 027E8288 00000000    *    A2345678.................*
000040 00000000 00BB19BC 037518D5 00000000    08000360 00000000 00000000 00000000    *..........N.................*
000060 00000000 00000000 027E8288 00000000    00000000 00000000 00000000 00000000    *.............................*
000080 00000000 00000000 00000000                                                    *............                 *

RAB    027E1228
RABCONID 01000028    RABSABPT 027E8748    RABFSM   01          RABCRPLA 00000000    RABPSFSM 020000000D000000

000000 62D9C1C2 00000000 028298E8 00000000    00000000 01000028 0100002B D5C5E3C2    *.RAB.......Y.............NETB*
000020 40404040 C1404040 40404040 C1F2F3F4    F5F6F7F8 A549A1A3 027E8748 00000000    *    A       A2345678.........*
000040 00000000 00000000 024110B4 00000000    32094000 00000040 00000000 00000000    *.................. .... ........*
000060 0101001B 00000000 00000000 01008002    00009000 00000000 00000000 00000000    *.............................*
000080 00000100 00000000 00000000 00000000    00000001 00000000 00000000 00000000    *.............................*
0000A0 00000000 00000000 00000000 00000000    00000000 00000000 00040188 00000000    *.............................*
0000C0 00000000 00000000 00000000 00000000    00000000 00000000 00000000 02000000    *.............................*
0000E0 0D000000 00000000 0280E258 00000000    00000000 00000000                      *..........S.............      *

QAB: 0295A738
   +0000  D9C4E340  000D0000  00B17AB0  02CFF808  | RDT .....£:[..8. |
   +0010  00100000  00700074                      | ........         |

RDT: 00B17AB0
   RPRNAME.. VTAMSEG   RPRENTRY. 02        RPRBITAN. 02000100  00
   RPRDEVCH. 00000000  00000000
DATA: 00B17AB0
   +0000  E5E3C1D4  E2C5C740  80000000  00020200  | VTAMSEG ........ |
   +0010  00000000  00000000  00000000  00B17B18  | .............£#. |
   +0020  00000000  00000B0  00000000  00000000  | .......[........ |
   +0030  00000000  00000000  00000000  05050505  | ................ |
   +0040  00020001  00000000  00000000  00000000  | ................ |
   +0050  00000000  00000000  00000000  00000000  | ................ |
   +0060  00000000  00000000  20000000  00000000  | ................ |
   +0070  00B18598  0295A738  00000000  00000000  | .£eq.nx......... |
   +0080  00000000  00000000  00000000  00000000  | ................ |
   +0090  00000000  00000000  00000000  00000000  | ................ |
   +00A0  00000000  00000000                      | ........         |

RDTE: 00B17B60
   RPRNAME.. SSCP7B    RPRENTRY. 11        RPRBITAN. 00000940  00
   RPRDEVCH. C06D0000  00800000
DATA: 00B17B60
   +0000  E2E2C3D7  F7C24040  80000000  00110200  | SSCP7B  ........ |
   +0010  00000000  00070001  00020009  00B17C5C  | .............£@* |
   +0020  00B17BC8  00000210  000000B0  00000000  | .£#H.......[.... |
   +0030  00B17AB0  00000000  00000000  05050505  | .£:[............ |
   +0040  00000009  40000100  00000000  00000000  | .... ............ |
   +0050  00000000  00000000  C06D0000  00800000  | ........{_...... |
   +0060  00000000  00000000  00000000  00000000  | ................ |
   +0070  00000000  00000000  00000000  00000000  | ................ |
   +0080  00000000  00000000  00000000  00000000  | ................ |
   +0090  00000000  3F013F01  00000000  00080000  | ................ |
⋮
```

```
SIB: 02CE21D0
   SIBFSMIN. FC        SIBFSMTM. 00        SIBFSENS. 00000000
   SIBBPRIQ. 02CE1D38  SIBBSECQ. 02CE1D38  SIBTTMFL. 00        SIBTREAS. 00
   SIBTSESE. 00
   SIBPCID = C2BB19BC037518D6   QUALIFIER = NETB.SSCP7B
DATA: 02CE21D0
   +0000  9800FC00  00000000  C2BB19BC  037518D6  | q.......B‘.°...O |
   +0010  40404040  40404040  5B404040  40404040  |          $       |
   +0020  5B404040  40404040  A549A2FC  3F762104  | $       v.s..... |
   +0030  02CE2048  00000000  02CE1D38  02CE1D38  | .ó.......ó...ó.. |
   +0040  00000000  02CE2290  02CE22F0  02C77054  | .....ó...ó.0.G.. |
   +0050  50311000  00000008  00000000  00000000  | &............... |
   +0060  00000000  00000000  84000000  00000000  | ........d....... |
   +0070  00000000  00000000  00000000  00000000  | ................ |
   +0080  00000000  00000000  00000000  00000000  | ................ |
   +0090  00000000  00000000  00000000  00000000  | ................ |

        SLU (OLU) RESOURCE
SIBRX: 02CE2270
   SIBRADJN. ........  SIBRALNM. ........  SIBRNID.. ........
   SIBRNETA. E3C24040  4040
   SIBRNETC = 00 = SIBRNTU  - Unknown
DATA: 02CE2270
   +0000  00000000  00000000  00000000  00000000  | ................ |
   +0010  00000000  00000000  00000000  00000000  | ................ |
   +0020  00000000  00000000  7C404040  40404040  | ........@        |
   +0030  D5C5E3C2  40404040  00000000  02CFF8B0  | NETB    ......8[ |
   +0040  00000000  001073E0                      | .......\         |

        SLU (DLU) RESOURCE
SIBRX: 02CE22B8
   SIBRADJN. ..a.....  SIBRALNM. ........  SIBRNID.. ........
   SIBRNETA. 00000000  0000
   SIBRNETC =     = Network configuration has an undefined value
DATA: 02CE22B8
   +0000  78008100  00000000  00000000  00000000  | ..a............. |
   +0010  00000000  00000000  00000000  00000000  | ................ |
   +0020  00000000  0007005B  00000000  00000000  | .......$........ |
   +0030  00000000  00000000  00000000  00000000  | ................ |
   +0040  7B404040  40404040                      | #                |
⋮
```

SIBCHECK

SIBCHECK analyzes important fields and relevant status flags in an SIB and related control blocks. The following control blocks are analyzed:

- SIB base
- SIB initiation extension (SIBIX)
- SIB termination extension
- PLU resource extension (SIBRX)
- SLU resource extension (SIBRX)
- OLU cross-network extension (SIBX)
- DLU cross-network extension (SIBX).

SIBCHECK determines:

- Originating logical unit
- Destination logical unit
- Initiating logical unit
- Primary logical unit
- Secondary logical unit
- RDTE address
- Resource type (APPL, LU, CDRSC, and so on)
- Configuration (cross-domain, cross-network, back-to-back, and so on)
- Real resource names and network IDs
- Alias resource names and network IDs
- Network addresses
- GWNCP names.

For each status bit in the SIB and related control blocks, the bit name, its value, and its meaning (from *VTAM Data Areas for MVS/ESA*) are listed.

Fields in the SIB that contain addresses are checked (such as SIBTV35P, the CV35 pointer, or SIBTNOTP, the pointer to NOTIFY RU). If these fields contain a nonzero address, the address and description of the field are also displayed.

Note: The SIBBTIME value is displayed in the format of Coordinated Universal Time (formerly known as Greenwich Mean Time).

Operands

You must specify one address or one symbol.

Address

Enter 1–8 hexadecimal digits in the form X'x...' for the address of the SIB to be analyzed. If the address is fewer than 8 digits, it is padded on the left with zeros.

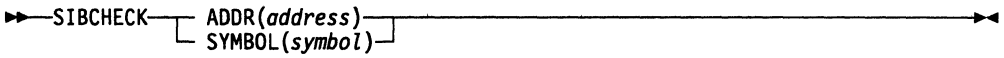
IPCS Symbol

Enter 1–31 alphanumeric characters for an IPCS symbol name that has been previously equated with the address of the SIB to be analyzed. Do not include a period.

Under IPCS, the symbol X represents the address currently being displayed. If the current address is an SIB, this symbol may be used to refer to it.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for SIBCHECK

SIBCHECK ADDR(X'02CCA020')

```
                      SIBCHECK Analysis
                              CDRM: VTAM
                             NetID: NETA
                    Network address: 00000001 0001
                       SIB address: 02CCA020
         OLU|PLU                              DLU|SLU
Name (from RDTE): APPCAP05          Name (from RDTE): APPCAP06
    RDTE address: 02CC7B7C              RDTE address: 02CC7CD8
       RDTE type: APPL                     RDTE type: APPL
     Owning CDRM: VTAM                   Owning CDRM: VTAM
           NetID: NETA                         NetID: NETA
      Alias name: APPCAP05                Alias name: APPCAP06
     Alias netid: NETA                   Alias netid: NETA
   Adjacent SSCP: ........             Adjacent SSCP: ........
 Network address: 00000001 000F      Network address: 00000001 0012
SIBFSMIN: F4 = SIBIFSST - Pending generic session start state
SIBFSMTM: 00 = SIBTFSIS - Initial state

                      Analyze SIB Base
             Original PCID for session: D5376DF4EA88AAB0
          Timestamp from SIB (SIBBTIME): A94FBAF5FD484FF1
        Converted Timestamp (SIBBTIME): 08/13/92 12:36:29.816279
  Primary SIB queue (SIBBPRIQ) elements: 0
Secondary SIB queue (SIBBSECQ) elements: 0
SIBBAUTO = 0 - This is not an AUTOLOGON (LOGAPPL) session
SIBBIOLU = 1 - OLU is the initiating LU
⋮
 Analysis of resource extension for DLU|SLU APPCAP06 at address 02CCA140
SIBRNETC = 10 = SIBRNTS  - Network configuration: Same domain
SIBRCDTC = 0 - Session has not been associated with the CDTAKEDOWN complete
               RU for the CDRM specified in SIBRADJN
⋮
Analysis of SIB Initiation Extension at address 02C6D020
Routing FSM (SIBIRFSM): 20
Failing RU (SIBIFRU): 00000000
Failing reason (SIBIFRSN): 00
Failure status (SIBIFST): 00
SIBICDR  = 0 - CDINIT DQ is not waiting on any I/O
SIBIDLUA = 0 - The DLU network ID is not assumed
⋮
Analysis of SIB Termination Extension
    RID for session takedown (SIBTTRID): 0000000000000000
                 Process RID (SIBTTRID): 0000000000000000
SIBTDSUS = 0 - Session was not suspended for duplicate session
SIBTORIG = 0 - SIBUSER - Termination originator is network user
⋮
Cross-network extension does not exist for OLU|PLU APPCAP05

Cross-network extension does not exist for DLU|SLU APPCAP06
```

SPANC

SPANC analyzes any or all of the VTAM storage pool anchors (SPANCS). If you use no operands, the number of pages in use, the page size, and where the storage is allocated (common or private) for every SPANC pool are displayed. Options are available to:

- Designate a specific SPANC pool to be analyzed
- Determine the number of FBQEs on each page
- Determine the size of each FBQE on each page
- Display a sample of storage from each page
- Determine the page addresses associated with pool(s)
- Process data in a specific pool through the use of an exit.

Operands

Pool

Specify the name of a specific SPANC pool to be analyzed. If the pool operand is not used, all SPANCs are processed. If a pool name other than one from the list of valid pool names is specified, no output will be produced.

Note: Pools named 'AVAIL' are not valid and are used only as place-holders. They are displayed in the event storage overlays occur.

Note: The FBQE Count, FBQE List, Process, and Exit operands are mutually exclusive; use only one of them.

FBQE Count

Specify Y to have the number of FBQEs on each page of the selected pool (or all pools if no pool was selected) listed. The FBQE contains the length of the free storage it describes. Use this option for performance or storage fragmentation problems. Long chains of FBQEs can cause VTAM performance problems.

FBQE List

Specify Y to have each FBQE on each page of the selected pool (or all pools if no pool was selected) listed. The FBQE contains the length of the free storage it describes. Use this option for storage fragmentation problems.

Length

Specify the number of bytes of storage you want displayed from the beginning of each page of the selected pool (or all pools if no pool was selected). Any hexadecimal number from X'001' to X'FF8' or any decimal number from 1 to 4088 may be specified. Use this option to get a sample of storage from each page of a specific SPANC pool.

Process

Specify **Map** to display the address of each page that is associated with the selected pool (or all pools if no pool was selected). Use this option in conjunction with VSMDATA to determine the SPANC pages mapped by each MVS subpool.

Exit

Use Exit to have one of the four exit functions process information on each page of selected SPANC pools. Specify exit FMCB, RU, RUPE, or SIB.

1. The FMCB exit searches SPANC pools FMCB, PLUSFMCB, or SSCPFMCB for FMCBs and formats those found.

2. The RU exit searches SPANC pools UTILCSAS, UTILCSAL, UTILPVTL, or UTILPVTS for all RUs or a specific RU on a page of storage and displays the address and data for those found.

   Note: These pools may contain data which is not an RU. To locate a specific RU, specify the actual RU in the Value field, a Type of X, and a displacement of X'06'.

3. The RUPE exit searches SPANC pools RUPECOMM or RUPEPRIV for all RUPEs in the pool. SPANC displays the RUPE address, CPCBOPC, RUPEOAF, RUPEDAF, and RU data for those found.

4. The SIB exit searches SPANC pool SIB for all SIBs in the pool. SPANC displays the SIB address, FSMs, sensecode, PLU NetID, PLU name, SLU NetID, SLU name, and procedure correlation identifier (PCID) for SIBs that are found.

   Note: For all EXIT routines, an address followed by an asterisk (*) indicates that the buffer pool is allocated.

Note: The following three operands, Displacement, Value, and Value Type, must be used together with the Exit operand. The Exit operand may be used alone.

Displacement
Enter the displacement into the data portion of a page where Value is to be found. The maximum decimal displacement is 4095, and the maximum hexadecimal displacement is X'FFF'.

Value
Enter a character, hex, or binary value to be searched for at the displacement specified by Displacement.

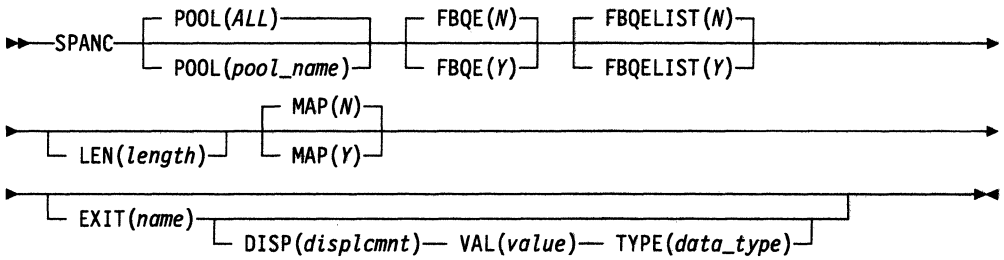
Value may contain character or hexadecimal data of 1–8 bytes in length. Character data should consist of alphanumeric characters. Hexadecimal data should contain an even number of hexadecimal digits in the form X'xx..', otherwise, the high order half-byte is assumed to be zero.

Binary data can be used to look at a particular bit within a byte. You may specify one byte of binary data in the form X'xx'. Only 1 bit within the byte may be selected. Therefore, you can specify only the following hexadecimal values: 01, 02, 04, 08, 10, 20, 40, and 80. A value with more than 1 bit set (for example, 82) will not be accepted.

Value Type
Enter B for binary, C for character, or X for hexadecimal to indicate the type of data entered for Value.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for SPANC

SPANC

```
                          SPANC Analysis
Pages in use for pool RUPEPRIV =        1    Page size = 00010000  PRIVATE
Pages in use for pool RUPECOMM =        1    Page size = 00001000  COMMON
Pages in use for pool SIB      =        1    Page size = 00010000  PRIVATE
Pages in use for pool SSCPFMCB =        1    Page size = 00010000  PRIVATE
Pages in use for pool AVAIL    =        0    Page size = 00001000  PRIVATE
Pages in use for pool EPTDVT   =        2    Page size = 00001000  COMMON
⋮
Pages in use for pool UTILFIXD =        0    Page size = 00001000  COMMON
Pages in use for pool POWMPRIV =        2    Page size = 00001000  PRIVATE
Pages in use for pool POWMCOMM =        0    Page size = 00001000  COMMON
Pages in use for pool POAPRIV  =        0    Page size = 00001000  PRIVATE
Pages in use for pool BFRTRFUL =        0    Page size = 00010000  COMMON
```

SPANC POOL(SIB) EXIT(SIB)

```
                          SPANC Analysis
 SIB ADDR   FSMs Sensecode PLUNETID  PLUNAME  SLUNETID   SLUNAME        PCID
 ---------  ---- --------- --------  -------- --------   ---------  ---------------
 02CCA020*  F400 00000000  NETA      APPCAP05 NETC       APPCAP06  D5376DF4EA88AAB
 Matches found in exit =              1
 Pages in use for pool SIB      =        1    Page size = 00010000  PRIVATE
```

```
SPANC POOL(FMCB) EXIT(FMCB)
                              SPANC Analysis
 02908028*
 TSPLNGTH   2C    TSPEPTA    0291A1F0   TSPDEBA    02923140
 TSPTCFL1   DC    TSPPSFL1   00         TSPSTAT1   8000
  FMCB: 02908028
     +0000  032C8000  02907028  0292A3D8  00000000  | .........ktQ.... |
     +0010  0291A1F0  00000000  00000000  00000000  | .j 0............ |
     +0020  02919270  00000000  1C016000  00000018  | .jk.......-..... |
     +0030  00000000  00000000  00000000  00000000  | ................ |
     +0040  00000000  00000000  02919040  00000000  | .........j. .... |
:
 02908160*
 TSPLNGTH   2C    TSPEPTA    0291A1F0   TSPDEBA    02923370
 TSPTCFL1   DC    TSPPSFL1   00         TSPSTAT1   0000
  FMCB: 02908160
     +0000  032C0000  029070B0  0292AE88  00000000  | .......[.k‡h.... |
     +0010  0291A1F0  00000000  00000000  00000000  | .j 0............ |
     +0020  02919270  00000000  1C016000  00000018  | .jk.......-..... |
     +0030  00000000  00000000  0101001D  0101001D  | ................ |
     +0040  00000000  00000000  02919040  00000000  | .........j. .... |
:
 Matches found in exit =              2
 Pages in use for pool FMCB     =       1    Page size = 00010000  COMMON

SPANC POOL(UTILPVTS) EXIT(RU)
                              SPANC Analysis
 RU ADDR                              RU Data
 ---------  ------------------------------------------------------------------
 02C72020   0E000000000100
 02C64020*  000000030DD5C5E3C14BE3C8C9E2C8D6E2E3000000000000000000000000000000
 02C64048   0000001F0019D4D6C4C9C6E840C9C47EC4C1E5C56BE3C1C2D3C54040404040
 02CBB020   0000003D00378106801040020200000E2D5C1E2E5C3D4C7F308C1D7D7C3C1D7F0
 02C74020   000E1C1C02C6709002C6706802C6706800000000000000000000000000000000
 02C740A8*  0000005F00598106200302D5376DF4EA88AAB0000000000006F308C1D7D7C3C1
 Matches found in exit =              6
 Pages in use for pool UTILPVTS =       4    Page size = 00001000  PRIVATE

SPANC POOL(RUPEPRIV) EXIT(RUPE)
                              SPANC Analysis
 RUPE ADDR  Op code     RUPEOAF      RUPEDAF              RU Data
 ---------  --------  ------------  ------------  ----------------------
 02C53020*  08810620  000000010001  000000010010  REQ=8106200302D5376DF4EA
 02C530C0*  0F310000  00000001000F  000000010012  REQ=31001307B0B050B30080
 02C53160*  0F310000  00000001000F  000000010012  REQ=31001307B0B050B30080
 02C53200*  04000000  000000010003  000000010003  REQ=C4C9E2D7D3C1E840C9C4
 02C532A0   04000000  000000010003  000000010003
 02C53340*  0B310000  00000001000F  000000010012  REQ=FF310281A02801880002
 Matches found in exit =              6
 Pages in use for pool RUPEPRIV =       1    Page size = 00010000  PRIVATE
```

SRTFIND

Use SRTFIND to locate a symbol resolution table entry (SRTE) in a dump.

Note: An attempt is always made to translate a symbol or a string regardless of the quality of the input data stream. The translation may produce unexpected results such as dots, random letters, or other combinations of symbols.

Operands

SRT Name

The SRT name is the symbolic name of a symbol resolution table (SRT) entry and can be entered as alphanumeric characters or hexadecimal digits.

- If alphanumeric characters are used, enter 1–8 characters in the form cccc. If fewer than 8 characters are entered, the name is padded on the right with blanks, and the tool will search only for the characters entered.

  For example, if APPL1 is entered, and APPL1, APPL1A, and APPL1B exist, the tool will find only APPL1.

- If hexadecimal digits are used, enter an even number of digits from 1–16 in the form X'xxxx'. If fewer than 16 digits are entered, the name is padded on the right with blanks. If an odd number of digits is entered, the name is padded on the left with a zero.

  For example, if X'00000010001' is entered, the tool will search for X'0000000100014040'.

The SRT name is required.

NetID

The NetID name representing the network ID of another network outside the host network where the resource resides should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks.

Type

Using the name or hexadecimal value, enter the type of SRTE for which you are searching. The default is RDTE. Enter hexadecimal values in the form of X'xx'.

Chain

Use **Display SRT Chain** to display all SRTEs on the chain, regardless of other search criteria. This option may help locate an SRTE whose storage has been corrupted. The default displays only the SRTEs that match all specified search criteria.

Note: **Chain** overrides the setting for **Process**.

Format

Use **Format** to have selected data formatted using the SRT control block. **Noformat**, the default, displays the SRT's name, address, and type.

Process

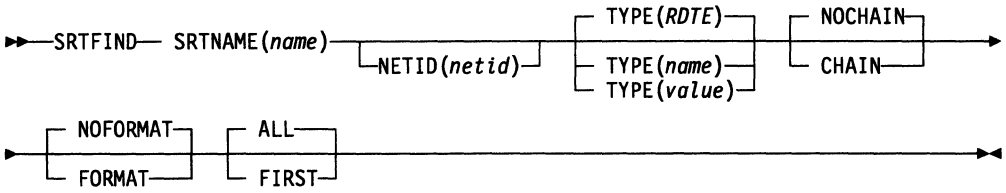
Use **All** or **First** to find the SRTs that match the search criteria. The default is **All**. **First** displays only the first SRTE that matches the search criteria.

Equated Symbols

Symbol	Description
ISTSRTsrtname	SRTE
ISTSRTDsrtname	SRTDATA (if present)

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for SRTFIND

```
SRTFIND SRTNAME(SSCP1A) FORMAT

                          SRTFIND Analysis

SRTE SSCP1A was found at address X'80BF02C8' with type RDTE

 SRT: 80BF02C8
    SRTSYMNM. SSCP1A    SRTSRTE.. 82DA0100  SRTTYPE.. 00
 SRTSRTE: 80BF02D0
    +0000  SPECE.... 82
 DATA: 80BF02C8
    +0000  E2E2C3D7  F1C14040  82DA0100  00550200  | SSCP1A  b.......  |
    +0010  00040000                                | ....              |
```

STORAGE

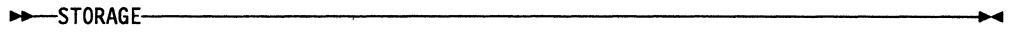
Use STORAGE to format BPCBs, BPDTYs, PXBs, SPANCs, and SPTAEs.

Operands

None.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for STORAGE

STORAGE

```
                        STORAGE Analysis
⋮
 BPD
  DATA: 02952000
     +0000  000C000C  00000000  02952508  00000000  | ........n...... |
     +0010  00000000  00002000  7FFFFFFF  02952390  | ........"....n.. |
     +0020  029521A8  00CC41F8  00000000  000003E8  | .n.y.ö.8.......Y |
     +0030  013400FD  02952054  0281E000  02957000  | .....n...a\..n.. |
     +0040  00000000  00000000  00000000  00000000  | ................ |
⋮
 Buffer pool ID SMS1
  BPCB: 02952390
     BPCBRPHA. 00000000  BPCBRPHB. 00000000  BPCBRPH1. 00000000
     BPCBRPH2. 00000000  BPCBAVNO. 00000000
  DATA: 02952390
     +0000  00000000  00000000  600000E7  02953FF8  | ........-..X.n.8 |
     +0010  00000000  00000000  00000000  00000000  | ................ |
     +0020  00000000  00000000  02952000  00000000  | .........n...... |
     +0030  00000000  00000000  00000000  00000000  | ................ |
     +0040  00000000  00000000  00000000  00000000  | ................ |
⋮
 SPANC  02957204
 POOLNUM 0000       POOLNAME RUPEPRIV  ASSOCID N/A
  DATA: 02957204
     +0000  D9E4D7C5  D7D9C9E5  000C0002  00000000  | RUPEPRIV........ |
     +0010  00000000  00000000                      | ........         |
⋮
  SPTAE: 0295721C
     SPTFLAGS. 10        SPTALLOC. 00000000  SPTFREE.. 02C53008
     SPTSIDEQ. 00000000  SPTUSECT. 00000320  SPTHIUSE. 000003C0
     SPTNBRPG. 00000001  SPTLNGTH. 000000A0
⋮
  DATA: 0295721C
     +0000  02957204  0295725C  00000000  00000000  | .n...n.*........ |
     +0010  00000000  00100000  00000000  02C53008  | .............E.. |
     +0020  00000000  00000320  000003C0  00000001  | ...........{.... |
     +0030  00000000  000000A0  00000199  00000001  | .......ff...r.... |
⋮
  SPTAE: 0295725C
     SPTFLAGS. 00        SPTALLOC. 00000000  SPTFREE.. 00000000
     SPTSIDEQ. 00000000  SPTUSECT. 00000000  SPTHIUSE. 00000000
     SPTNBRPG. 00000000  SPTLNGTH. 00000178
⋮
  DATA: 0295725C
     +0000  02957204  00000000  00000000  00000000  | .n.............. |
     +0010  00000000  00000000  00000000  00000000  | ................ |
     +0020  00000000  00000000  00000000  00000000  | ................ |
     +0030  00000000  00000178  0000000A  00000000  | ................ |
```

VITAL

Use the VITAL function to extract an internal VIT from a dump for use with the VIT analysis tool. See Chapter 10, "Using the VIT Analysis Tool (MVS, VM)" on page 411.

Before using VITAL, allocate a data set for the specified DD name. The data set, when VITAL is invoked, must have these attributes.

RECFM=VB
LRECL=284
DSORG=PS

Notes:

1. The VIT extracted by VITAL can be used as input to the VIT analysis tool, but not to the IPCS GTFTRACE subcommand.
2. The VIT analysis tool and ACF/TAP options that rely on the timestamp might not run as expected because individual entries in the VITAL output have the same timestamp.

Operands

DD Name

Specify the name of the DD statement allocated to receive the extracted VIT. The DD name should be a 1–8 alphanumeric character name. If it contains fewer than 8 characters, it is padded on the right with blanks.

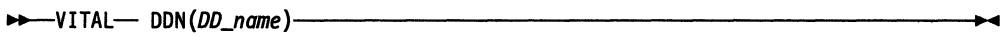
The DD name is required.

You must allocate the specified data set before VITAL is invoked. VITAL will not allocate the data set for you.

Note: The jobname field in the GTF header is set to VFDTRACE. The ASCB address field is set to zero.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for VITAL

VITAL

```
 VITAL DD(VITDATA)

                              VITAL Analysis

VITAL processing completed successfully
```

VTAM

Use VTAM to format and display the following:

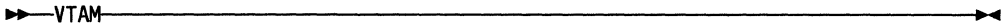
- RDT and RDTEs
- Memory process scheduling table (MPST) and process scheduling table (PST)
- ACDEBs, APPCBs, COPRs, FMCBs, FMCBEXTs, HSICBs, LUCBs, NSICBs, NSSCBs, RABs, and SABs
- NCBs
- Buffer pool control blocks (BPCBs), buffer pool directory (BPDTY), pool extension blocks (PXBs), storage pool anchor block (SPANC), SPANC task-associated element (SPTAE), storage pool page table (PAGTB), and storage pool page table entries (PTEs)
- Locked queue anchor block (LQAB)
- Waiting request elements (WREs) and event identifiers (EIDs)
- Modules from the ATCVT, in the form *module name* and *module address*, sorted by module name.

Operands

None.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for VTAM

VTAM

```
                        VTAM Analysis
BPD
 DATA: 02953000
    +0000  000C000C  00000000  02953508  00000000  | .........n..... |
    +0010  00000000  00002000  7FFFFFFF  02953390  | ........"....n.. |
    +0020  029531A8  00CC41F8  00000000  000003E8  | .n.y.ö.8.......Y |
⋮
Buffer pool ID SMS1
 BPCB: 02953390
    BPCBRPHA. 00000000  BPCBRPHB. 00000000  BPCBRPH1. 00000000
    BPCBRPH2. 00000000  BPCBAVNO. 00000000
 DATA: 02953390
    +0000  00000000  00000000  600000E7  02954FF8  | ........-..X.n|8 |
    +0010  00000000  00000000  00000000  00000000  | ................ |
    +0020  00000000  00000000  02953000  00000000  | .........n...... |
⋮
SPANC  02958204
POOLNUM 0000       POOLNAME RUPEPRIV  ASSOCID N/A
 DATA: 02958204
    +0000  D9E4D7C5  D7D9C9E5  000C0002  00000000  | RUPEPRIV........ |
    +0010  00000000  00000000                      | ........         |
 SPTAE: 0295821C
    SPTFLAGS. 10        SPTALLOC. 00000000  SPTFREE.. 02C4F008
    SPTSIDEQ. 00000000  SPTUSECT. 000000A0  SPTHIUSE. 00000500
    SPTNBRPG. 00000001  SPTLNGTH. 000000A0
 DATA: 0295821C
    +0000  02958204  0295825C  00000000  00000000  | .nb..nb*........ |
    +0010  00000000  00100000  00000000  02C4F008  | .............D0. |
    +0020  00000000  000000A0  00000500  00000001  | .......ff........ |
⋮
 MPST: 00CBED38
    MPSCHAIN. 00CBEDB8  MPSPSTQ.. 02818328
 DATA: 00CBED38
    +0000  D4D7E2E3  80000000  00CBEDB8  00700016  | MPST.......½.... |
    +0010  00F74180  02818328  0290EB18  00AFB040  | .7...ac......„[  |
    +0020  C1E4E3C8  823BA7CA  00FDD6F6  00160002  | AUTHb.x...O6.... |
⋮
 PST: 02818328
    +0000  61000480  00000000  02818328  00CBED38  | /........ac..... |
    +0010  00000000  00000000  00000000  00000000  | ................ |
    +0020  00000000  00000051  00AF8FF0  00000000  | .........„ff0.... |
⋮
 QAB: 02955740
    +0000  D9C4E340  000D0000  00CC1AB0  02CE0008  | RDT .....ö.[."..  |
    +0010  00100000  00700074                      | ........         |
 RDT: 00CC1AB0
    RPRNAME.. VTAMSEG   RPRENTRY. 02        RPRBITAN. 02000100  00
    RPRDEVCH. 00000000  00000000
 DATA: 00CC1AB0
    +0000  E5E3C1D4  E2C5C740  80000000  00020200  | VTAMSEG ........ |
    +0010  00000000  00000000  00000000  00CC1B18  | .............ö.. |
    +0020  00000000  000000B0  00000000  00000000  | .......[........ |
⋮
 RDTE: 00CC1B60
    RPRNAME.. SSCP1A    RPRENTRY. 11        RPRBITAN. 00000940  00
    RPRDEVCH. C06D0000  00800000
 DATA: 00CC1B60
⋮
 ATCVT: 00CC41F8
    ISTACC00. 82A957A0  ISTACC01. 82ACD8B8  ISTAICIR. 80DD9000
    ISTAICPT. 823B7014  ISTAPCAD. 823B8298  ISTAPCES. 823BC560
    ISTAPCGT. 823C5078  ISTAPCIE. 822E29F0  ISTAPCIN. 823BD990
    ISTAPCKU. 80DDA5A0  ISTAPCPC. 823BCE58  ISTAPCPD. 823B8920
    ISTAPCPS. 823C0580  ISTAPCRP. 822E2934  ISTAPCRS. 823B8F84
⋮
```

```
ATCIOLQB
 LQAB: 02C6DF30
    LQABFRST. 00000000  LQABLAST. 00000000  LQGSUBA.. 00000001
 DATA: 02C6DF30
    +0000  00000000  00000000  02D3D8C7  00000001  | .........LQG.... |
    +0010  00000000  00000000                      | ........        |
ATCLUSMQ
 LQAB: 02A3BC14
    LQABFRST. 00000000  LQABLAST. 00000000
 DATA: 02A3BC14
    +0000  00000000  00000000  03D3D8C7            | .........LQG    |
ATCMCQAB
 LQAB: 02A3BC08
    LQABFRST. 00000000  LQABLAST. 00000000
 DATA: 02A3BC08
    +0000  00000000  00000000  01D3D8C7            | .........LQG    |
ATCPULQB
 LQAB: 02A3BCB4
    LQABFRST. 00000000  LQABLAST. 00000000
 DATA: 02A3BCB4
    +0000  00000000  00000000  04D3D8C7            | .........LQG    |
ATCNOSQ
 LQAB: 02A3BCC0
    LQABFRST. 00000000  LQABLAST. 00000000
 DATA: 02A3BCC0
    +0000  00000000  00000000  05D3D8C7            | .........LQG    |
ATCSSLQB
 LQAB: 02A3BCCC
    LQABFRST. 00000000  LQABLAST. 00000000
 DATA: 02A3BCCC
    +0000  00000000  00000000  06D3D8C7            | .........LQG    |
ATCSSMQB
 LQAB: 02A3BD6C
    LQABFRST. 00000000  LQABLAST. 00000000
 DATA: 02A3BD6C
    +0000  00000000  00000000  07D3D8C7            | .........LQG    |
```

VTBASIC

Use the VTBASIC function to display the ATCVT, the configuration table (CONFT), the component recovery areas (CRAs), and the VTAM internal trace (VIT).

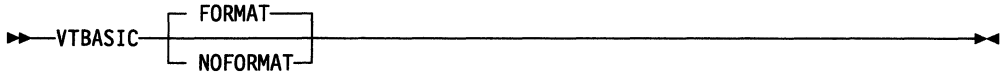
Operands

Trace output

Enter **Format** to format the VIT and **No format** to display the VIT in hexadecimal format. **Format** is the default.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for VTBASIC

VTBASIC NOFORMAT

```
                          VTBASIC Analysis

 VTAM INTERNAL TRACE TABLE    05F6C000

 PRESENT WRAP  A6567A5FFAEE3604  LAST WRAP    A6567A57BB883903
 CURRENT ENTRY 05F6C420          LAST ENTRY   05F9DFE0
    C4E2D740 12582410 02915E88 00CC4908   02A275F8 02A275F8 E3E2E6E4 02929010
    D3D2E2C8 12000100 00CC4C70 00000000   82A95442 00000000 00000000 02929010
    E4D5D3D2 12000100 00CC4C70 00000100   82A9546C 00000000 01000000 02929010
    D8E4C558 12482810 02915E88 00CC4248   82A954F8 02A275F8 C9D5E3D4 02929010
    C5E7C9E3 12000010 02915E88 00CC4908   82A95668 80000000 E3E2E6E4 02929010
    D9C5D8E2 12170000 02915E88 02928810   822E2E16 00010000 0290DDA0 00000000
    C4E2D740 12582810 02915E88 00CC4248   02A275F8 02A275F8 C9D5E3D4 02928810
    C5E7C9E3 12000010 02915E88 00CC4248   82A22B1C 80000000 C9D5E3D4 029 28810
    D9C5D3E2 12170000 02915E88 02928810   822E2CB6 00000000 0290DDA0 00000000

⋮
 ATCVT: 00CC41F8
    ATCRDT... 02955740  ATCSRT... 02C35008  ATCCONFT. 00CC18E8
    ATCBPDA.. 02953000  ATCACTRM. 0000      ATCVTLOD. 02A27650
    GWSSCP =  YES
 DATA: 00CC41F8
    +0000  E5C5F4F2  40404040  FFF900C8  02825000  | VE42    .9.H.b&. |
    +0010  00000000  0000FFF9  11280000  00000000  | .......9........ |
    +0020  02915E88  00000000  00000000  00000000  | .j;h............ |
    +0030  00CC4524  00000000  13201000  00000010  | .ö.............. |
    +0040  11280000  00000000  02915E88  00000000  | .........j;h.... |
⋮
    +07B0  00000000  00000000  00000000  00000000  | ................ |
    +07C0  00000000  00000000  00000000  00000000  | ................ |
    +07D0  00000000  00000000  00000000  00000000  | ................ |
    +07E0  00000000  00000000  00000000  00000000  | ................ |
```

VTBUF

Use VTBUF to analyze buffer pool control blocks (BPCBs) and obtain a status summary for all buffer pools or a specific buffer pool. For each buffer pool, the following information is displayed:

- Starting and ending address of buffer pools
- Buffer pool address (BPCB)
- Buffer type (fixed or pageable)
- Buffer size
- Number of buffers allocated
- Slowdown threshold
- Number of buffers available
- Expansion threshold
- Contraction threshold
- Number of times expanded
- Maximum number of buffers
- Expansion increment
- Expansion size
- Percentage of buffers in use
- Total number of buffers
- Bytes in static and expanded areas
- Buffers in other pools
- Buffers in static area
- Total queued request parameter headers (RPHs).

If any expansions have occurred, the pool extension block (PXB) address, number of buffers available, totals buffers, beginning of the extent, and the first available extent are also presented.

Operands

Buffer Name

Enter a 2–7 character buffer name in the form *cc* or *cccc*BUF where *cccc* is the buffer name. Counts and totals information will be displayed in decimal form.

The default is ALL.

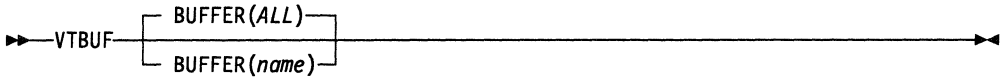
Equated Symbols

Symbol	Description
buffername BSTART	Each starting buffer address
buffername BEND	Each ending buffer address

Note: For information on the DISPLAY BFRUSE (buffer use) command, which displays information about VTAM buffer use, see "DISPLAY BFRUSE Command" in *VTAM Operation*.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for VTBUF

VTBUF BUFFER(IO)

```
                          VTBUF Analysis

                      IO Buffer Analysis

Size of Buffers(bytes)          345   Buffer maximum              110
Total buffers available          94   Static buffers allocated    110
Total number of buffers         110
Buffers in use (%)               15   Available static buffers     94
Bytes in static &
  expanded areas              37950
Slowdown threshold                5   Expansion threshold          6
Contraction threshold         32767
Number of expansions              0   Expansion increment         11
Expansion size                 4096
Total queued RPHs                 0
Fixed or pageable?         FIXED
Buffer pool address            X'028DB410'
Beginning address of pool      X'028F9000'
Ending address of pool         X'02903000'
Buffer pool has no extensions
```

VTCVTPAB

Use VTCVTPAB to obtain a list of the PABs and DYPABs in the ATCVT. The PAB control block for each PAB is formatted. For the very extended PABs, the address of the first element on the PAB is displayed. The following PABs and DYPABs are processed:

- ATCITPAB - Internal trace DYPAB
- ATCTMRPB - Timer services DYPAB
- ATCCSPAB - Configuration services DYPAB
- ATCVDPAB - SSCP VARY definition PAB
- ATCPXPAB - Dynamic buffer pool expansion PAB
- ATCPUPAB - PU services PAB
- ATCPUIOP - SSCP/PU services I/O PAB
- ATCLUSRT - LU services router PAB
- ATCNSPAB - TSC no session PAB
- ATCNOSPB - Network operator services PAB
- ATCSSPAB - Session serialization PAB
- ATCSOPAB - Session outage notify PAB
- ATCRYPAB - Definition for CRYPTO task
- ATCWUPAB - Wake up PAB
- ATCTPMPB - TPMSG PAB
- ATCTRMPB - Termination task
- ATCPDPAB - Problem determination trace PAB.

Operands

None.

Use the following syntax as an alternative to the panel interface.

Syntax

Sample Output for VTCVTPAB

VTCVTPAB

```
                          VTCVTPAB Analysis

Configuration Services elements
 PAB: 00C17278
    +0000  86385C68  80000000  065E4100  00000000  | f.*.....;..... |
    +0010  10321000  00000010  00000000  00000000  | ............... |
    +0020  00000000  00000000  00000000  00000000  | ............... |
         1 level elements          0
         2 level elements          0
         3 level elements          0
         4 level elements          0
         5 level elements          4     07524DE0
         6 level elements         55     073BAA60
         7 level elements         45     0751FAC0


Definition for Crypto Task elements              0
 PAB: 00C17CC8
    +0000  00000000  00000000  00C17CA4  00000000  | .........A@u.... |
    +0010  60001000  00000010                      | -.......        |

Dynamic Buffer Pool Expansion elements           0
 PAB: 00C172E8
    +0000  00000000  00000000  00C17D84  06312010  | .........A'd.... |
    +0010  0E0C1000  00000010                      | ........        |

VTAM Termination Task elements                   0
 PAB: 00C17D30
    +0000  00000000  00000000  00C17DAC  00000000  | .........A'..... |
    +0010  0D201000  00000010                      | ........        |

Internal Trace elements                          0
 PAB: 00C17220
    +0000  00000000  00000000  00C17524  00000000  | .........A...... |
    +0010  13201000  00000010                      | ........        |

LU Services Router elements
 PAB: 00C17380
    +0000  86385D40  80000000  00C174DC  06312810  | f.) .....A...... |
    +0010  0F321000  00000010  00000000  00000000  | ............... |
    +0020  00000000  00000000  00000000  00000000  | ............... |
         1 level elements          0
         2 level elements          0
         3 level elements          0
         4 level elements          1     06204708
         5 level elements          9     06204CA8
         6 level elements          8     073BA920
         7 level elements          0
```

```
Network Operator Services elements                0
 PAB: 00C17408
    +0000  00000000  00000000  00C174B4  00000000  | .........A...... |
    +0010  14201000  00000010                      | ........        |

PD Trace elements                                 0

 PAB: 00C17B18
    +0000  00000000  00000000  00C17B04  00000000  | .........A#..... |
    +0010  12201000  00000010                      | ........        |

PU Services elements
 PAB: 00C17310
    +0000  06385E18  00000000  06572F38  00000000  | ..;.............. |
    +0010  06321000  00000010  00000000  00000000  | ................ |
    +0020  00000000  00000000  00000000  00000000  | ................ |
         1 level elements          0
         2 level elements          0

Session Outage Notify elements                    0
 PAB: 00C17458
    +0000  00000000  00000000  00C17650  00000000  | .........A.&.... |
    +0010  28201000  00000158                      | ........        |

Session Serialization elements                    0
 PAB: 00C17440
    +0000  00000000  00000000  00C1764C  00000000  | .........A.<.... |
    +0010  27001000  00000140                      | .......         |

VARY Definition elements                          0
 PAB: 00C172B8
    +0000  00000000  00000000  065A18D4  00000000  | .........!.M.... |
    +0010  0B211000  00000010  00000000  00000000  | ................ |


SSCP/PU Services I/O elements                     0
 PAB: 00C17350
    +0000  80000000  80000000  00C17750  06316010  | .........A.&..-. |
    +0010  11211000  00000010  00000000  00000000  | ................ |

Timer Services elements                           0
 PAB: 00C17248
    +0000  00000000  00000000  00C174CC  00000000  | .........A...... |
    +0010  0C281000  00000010                      | ........        |

TPMSG elements                                    0
 PAB: 00C179C8
    +0000  80000000  80000000  00C175A8  06383810  | .........A.y.... |
    +0010  17251000  00000010  00000000  00000000  | ................ |

TSC No Session elements                           0
 PAB: 00C173C0
    +0000  00000000  00000000  00C17548  00000000  | .........A...... |
    +0010  26001000  00000010                      | ........        |

Wakeup PAB elements                               0
 PAB: 00C17908
    +0000  00000000  00000000  00C1792C  06313010  | .........A ..... |
    +0010  29241000  00000010                      | ........        |
```

VTFNDMOD

Use VTFNDMOD to determine the VTAM module name and displacement for a given address.

VTFNDMOD is particularly useful for converting the issuer address (ISSR) in a VIT into a module name and displacement. It searches up to 5000 bytes before the specified address.

In addition to the module name and displacement into the module that the specified address represents, the following is displayed:

- Date module compiled
- PTF level, if any
- Address entered
- Module entry point address
- Address displacement into module
- First hexadecimal 40 bytes of the module
- Hexadecimal 40 bytes around the entered address
- If the address is in FLPA, MLPA, PLPA, extended PLPA, extended FLPA, or extended MLPA, the beginning and ending address of that region is displayed.

Operands

You must specify one address or one symbol.

Address

Enter 1–8 hexadecimal digits in the form X'x...' for the address used to determine the VTAM module name and displacement. If the address is fewer than 8 digits, it is padded on the left with zeros.

IPCS Symbol

Enter 1–31 alphanumeric characters for an IPCS symbol name that has been previously equated to a location within a VTAM module. Do not include a period.

Under IPCS, the symbol X represents the address currently being displayed. If the current address points to a location within a VTAM module, X may be used to refer to it.

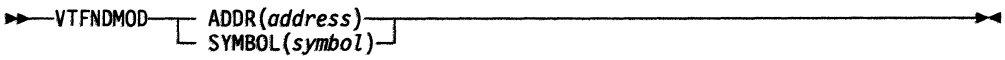
Equated Symbols

After the module name is determined, an IPCS symbol (the module name) is equated to the beginning of the CSECT.

Symbol	Description
module eye-catcher	Module entry point

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for VTFNDMOD

```
VTFNDMOD ADDR(X'2B023C0')

                              VTFNDMOD Analysis

Module name:                 ISTORCEI
Compile date:                92.262

Address entered:             02B023C0
Module entry point:          02B022C0
                             --------
Displacement into module:         100

First '40'X bytes of module:

 DATA: 02B022C0
    +0000  47F0F014  0FC9E2E3  D6D9C3C5  C940F9F2  | .00..ISTORCEI 92 |
    +0010  4BF2F6F2  90ECD00C  18CF41B0  CFFF41A0  | .262..}......... |
    +0020  BFFF4190  AFFF4180  9FFF4170  8FFF50D0  | ..............&} |
    +0030  C5C84160  C5C45060  D00818D6  1F005000  | EH.-ED&-}..O..&. |

Storage around address entered:

 DATA: 02B023AC
    +0000  C54A5860  04085800  65701860  D203602C  | E¢.-........-K.-. |
    +0010  C61CD203  6028C618  5800C630  18204100  | F.K.-.F...F..... |
    +0020  00585830  20005030  60001E06  8B300002  | ......&.-....... |
    +0030  1E035000  C7085820  C7145020  60485020  | ..&.G...G.&.-.&. |
```

VTMODS

Use VTMODS to find the entry point of the VTAM modules that reside in the VTAM private region. VTMODS reports the number of modules found and equates the entry point of each module found to its module name in the IPCS symbol table. After VTMODS executes, the VTAM modules found can be located using the module name in the IPCS LIST command.

VTMODS is particularly useful when you are checking the PTF level of several modules or when you want to quickly verify the PTF or APAR level of a module in a dump.

Note: For VTMODS to execute successfully, the VTAM private region must have been dumped (that is, the RGN parameter must have been specified when the dump was taken). If fewer than 10 modules are found, VTAM private storage is missing from the dump.

If the dump does contain the VTAM private region but is a partial dump, VTMODS attempts to find as many VTAM modules as possible.

Operands

List

The default is N. The default is N. Specify Y to receive a list of each module found, its entry point address, compile date, and PTF level, if present. The modules are in the order that they were found in storage (that is, by storage address, lowest to highest), followed by a list of the modules in alphanumeric order.

Equated Symbols

For each VTAM module that is found, an IPCS symbol (the name of the module as it appears in the module eye-catcher) is equated to the entry point of the module.

Symbol	Description
module eye-catcher	Module entry point

Additional Information

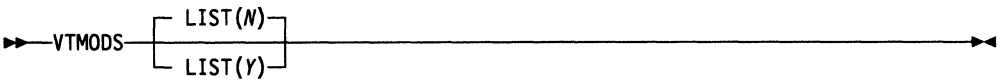
The symbols created remain in the IPCS dump directory until the dump directory is deleted or until an IPCS DROPDUMP command is issued for the dump.

VTMODS scans VTAM private storage for the character string IST. When IST is found, a check is made to determine if this occurrence of IST represents a VTAM module eye-catcher. Most VTAM modules have a branch instruction hexadecimal 47F0F0xx at the entry point to the module to branch around the eye-catcher. If hexadecimal 47F0F0xx appears 5, 7, or 9 bytes before the module eye-catcher, the location is considered in most cases to be a VTAM module entry point.

Storage is scanned starting at the lowest address of private storage to the top of the private region (below the 16MB line). The scan then continues starting at the lowest address of extended private storage (above the 16MB line) and continues for about hexadecimal 400000 bytes.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for VTMODS

VTMODS LIST(Y)

```
                                VTMODS Analysis
Address      Module     Compiled      PTF
--------     --------   --------    -------
000063CA     ISTATM00   91.322
00006A70     ISTINCBX   91.322
00006CD0     ISTINCRS   91.320
000072E0     ISTIECHS   91.320
0000772A     ISTCPM01   91.319
00007768     ISTSSCXY   91.329
00009DC8     ISTCPCIT   91.322
0000A488     ISTINCR4   91.322
0000B7E8     ISTPUCWI   91.320
0000BD10     ISTSSCXX   91.320
0000C0F0     ISTCSCEX   91.322
0000CDA0     ISTPDCLU   91.322
0000EAA8     ISTPDCSE   91.320
00011588     ISTPUCX0   91.320
000124B0     ISTCSCSD   91.319
00015C00     ISTENQIO   91.336
00015EC0     ISTENQPR   91.326
00016A18     ISTENQIN   91.320
00016D90     ISTENQRT   91.320
02A00138     ISTINM01   91.338
02A01C68     ISTLUCQD   91.320
02A01EB0     ISTCPCQD   91.353
02A021B0     ISTPUCQD   91.319
02A021F8     ISTINCXR   92.003
02A02788     ISTSCCIT   92.002
02A02BC8     ISTINCCT   91.352
02A02EA8     ISTINCIT   91.352
02A03100     ISTINCPD   91.352
02A03E78     ISTINFIC   91.352
02A04680     ISTCICPR   91.352
02A082A0     ISTCICDF   91.352
02A08968     ISTINCCP   91.350
02A094E0     ISTORCEI   91.346
02A0EE88     ISTCICTR   91.346
02A10DA0     ISTINCTR   91.346
02A11EC0     ISTINCCF   91.346
02A132F0     ISTDRCIT   91.344
02A13B50     ISTINCSA   91.344
:
02C31168     ISTXP1WB   91.351
02C31848     ISTXP1WC   91.351
02C31D48     ISTXP1WR   91.351
02C32748     ISTXP1WS   91.351
```

VTNODE

Use VTNODE to determine:

- If a SIB exists on the secondary chain where the RDTE is the SLU
- If an RDTE application exists, and if the ACDEB, LUCB, FMCB, and FMCB extension associated with the session exist.

If any SIBs exist on the secondary chain, only the first SIB is processed. To process the PLU, use the SES function. Excerpts of the SIB, LU RDTE, APPL RDTE, ACDEB, LUCB, FMCB, and FMCB extension are displayed if present.

Operands

RDTE Name

The RDTE name of a CDRSC or LU RDTE should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks. The entered RDTE name must be the SLU, and the PLU must be an application. For CDRSC independent LUs, VTNODE will only process the first entry.

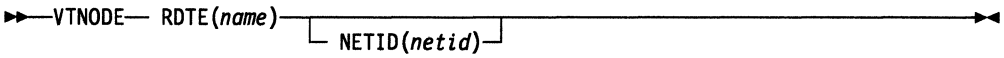
The RDTE name is required.

NetID

The NetID name representing the network ID of another network outside the host network where a resource resides, should be 1–8 alphanumeric characters. If it contains fewer than 8 characters, it is padded on the right with blanks.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for VTNODE

VTNODE RDTE(ECHOB1B) NETID(NETB)

```
                              VTNODE Analysis

RDTE: 06A93DA0
   RPRNAME.. ECHOB1B   RPRENTRY. 83        RPRBITAN. 01000510  00
   RPRDEVCH. C06D0000  00800000
DATA: 06A93DA0
   +0000  C5C3C8D6  C2F1C240  869CF568  0083070A  | ECHOB1B fæ5..c.. |
   +0010  00000000  00000000  00000000  06A93E50  | .............z.& |
   +0020  06A93E08  000A9778  0000AF80  00000000  | .z....p......... |
     .
     .

SIB: 06C93320
   SIBFSMIN. FC        SIBFSMTM. 00        SIBFSENS. 00000000
   SIBBPRIQ. 06C8F9D8  SIBBSECQ. 06C8F9D8  SIBTTMFL. 00        SIBTREAS. 00
   SIBTSESE. 00
DATA: 06C93320
   +0000  9800FC00  00000000  E7F38C74  32DF4AEA  | q.......X3%...¢. |
   +0010  40404040  40404040  C4E2C9D3  C7D4D6C4  |         DSILGMOD |
   +0020  C4E2C9D3  C7D4D6C4  A6567A2C  BAE1D903  | DSILGMODw.:...R. |
     .
     .

RDTE: 069546D0
   RPRNAME.. ECHOC1C   RPRENTRY. 55        RPRBITAN. 01000910  01
   RPRDEVCH. C06D0000  00800000
DATA: 069546D0
   +0000  C5C3C8D6  C3F1C340  867A7320  00550202  | ECHOC1C f:...... |
   +0010  40040000  0001007A  00000000  06954780  |  ......:.....n.. |
   +0020  06954738  00000110  00000110  00000000  | .n.............. |
     .
     .

ACDEB: 060BD7D0
   ACDTCB... 007F0D18  ACDCHN... 00000000  ACDRDTE.. 069546D0
DATA: 060BD7D0
   +0000  0F100000  00000000  062C1DA0  00000000  | ...........ff.... |
   +0010  00000000  00000000  00C17590  00000000  | .........A...... |
   +0020  2D010800  00000010  00000000  00000000  | ................ |
     .
     .


LUCB: 06306490
   +0000  52700079  00000000  062C1DA0  00000000  | ... .......ff.... |
   +0010  00000000  80000000  0629D018  01FD0000  | .........}..... |
   +0020  00060000  00000000  00000000  069546D0  | ............n.} |
     .
     .

TSCB: 86355948
   +0000  99540000  8636FB20  00000000  0635599C  | r...f.........æ |
   +0010  00000000  00000003  00000200  A6567A5C  | ............w... |
   +0020  81000000  00000001  00000000  00000000  | a............... |
     .
     .

No FMCB address was found for the RDTE
```

VTREADYQ

Use VTREADYQ to analyze some of the major control blocks associated with an application. For each memory process schedule table (MPST) chain, the VTAM data extent control blocks (ACDEBs), process scheduling tables (PSTs), logical unit control blocks (LUCBs), and function management control blocks (FMCBs) are checked for PABs, DYPABs, and ready queues that contain queued elements. VTREADYQ lists the first elements on those queues and PABs.

Note: For a large network, this could take several minutes to run.

Operands

None.

Use the following syntax as an alternative to the panel interface.

Syntax

Sample Output for VTREADYQ

VTREADYQ

```
                          VTREADYQ Analysis
MPST     1
Processing begins on FMCB extension for LU TSO0001
MPST     2
Processing begins on FMCB extension for LU TSO
MPST     3
Processing begins on FMCB extension for LU APPCAP06
Processing begins on FMCB extension for LU APPCAP05
MPST     4
Processing begins on FMCB extension for LU ISTATA00
There are no FMCB extensions off of LUCB 02924090
Processing begins on FMCB extension for LU ISTPDCLU
Synchronous TPPOSTed RPH count          1
 ELEMENT: 8295B500
    +0000  01C40020  80000000  02915E88  00CC45AC  | .D.......j;h.ö.œ |
    +0010  82A332A4  6C000010  00CC49B8  00000000  | bt.u%....ö.½.... |
    +0020  00000000  0295B500  00000000  0295B500  | .....n§......n§. |
    +0030  80000000  00000000  00CC41F8  00000041  | .........ö.8.... |
    +0040  00000000  00000300  00000000  80000000  | ................ |
    +0050  00000000  00000000  82A32800  02A334B0  | ........bt...t.[ |
    +0060  00CC41F8  00000000  00000000  00000000  | .ö.8............ |
    +0070  00000000  00000000  00000000  00000000  | ................ |
    +0080  00000000  00000000  00000000  00000000  | ................ |
    +0090  00000000  00000000  00000000  00000000  | ................ |
    +00A0  00000000  00000000  00000000  00000000  | ................ |
    +00B0  00000000  00000000  00000000  00000000  | ................ |
    +00C0  00000000                                 | ....             |
Synchronous normal PAB                  1
 ELEMENT: 00CC4248
    +0000  80000000  80000000  00CC44CC  00000000  | .........ö.ö.... |
    +0010  0CA81000  00000010                       | .y......         |
Processing begins on FMCB extension for LU SSCP1A
Processing begins on FMCB extension for LU VTAM
```

VTRPH

Use VTRPH to analyze the entire LP buffer pool of request parameter headers (RPHs) and display those that are waiting, running, holding locks, or are in error.

If an RPH is waiting at an address other than X'0' or X'FFFFFFFF', the major control block and the current process anchor block (PAB) are listed. In addition, the resume addresses are shown with the number of RPHs that were waiting at those addresses.

Operands

None.

Use the following syntax as an alternative to the panel interface.

Syntax

Sample Output for VTRPH

VTRPH

```
                          VTRPH Analysis
                      LP Buffer Analysis
Buffers available        56
Total number of buffers  64
Number of expansions         0
Buffer found does not contain an RPH at address X'02928010'
RPH at buffer address X'02929010' is running, RPHRESUM = 0
Module was not found
RPH Major control block:
 DATA: 00CC48F8
    +0000  11280000  00000000  02915E88  00000000  | .........j;h.... |
    +0010  00000000  00000000  00CC492C  02929010  | .........ö...k.. |
    +0020  29241000  00000010                      | ........         |
Work elements found    4
RPH work element address X'02A275F8'
RPH at buffer address X'0292B010' is running, RPHRESUM = 0
Module was not found
RPH Major control block:
 DATA: 00CC43F8
    +0000  11280000  00000000  02915E88  00000000  | .........j;h.... |
    +0010  80000000  80000000  00CC44B4  0292B010  | .........ö.fl.k[. |
    +0020  14201000  00000010                      | ........         |
Work elements found    1
RPH work element address X'02C4F340'
Error buffers found                    1
Unallocated buffers found             61
Total number of buffers processed     64
No allocated CRAs were found
```

VTVIT

Use VTVIT to determine which VIT options were in effect at the time of a dump, and whether the trace was running internally (MODE=INT), externally (MODE=EXT), or internally and externally.

If the VIT was running externally, no further processing occurs. IPCS symbols are created for the beginning and end of the internal VIT table, and for the current, oldest, and last VIT entries.

An option is available to produce an unformatted listing of the entire VIT table. Use VTBASIC to format the VIT table.

To extract a VIT from a dump for use with the VIT analysis tool, use VITAL.

Operands

Search Argument

Scan displays the VIT entries containing a specified search argument. Enter 1–8 alphanumeric characters or 1–16 hexadecimal digits in the form X'x...' for the search argument.

If the hexadecimal data string is not an even number of digits, the high order half-byte is set to zero.

List VIT

The default is N. Specify Y to list the entire VIT table. The internal VIT table is processed in the following order:

1. From the oldest trace entry to the end of the trace table
2. From the beginning of the trace table to the current entry.

This results in "unwrapping" the trace table so the trace entries are processed and listed in chronological order (that is, the oldest trace entry is listed first at the top of the output, and the newest trace entry is listed last at the bottom of the output). A message is included in the output to indicate where the physical end of the trace table was encountered.

Equated Symbols

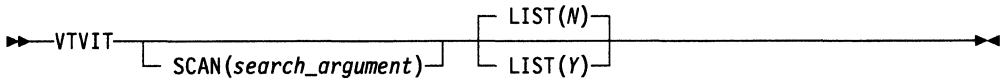
Symbol	Description
VIT	The beginning of the VIT table
VITC	The current VIT entry
VITO	The oldest VIT entry
VITE	The end of the VIT table.

Additional Information

The beginning of the trace table is mapped by ITTRC in *VTAM Data Areas for MVS/ESA*. It contains the present-wrap time stamp, last-wrap time stamp, current-entry address, and last-entry address. The current entry is the most recent entry (that is, the last entry to be written before the dump was taken). The last entry is the one that was written in the last position of the in-storage trace table before wrapping to the beginning of the trace table.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for VTVIT

VTVIT

```
                              VTVIT Analysis

 VTAM external trace options active at the time of this dump

 API LOCK PSS SMS PIU MSG SSCP CIO NRM APPC VCNS LCS

 VTAM internal trace options active at the time of this dump

 API PIU MSG SSCP NRM

 Pages in VTAM internal trace table (Decimal) =        50

 VIT  - Start of VTAM internal trace table:      05F8D000
 VITC - Current VTAM internal trace table entry: 05F92EE0
 VITO - Oldest VTAM internal trace table entry:  05F92F00
 VITE - End of VTAM internal trace table:        05FBF000
VTAM internal trace table (oldest to newest entry)
 VITPAGE: 0624A5A0
    +0000  D8E4C500  1F480110  06337250  063372A0  | QUE........&.... |
    +0010  824C3110  062CDB50  C1D7D9D7  06366010  | b<.....&APRP..-. |
    +0020  C5E7C9E3  1F000018  06337250  063552D8  | EXIT.......&...Q |
    +0030  825BC9F6  80000000  E3E2C9D9  06366010  | b$I6....TSIR..-. |
    +0040  C4E2D740  1F000110  06337250  063372A0  | DSP .......&.... |
    +0050  062CDB50  062CDB50  C1D7D9D7  06366010  | ...&...&APRP..-. |
    +0060  D8E4C500  1F4B2110  06337250  063372C0  | QUE........&...{ |
    +0070  81EAC990  062CDB50  C1D7E4C5  06366010  | a.I....&APUE..-. |
    +0080  C5E7C9E3  1F000050  06337250  063372A0  | EXIT...&...&.... |
    +0090  81EAC9E8  80000000  C1D7D9D7  06366010  | a.IY....APRP..-. |
    +00A0  D9C5D3E2  1F170000  06337250  06366010  | RELS.......&..-. |
    +00B0  824CD2A6  00000000  06366010  00000000  | b<Kw......-..... |
    +00C0  E2D9C2E7  1F000000  06337250  80000000  | SRBX.......&.... |
    +00D0  00000000  00F05500  00F05500  824C5BD8  | .....0...0..b<$Q |
    +00E0  E2D9C2C4  1F000000  06337250  80000000  | SRBD.......&.... |
    +00F0  00000000  00000000  007F0D18  00800000  | .........".......|
    +0100  C9D5E3E7  1D050000  F0C2C6F0  060B0818  | INTX....0BF0.... |
```

VTVRBLK

VTVRBLK looks at VRs for all subareas and displays the following:

- Number of subareas supported
- Number of VRBLKs processed
- Number of subareas containing virtual routes
- Number of subareas with no virtual routes
- Number of blocked routes found
- Number of held routes found.

For each blocked or held route found, the status areas for each transmission priority (TP0, TP1, and TP2) are analyzed and the following information is displayed if present:

- VR number
- Adjacent subarea
- Destination subarea
- Window sizes
- Pacing limit
- Inbound and outbound sequence numbers
- Selected flags
- VR FSM
- Flow control FSM
- Count of TSCBs on VR hold queue (if any)
- Last pacing request number.

Operands

Subarea

Specify a 1–8 hexadecimal digit number in the form X'x...' or a number between 1- 2147483647. This represents the number of the subarea. See field ATCSASUP in the ATCVT for the maximum number of subareas available in a dump. Subarea 0 is not valid.

If you are not sure which subarea, if any, is having a problem, specify no subarea to analyze all VRs for all destination subareas.

Report Data

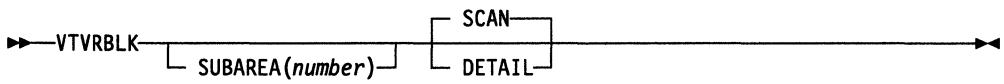
Use **Detail** to display further information on every VR found, similar to the information described above for blocked and held routes. **Scan** is the default and provides count information based on the search criteria.

Additional Information

See *VTAM Data Areas for MVS/ESA* for more information on the VRBLK and its contents.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for VTVRBLK

VTVRBLK

```
                              VTVRBLK Analysis


                              Route Blocked

Subarea (Decimal) =             1

  VR block 00690F18     VR number 02    ADJSUBA 00000003    DESTSUBA 00000001
    Transmission priority  0  VRFSM      05  FCFSM    00
    Window sizes:
          Current (VRBPALIM)             06
          Minimum (VRBMINWS)             02
          Maximum (VRBMAXWS)             06
          Pacing request send count      0000
          Inbound sequence number        0218
          Outbound sequence number       020E
          Last pacing for request number 0212
    Status flags:
  VRBCWRI  = 0  Route change window response NOT required
  VRBRWI   = 0  Route RESET window NOT required
  VRBHLD   = 1  Half session held NOT required
  VRBSCNHQ = 0  All HSQHs have been checked for held sessions


Host subarea skipped          2

VRBLKs processed             32
Subareas supported          511
Subareas with VRs             8
Subareas without VRs        503
Blocked routes                1
Held routes                   0
```

VTWRE

Use VTWRE to count or help analyze waiting request elements (WREs). A WRE represents a VTAM process that is waiting for the completion of some event. A WRE contains a pointer to an event ID that indicates the reason for the wait state. WREs are queued to locked queue anchor blocks (LQABs). The LQABs to be looked at might be:

- All LQABs
- An SSCP I/O LQAB for a specific subarea
- Another specific LQAB.

If you invoke VTWRE without operands, the number of WREs queued to all LQABs are counted and the counts are displayed.

Use the DETAIL option to get additional information on each WRE. Also, several operands are available to limit processing to a specific LQAB or to specific WREs.

Operands

Event ID

Enter a 1–100 hexadecimal digit value to be used in matching a WRE event ID found in the dump. The specified event ID is left-justified when comparing with the contents of the dump. For example, if X'1234' is specified and the dump contained X'F1F0F41234', this would not be a match.

Event ID Code

Enter a 4 hexadecimal digit value. If the entered code is fewer than 4 digits, results are unpredictable. Only WREs containing this event ID code are processed. The event ID code identifies the reason for the wait state.

LQAB

Enter a specific LQAB name from the following list to limit processing to a single LQAB. The following LQABs, which are pointed to by the ATCVT, can be examined. The default is ALL.

LQAB name	Description
IOLQB	SSCP I/O LQABs (one per attached subarea)
LUSMQ	Service manager LQAB
MCQAB	Miscellaneous command LQAB
PULQB	Physical unit services LQAB
NOSQ	Network operator services LQAB
SSLQB	Miscellaneous LQAB for session services
SSMQB	Second miscellaneous LQAB for session services
All	All of the above LQABs.

If a subarea is specified, the LQAB must be entered as IOLQB. If ALL is specified all subareas defined to IOLQB by ATCSASUP will be processed.

Subarea

Use this operand to limit processing to the SSCP I/O LQAB for a specific subarea. When a subarea is specified, IOLQB must be specified for the Queue, and Subarea is used as an index into the SSCP I/O LQABs. Specify a 1–8 hexadecimal digit number in the form X'x...' or a number between 1-2147483647. This represents the number of the subarea. See field ATCSASUP in the ATCVT for the maximum number of subareas available in a dump. Subarea 0 is not valid.

Mask

Enter a 1–100 hexadecimal digit mask. The mask is left justified and ANDed with the event identifier in the dump to determine whether the specified event identifier was found.

Note: Mask must be used with Event ID.

Max

Enter a number between 1–99999 (1–5 decimal digits or 1–4 hexadecimal digits) for the maximum number of WREs to be processed for the selected LQAB(s). The default is 100.

Control Op Code

Only WREs containing this CPCB op code are eligible for selection. The Control Op Code must be 1–8 hexadecimal digits in the form X'x...'. If the op code is fewer than eight digits, it is left-justified and compared with the leftmost digits in the dump.

User Correlator

Enter a 1–8 character value. Only WREs containing this User Request Correlator (URC) are processed. The URC is typically the resource name of the target of a request.

Format

Use **Format** to format the WRE and the EID, if present. With **Noformat**, which is the default, the WRE and EID are not formatted.

Note: Do not specify **Format** if you use **Detail** for Report Data. **Format** is valid only for **Scan**.

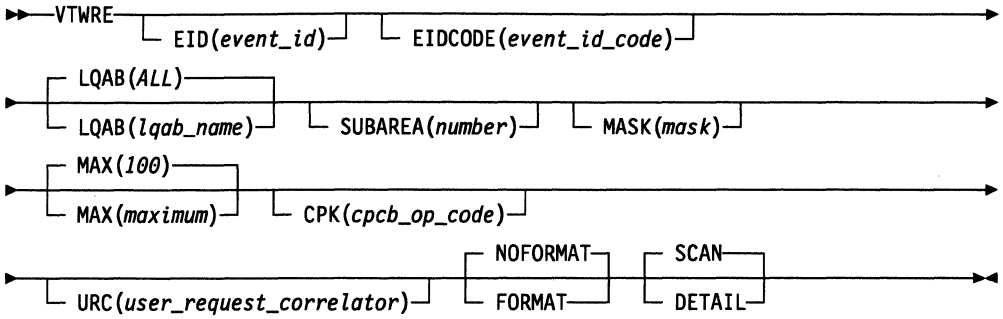
Report Data

Scan, which is the default, counts and displays the number of WREs meeting the specified search criteria. Use **Detail** to have the following fields extracted and displayed from each selected WRE:

- WRE address
- Control block (RUPE) address
- CPCBOPC
- URC
- Event ID
- Event ID Code.

Use the following syntax as an alternative to the panel interface.

Syntax



Sample Output for VTWRE

VTWRE LQAB(IOLQB) DETAIL

```
                              VTWRE Analysis
ATCIOLQB
WRE ADDR RUPEADDR CPCBOPC    URC     CODE             EVENT ID
-------- -------- -------- -------- ---- ----------------------------------
0666D200 0664B840 08810680 ECHO50Z  0201 00000001007100000001000102010881068O
Elements found:                 1
Elements processed:             1
```

Chapter 8. Using Traces

This chapter describes when to use traces and shows where in the network you can use each trace to collect data (see Figure 33 on page 348). Examples are included to help you interpret trace output.

This chapter includes the following sections:

- "Traces Provided by VTAM"
- "Traces Provided by NCP" on page 384.

Traces Provided by VTAM

The VTAM program provides several kinds of traces to record the flow of network events. Each trace occurs at a different point in the network (see Figure 33 on page 348). This difference allows you to narrow down the problem by following a request/response unit (RU) through the network and determining where in the network the RU is incorrect. (The RU could be out of sequence or lost, the data in the RU could have been changed, and so forth.)

This section includes the following information:

- "Activating Network Traces" on page 349
- "Formatting and Printing Trace Records" on page 361
- "Trace Output" on page 367
- VTAM traces and their results
  - The buffer contents trace shows the contents of inbound and outbound message buffers. See "Buffer Contents Trace" on page 369 for more information.
  - The I/O trace shows (in order) all I/O sent between VTAM and a particular network resource. See "I/O Trace" on page 373 for more information.
  - The SMS (buffer use) trace shows information about the use of buffers, including how often a buffer pool has expanded, how many buffers are currently being used, and what was the maximum number of buffers used since the last trace record was written. See "SMS (Buffer Use) Trace" on page 377 for more information.
  - MVS The TGET/TPUT trace shows each message as it passes between a TSO command processor and TSO/VTAM. See "TGET/TPUT Trace for TSO/VTAM (MVS)" on page 380 for more information.

The VTAM internal trace (VIT) is discussed in Chapter 9, "Using the VTAM Internal Trace (VIT)" on page 397.

The VM VSCS trace facility is discussed in Chapter 11, "Using VSCS Service Aids (VM)" on page 449.

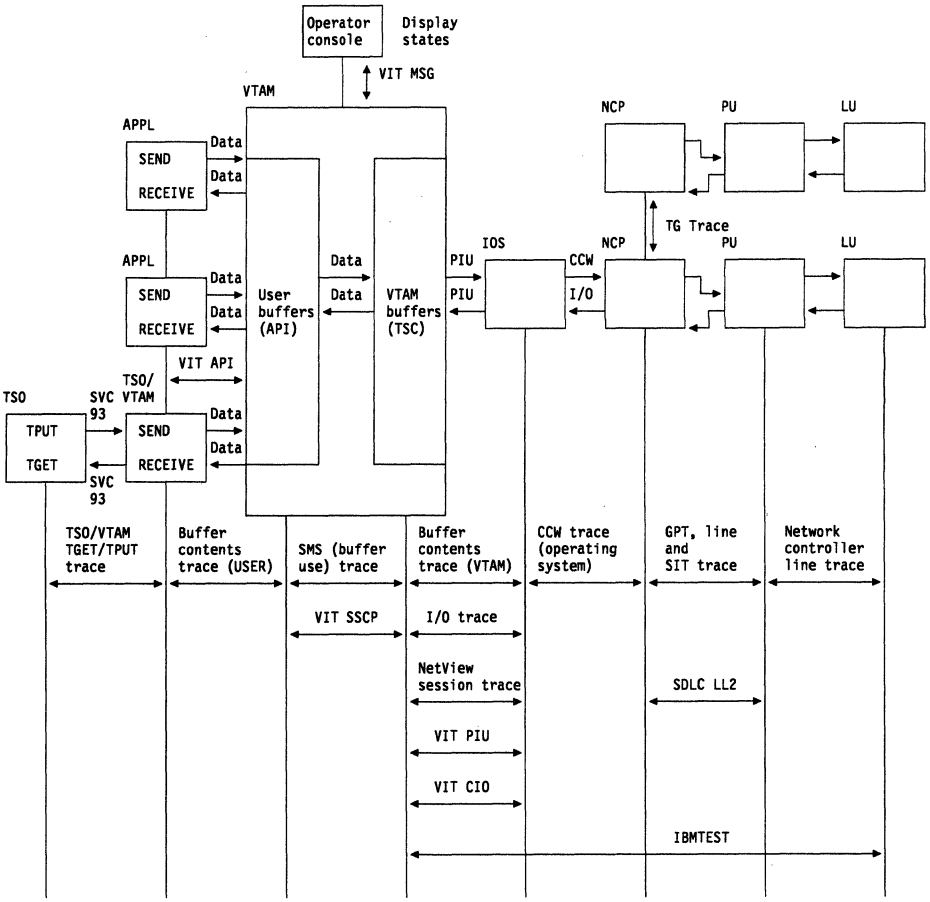


Figure 33. Network Traces Provided by VTAM

Activating Network Traces

You can activate VTAM traces when you start VTAM, using the TRACE option on the START command, or you can activate them when VTAM is already running, using the MODIFY TRACE command. Table 15 on page 350 shows the format of both commands for each type of trace.

Important:

1. The TRACE start option and its qualifiers must be coded on one line.
2. MVS GTF **must** be active to record VTAM traces externally.
3. VM "VTAM" must precede each of these commands. For example, to start a buffer contents trace you must enter:

   ```
   VTAM MODIFY TRACE,TYPE=BUF,ID=name
   ```

   VM VTAM and VSCS pass all trace data to GCS, which passes the data to TRSOURCE, which records all system trace data. Therefore, TRSOURCE *must* be active with TYPE=GT. In addition, you must enter a GCS ETRACE command (with the GTRACE and GROUP operands) for the VTAM and VSCS virtual machine.

   For more information see "Collecting External Trace Data (VM)" on page 359.

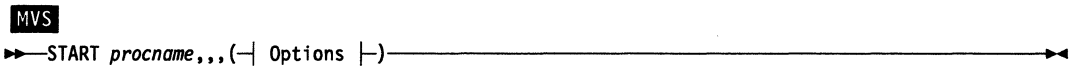
4. VSE VTAM passes all trace data except `TYPE=SMS` records to the VTAM trace file. For `TYPE=SMS`, SDAID must be active with `TRACE VTAMBU` specified.

For more information on activating these traces and optional operands not shown in this table, see "TRACE for Buffer, I/O, NCP Line, SIT, or SMS Traces" in *VTAM Resource Definition Reference* and "MODIFY TRACE Command" in *VTAM Operation*.

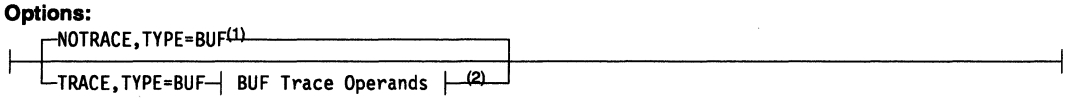
Table 15 (Page 1 of 9). Activating Network Traces

Buffer contents trace

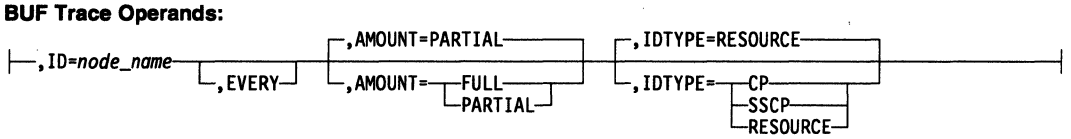
MVS



Options:



BUF Trace Operands:

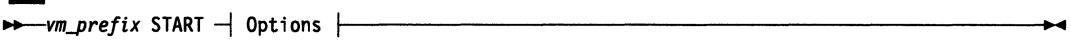


Notes:

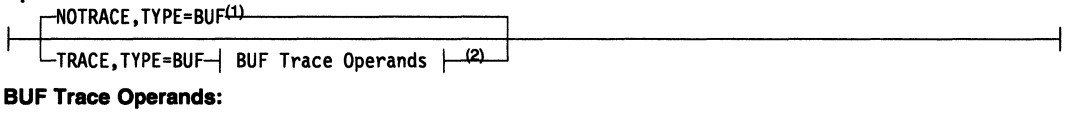
1 Do not use NOTRACE when starting VTAM, except to override a TRACE start option coded in a predefined list.

2 Code TRACE and its qualifiers on one line. Code the TYPE qualifier immediately following TRACE.

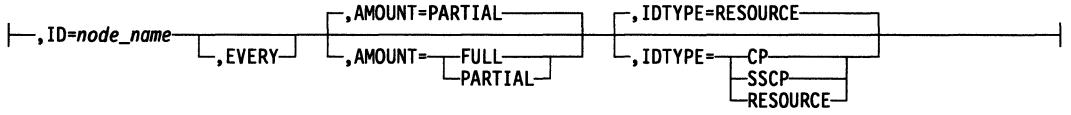
VM



Options:



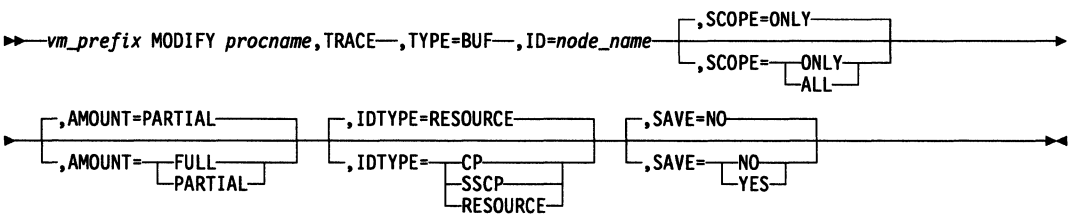
BUF Trace Operands:



Notes:

1 Do not use NOTRACE when starting VTAM, except to override a TRACE start option coded in a predefined list.

2 Code TRACE and its qualifiers on one line. Code the TYPE qualifier immediately following TRACE.



vm_prefix VM

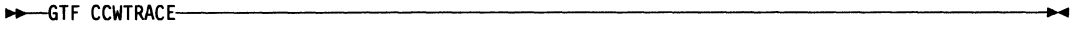
is the prefix required for commands issued to VTAM running under VM. This prefix (usually **VTAM**) is determined by an operand of the GCS LOADCMD command. Use the prefix required by your network.

MVS, VSE Do not use a command prefix.

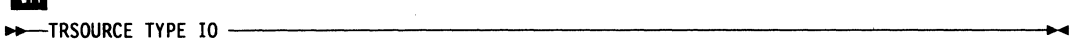
Table 15 (Page 2 of 9). Activating Network Traces

CCW trace

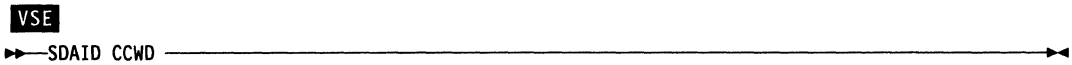
MVS



VM



VSE

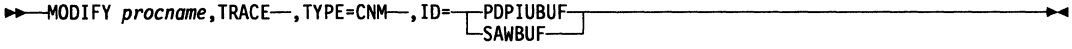


See your operating system books for more information on the CCW trace.

CNM trace (communication network management)

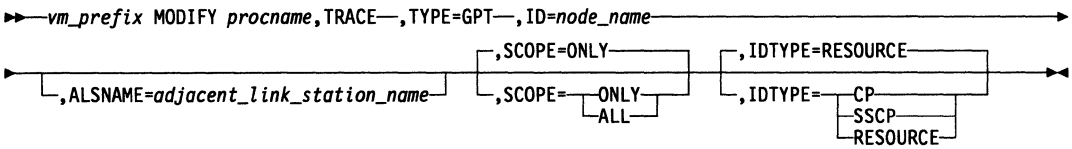
Cannot be activated with the TRACE start option.

MVS



GPT (generalized PIU trace)

Cannot be activated with the TRACE start option.



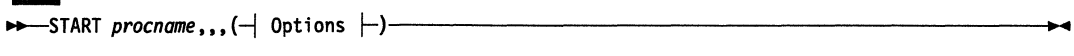
vm_prefix VM
is the prefix required for commands issued to VTAM running under VM. This prefix (usually **VTAM**) is determined by an operand of the GCS LOADCMD command. Use the prefix required by your network.

MVS,VSE Do not use a command prefix.

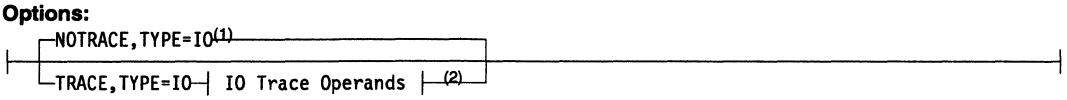
Table 15 (Page 3 of 9). Activating Network Traces

I/O trace

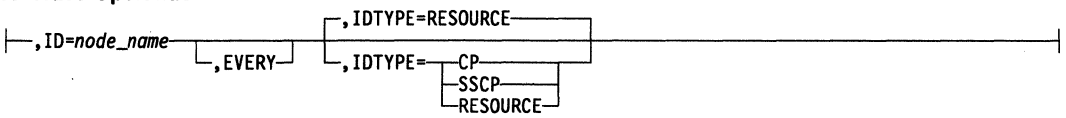
MVS



Options:



IO Trace Operands:

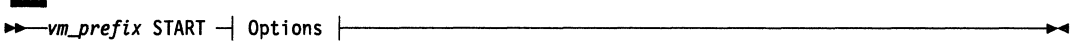


Notes:

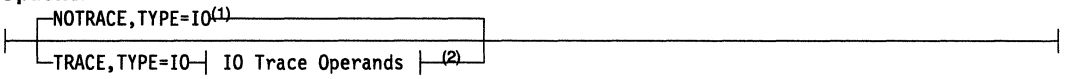
1 Do not use NOTRACE when starting VTAM, except to override a TRACE start option coded in a predefined list.

2 Code TRACE and its qualifiers on one line. Code the TYPE qualifier immediately following TRACE.

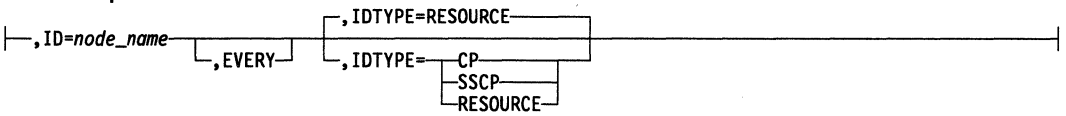
VM



Options:



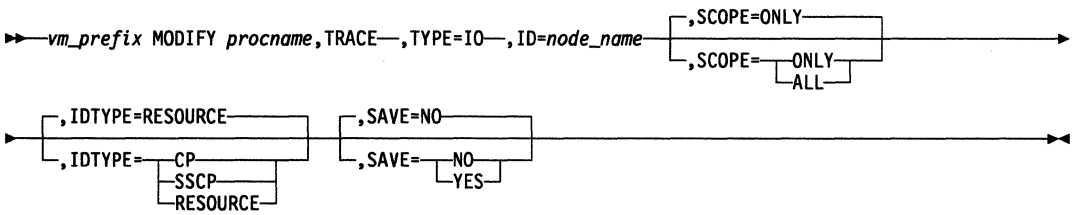
IO Trace Operands:



Notes:

1 Do not use NOTRACE when starting VTAM, except to override a TRACE start option coded in a predefined list.

2 Code TRACE and its qualifiers on one line. Code the TYPE qualifier immediately following TRACE.



vm_prefix VM

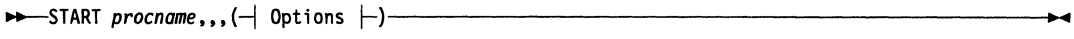
is the prefix required for commands issued to VTAM running under VM. This prefix (usually **VTAM**) is determined by an operand of the GCS LOADCMD command. Use the prefix required by your network.

MVS,VSE Do not use a command prefix.

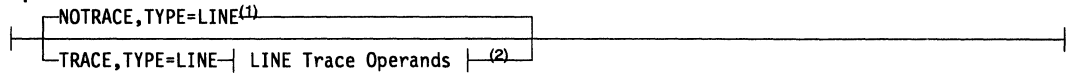
Table 15 (Page 4 of 9). Activating Network Traces

Line trace

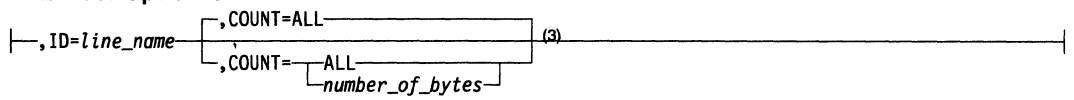
MVS



Options:



LINE Trace Operands:



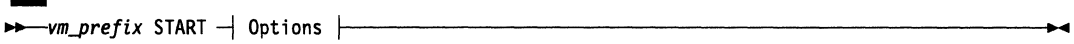
Notes:

1 Do not use NOTRACE when starting VTAM, except to override a TRACE start option coded in a predefined list.

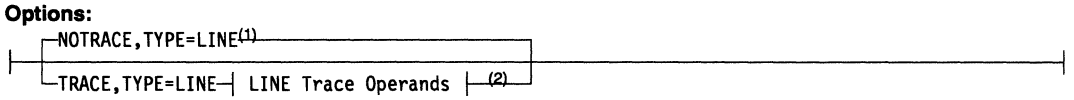
2 Code TRACE and its qualifiers on one line. Code the TYPE qualifier immediately following TRACE.

3 COUNT applies only to the IBM 3720 and 3745 Communication Controllers.

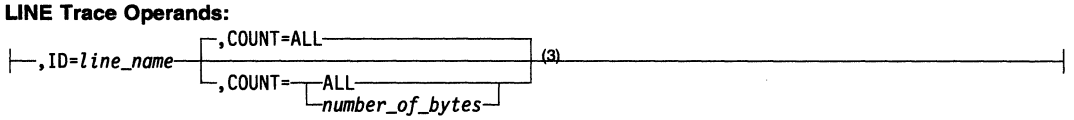
VM



Options:



LINE Trace Operands:

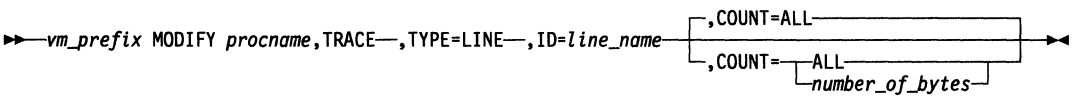


Notes:

1 Do not use NOTRACE when starting VTAM, except to override a TRACE start option coded in a predefined list.

2 Code TRACE and its qualifiers on one line. Code the TYPE qualifier immediately following TRACE.

3 COUNT applies only to the IBM 3720 and 3745 Communication Controllers.

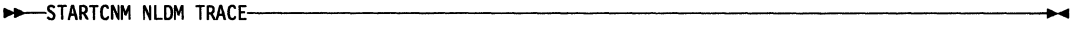


vm_prefix VM

is the prefix required for commands issued to VTAM running under VM. This prefix (usually **VTAM**) is determined by an operand of the GCS LOADCMD command. Use the prefix required by your network.

MVS,VSE Do not use a command prefix.

NetView session trace

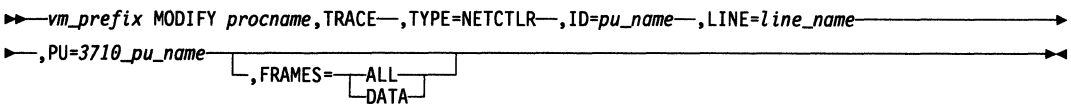


For more information on the NetView session trace, see *NetView Operation*.

Table 15 (Page 5 of 9). Activating Network Traces

Network controller line trace

Cannot be activated with the TRACE start option.



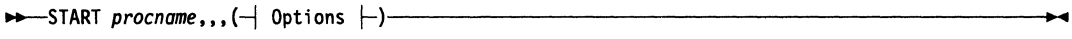
vm_prefix VM
is the prefix required for commands issued to VTAM running under VM. This prefix (usually **VTAM**) is determined by an operand of the GCS LOADCMD command. Use the prefix required by your network.

MVS,VSE Do not use a command prefix.

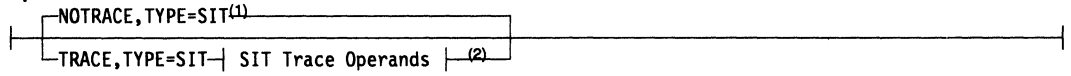
Table 15 (Page 6 of 9). Activating Network Traces

SIT (scanner interface trace)

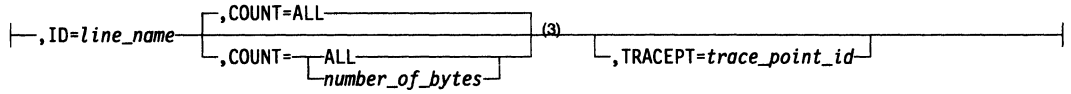
MVS



Options:



SIT Trace Operands:



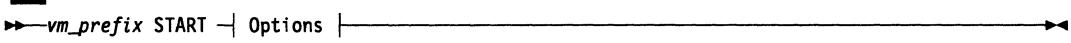
Notes:

1 Do not use NOTRACE when starting VTAM, except to override a TRACE start option coded in a predefined list.

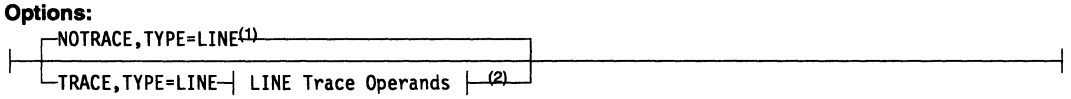
2 Code TRACE and its qualifiers on one line. Code the TYPE qualifier immediately following TRACE.

3 COUNT applies only to the IBM 3720 and 3745 Communication Controllers.

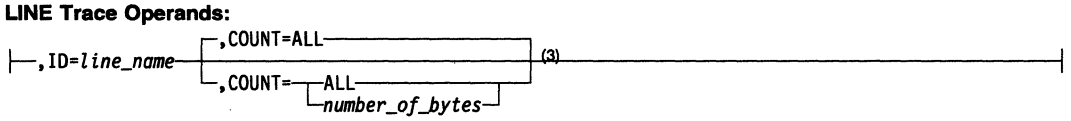
VM



Options:



LINE Trace Operands:

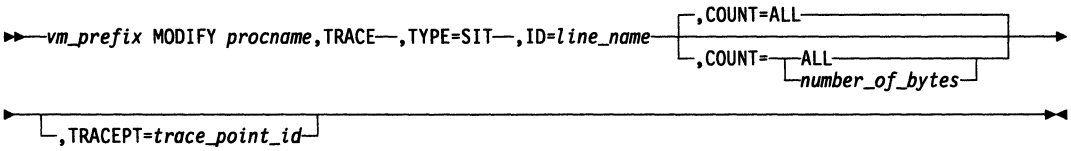


Notes:

1 Do not use NOTRACE when starting VTAM, except to override a TRACE start option coded in a predefined list.

2 Code TRACE and its qualifiers on one line. Code the TYPE qualifier immediately following TRACE.

3 COUNT applies only to the IBM 3720 and 3745 Communication Controllers.



vm_prefix VM

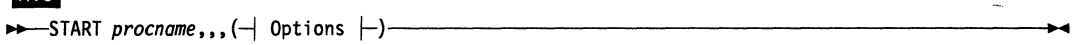
is the prefix required for commands issued to VTAM running under VM. This prefix (usually **VTAM**) is determined by an operand of the GCS LOADCMD command. Use the prefix required by your network.

MVS,VSE Do not use a command prefix.

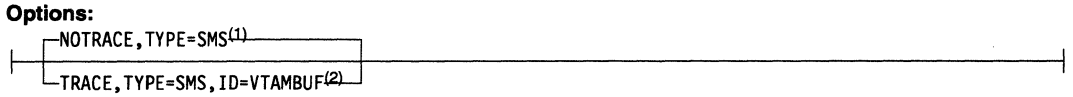
Table 15 (Page 7 of 9). Activating Network Traces

SMS (buffer use) trace

MVS



Options:

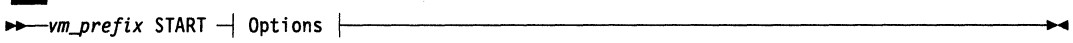


Notes:

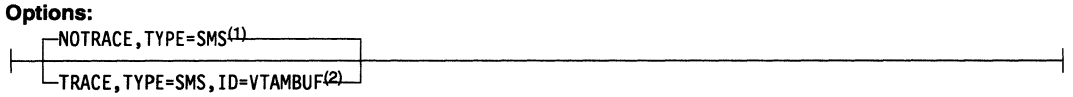
1 Do not use NOTRACE when starting VTAM, except to override a TRACE start option coded in a predefined list.

2 Code TRACE and its qualifiers on one line. Code the TYPE qualifier immediately following TRACE.

VM



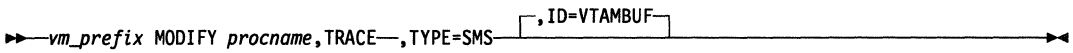
Options:



Notes:

1 Do not use NOTRACE when starting VTAM, except to override a TRACE start option coded in a predefined list.

2 Code TRACE and its qualifiers on one line. Code the TYPE qualifier immediately following TRACE.

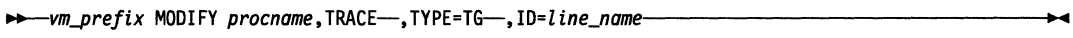


vm_prefix VM
is the prefix required for commands issued to VTAM running under VM. This prefix (usually **VTAM**) is determined by an operand of the GCS LOADCMD command. Use the prefix required by your network.

MVS,VSE Do not use a command prefix.

TG trace (transmission group)

Cannot be activated with the TRACE start option.



vm_prefix VM
is the prefix required for commands issued to VTAM running under VM. This prefix (usually **VTAM**) is determined by an operand of the GCS LOADCMD command. Use the prefix required by your network.

MVS,VSE Do not use a command prefix.

TSO/VTAM TGET/TPUT trace

Cannot be activated with the TRACE start option.

MVS

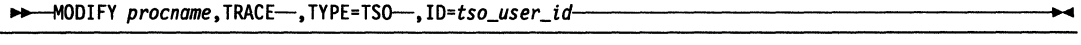
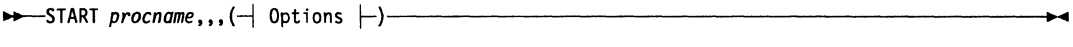


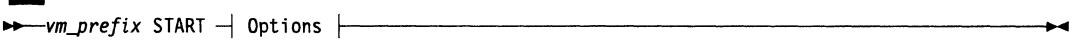
Table 15 (Page 8 of 9). Activating Network Traces

VTAM internal trace (VIT)

MVS



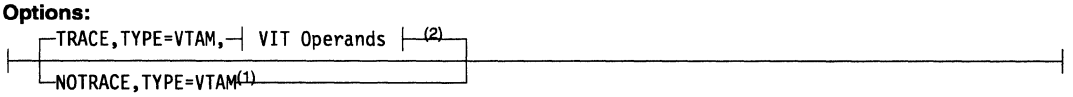
VM



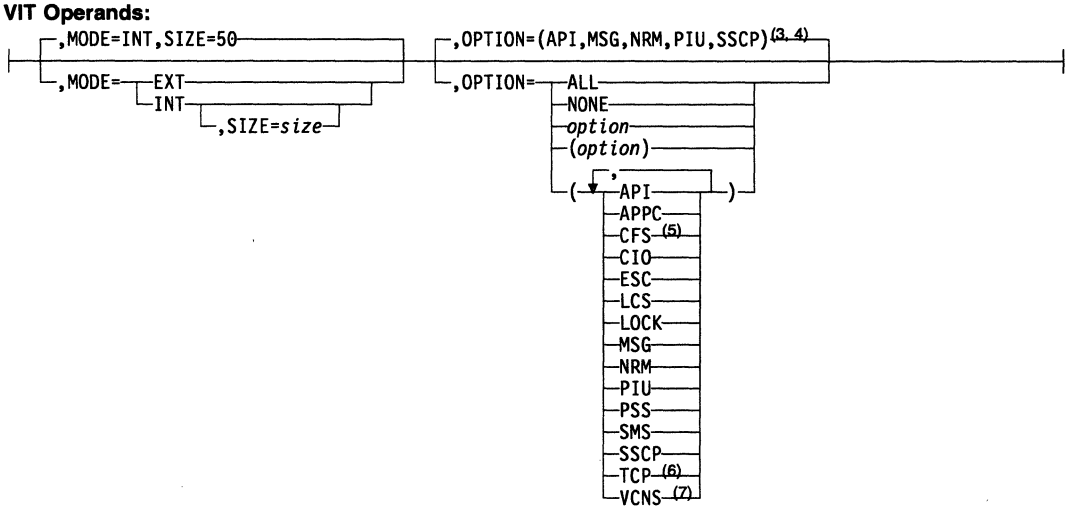
Notes:

1. The start options are listed in this section alphabetically; however, you can code them in any order.
2. MVS Precede the option list with three commas and enclose the group of options in parentheses.
3. Start options that are entered on the START command must be separated by commas. Do not leave any blanks between options.

Options:



VIT Operands:



Notes:

1 NOTRACE,TYPE=VTAM is accepted but ignored. Tracing is started with the default trace table size and the default options.

2 Code TRACE and its qualifiers on one line. Code the TYPE qualifier immediately following TRACE.

3 The default options apply only to MODE=INT for MVS and VM.

4 The default options apply to MODE=INT and MODE=EXT for VSE.

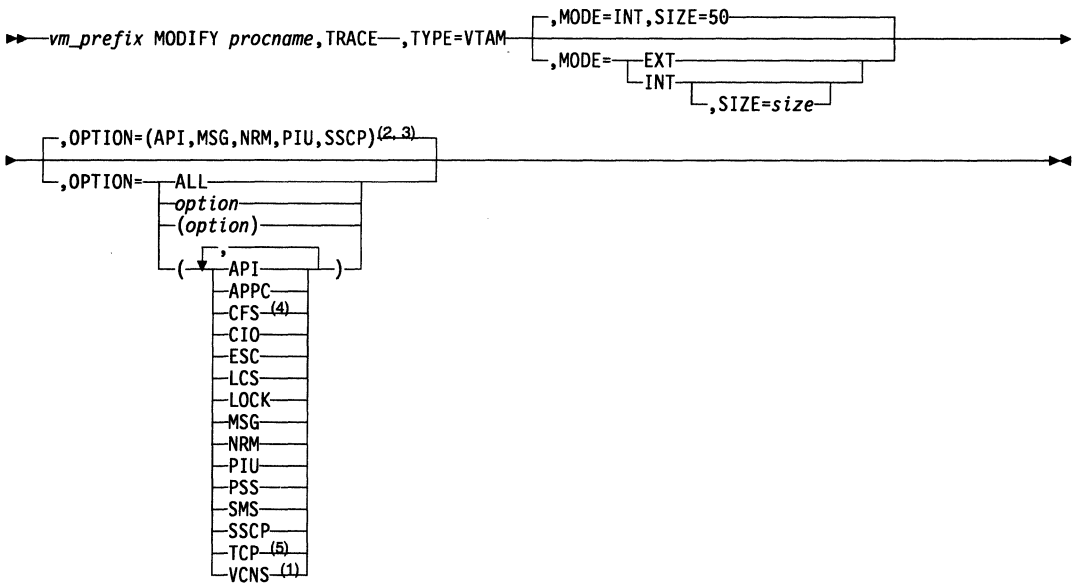
5 The CFS trace option applies to MVS only.

6 The TCP trace option applies to MVS only.

7 The VCNS trace option applies to MVS and VM only.

Table 15 (Page 9 of 9). Activating Network Traces

VIT (continued)



Notes:

1 The VCNS trace option applies to MVS and VM only.

2 MVS,VM These five options are defaults for MODE=INT. There are no default options for MODE=EXT.

3 VSE These five options are defaults for either MODE=INT or MODE=EXT.

4 The CFS trace option applies to MVS only.

5 The TCP trace option applies to MVS only.

vm_prefix VM
is the prefix required for commands issued to VTAM running under VM. This prefix (usually **VTAM**) is determined by an operand of the GCS LOADCMD command. Use the prefix required by your network.

MVS,VSE Do not use a command prefix.

Starting the Generalized Trace Facility (GTF) (MVS)

Because VTAM passes all external trace data to the generalized trace facility (GTF), GTF must be active to use VTAM traces. Specify TRACE=USRP to receive a prompt for all VTAM traces except the VTAM I/O trace. You will then be able to select the specific event identifier (EID) to be traced. Refer to APAR II03922 for additional information.

Note: If you do not limit the GTF trace output using the USRP option, GTF collects all USR events issued in the MVS system, often resulting in a large amount of unwanted information.

```
NC0000000 V535M433 93306 14:25:04.88 01       00000290  S FVGTF.B
N 4020000 V535M433 93306 14:25:10.06 STC00017 00000090  AHL121I  TRACE OPTION INPUT INDICATED FROM MEMBER GTFPARM  OF PDS
S                                                        SYS1.PARMLIB
N 0020000 V535M433 93306 14:25:10.36 STC00017 00000090  TRACE=SYSM,USR,TRC,DSP,PCI,SRM
N 4020000 V535M433 93306 14:25:10.38 STC00017 00000090  AHL103I  TRACE OPTIONS SELECTED --SYSM,USR,TRC,DSP,PCI,SRM
W 4020000 V535M433 93306 14:25:10.38 STC00017 00000090 *25 AHL125A  RESPECIFY TRACE OPTIONS OR REPLY U
NC0000000 V535M433 93306 14:25:19.47 01       00000290  R 25,TRACE=USRP
NR4020000 V535M433 93306 14:25:19.57          00000090  IEE600I REPLY TO 25 IS;TRACE=USRP
N 0020000 V535M433 93306 14:25:19.73 STC00017 00000090  TRACE=USRP
W 4020000 V535M433 93306 14:25:19.87 STC00017 00000090 *26 AHL101A  SPECIFY TRACE EVENT KEYWORDS --USR=
NC0000000 V535M433 93306 14:25:35.68 01       00000290  R 26,USR=(FEF,FF1)
NR4020000 V535M433 93306 14:25:35.76          00000090  IEE600I REPLY TO 26 IS;USR=(FEF,FF1)
N 0020000 V535M433 93306 14:25:35.88 STC00017 00000090  USR=(FEF,FF1)
W 4020000 V535M433 93306 14:25:35.94 STC00017 00000090 *27 AHL102A  CONTINUE TRACE DEFINITION OR REPLY END
NC0000000 V535M433 93306 14:25:41.00 01       00000290  R 27,END
NR4020000 V535M433 93306 14:25:41.09          00000090  IEE600I REPLY TO 27 IS;END
N 0020000 V535M433 93306 14:25:41.25 STC00017 00000090  END
N 4020000 V535M433 93306 14:25:41.25 STC00017 00000090  AHL103I  TRACE OPTIONS SELECTED --USR=(FEF,FF1)
W 4020000 V535M433 93306 14:25:41.25 STC00017 00000090 *28 AHL125A  RESPECIFY TRACE OPTIONS OR REPLY U
NC0000000 V535M433 93306 14:25:46.83 01       00000290  R 28,U
NR4020000 V535M433 93306 14:25:46.99          00000090  IEE600I REPLY TO 28 IS;U
N 4020000 V535M433 93306 14:25:48.30 STC00017 00000090  AHL031I GTF INITIALIZATION COMPLETE
```

Figure 34. Starting the VTAM Buffer Contents Trace

Specify TRACE=RNIO for the VTAM I/O trace.

```
NC0000000 V535M433 93306 14:49:46.17 01       00000290  S FVGTF.B
N 4020000 V535M433 93306 14:49:49.85 STC00021 00000090  AHL121I  TRACE OPTION INPUT INDICATED FROM MEMBER GTFPARM  OF PDS
S                                                        SYS1.PARMLIB
N 0020000 V535M433 93306 14:49:49.93 STC00021 00000090  TRACE=SYSM,USR,TRC,DSP,PCI,SRM
N 4020000 V535M433 93306 14:49:49.95 STC00021 00000090  AHL103I  TRACE OPTIONS SELECTED --SYSM,USR,TRC,DSP,PCI,SRM
W 4020000 V535M433 93306 14:49:49.96 STC00021 00000090 *36 AHL125A  RESPECIFY TRACE OPTIONS OR REPLY U
NC0000000 V535M433 93306 14:49:57.27 01       00000290  R 36,TRACE=RNIO
NR4020000 V535M433 93306 14:49:57.32          00000090  IEE600I REPLY TO 36 IS;TRACE=RNIO
N 0020000 V535M433 93306 14:49:57.39 STC00021 00000090  TRACE=RNIO
N 4020000 V535M433 93306 14:49:57.49 STC00021 00000090  AHL103I  TRACE OPTIONS SELECTED --RNIO
W 4020000 V535M433 93306 14:49:57.49 STC00021 00000090 *37 AHL125A  RESPECIFY TRACE OPTIONS OR REPLY U
NC0000000 V535M433 93306 14:50:02.97 01       00000290  R 37,U
NR4020000 V535M433 93306 14:50:02.99          00000090  IEE600I REPLY TO 37 IS;U
N 4020000 V535M433 93306 14:50:03.97 STC00021 00000090  AHL031I GTF INITIALIZATION COMPLETE
NC0000000 V535M433 93306 14:50:27.67 01       00000290  F NET,TRACE,TYPE=IO,ID=APPL1
NR0000000 V535M433 93306 14:50:27.74 STC00008 00000090  IST097I NOCTR MODIFY ACCEPTED
NR0000000 V535M433 93306 14:50:27.74 STC00008 00000090  IST513I INCTR TRACE INITIATED FOR NODE APPL1
NC0000000 V535M433 93306 14:50:45.17 01       00000290  F NET,NOTRACE,TYPE=IO,ID=APPL1
NR0000000 V535M433 93306 14:50:45.21 STC00008 00000090  IST097I NOCTR MODIFY ACCEPTED
NR0000000 V535M433 93306 14:50:45.24 STC00008 00000090  IST512I INCTR TRACE TERMINATED FOR NODE = APPL1
```

Figure 35. Starting the VTAM I/O Trace

See Table 80 on page 1165 to determine which book contains more information on GTF.

Collecting External Trace Data (VM)

This section describes how to collect external trace output in VM. For information on starting and stopping the VSCS trace, see "Using the External VSCS Traces" on page 460.

VM external trace data is contained in TRF files. The size of each TRF file is defined on the TRSAVE command. When a TRF file reaches that size, it is closed automatically, and another is opened. When the total number of closed TRF files exceeds the TRSAVE KEEP value (default is 2), CP will automatically purge the oldest TRF file when the newest one is closed.

Important: To collect VTAM trace output in VM, issue the following commands from the VTAM console in the EXACT order listed here.

Notes:

1. If the VTAM virtual machine has privilege class A or C, all of the following commands can be issued directly from it. If the VTAM virtual machine does not have the authority to issue these commands, you can issue them from any virtual machine having the correct authority.
2. Unless it is defined otherwise, *userid* is the receiver of the TRF files. It is also the user ID from which the traces are processed.

`CP TRSAVE ID` *traceid* `TO` *userid* — *traceid* is the modifier referred to by subsequent TRSOURCE, TRSAVE, and TRACRED commands.

`CP TRSOURCE ID` *traceid* `TYPE GT FOR VMGROUP` *gcsname* — *gcsname* is the name of the GCS group that is being traced. It is the NSS name that was IPL-ed. To determine the GCS group name, issue `QUERY GROUP` from the VTAM virtual machine.

`CP TRSOURCE ENABLE ID` *traceid*
`ETRACE GTRACE GROUP`
`VTAM F TRACE,TYPE=`*type*`,...`

VTAM starts external tracing. When you have collected enough data, and you want to stop tracing, issue the following commands:

`ETRACE GTRACE OFF GROUP` — You must specify `GROUP`.
`CP TRSOURCE DISABLE ID` *traceid*
`VTAM F NOTRACE,TYPE=`*type*`,...`

At this point, the user ID specified on the TRSAVE command has all the TRF files of the name *traceid* specified in the TRSOURCE and TRSAVE commands. To process these TRF files, logon to the user ID.

Use the CP QUERY TRFILES command to determine whether TRF files were produced, the number and size of the files, and the user ID that produced them.

VTAM records trace entries in every virtual machine in the GCS group that contains a VTAM application, not only in the VTAM virtual machine. Also, a small number of trace entries are recorded in the GCS recovery machine. Consequently, it is normal to have TRF files from multiple virtual machines. If an application virtual machine's user ID is not listed under the ORIGINID header on the CP QUERY TRFILES command, then that user ID did not record trace entries.

See "Formatting and Printing Trace Records" on page 361. To determine what book contains information on using TRSAVE and TRSOURCE, see Table 80 on page 1165

Virtual Machine Tracing (VM)

The PER command does not apply to VM. The equivalent command is CP TRACE; however, you can specify PER as an alias for the TRACE command.

See Table 80 on page 1165 to determine what book contains more information on the CP TRACE command.

Formatting and Printing Trace Records

Table 16 indicates which traces can be formatted and printed by each of the formatting programs.[3] Descriptions of the programs appear later in the chapter. In this table:

- FP indicates that the trace is both formatted and printed (for VTAM records only)
- UP indicates that the trace is printed but not formatted.
- A blank entry indicates that the trace is neither formatted nor printed.

Table 16. Processing Externally Recorded Trace Data

Trace	ACF/TAP	IPCS MVS	TRACERED VM	TPRINT VSE	DOSVSDMP VSE
Buffer contents trace	FP	FP	FP	FP	
Generalized PIU trace	FP				
I/O trace	FP	FP	FP	FP	
Line trace[5]	FP	FP[4]	FP[4]	FP[4]	
Network controller line trace	FP	FP[4]	FP[4]	FP[4]	
Scanner interface trace	FP				
SMS (buffer use) trace	UP	FP	FP		FP
TGET/TPUT trace	UP MVS	UP			
Transmission group trace	FP	FP[4]	FP[4]	FP[4]	
VSCS internal trace			FP		
VTAM internal trace[6]	UP	FP	FP	FP	

Using ACF/TAP

Use the trace analysis program (ACF/TAP) to print all VTAM external trace entries or up to ten entry types. Table 17 lists the options to use on the INPUT operand for formatting and printing traces. See Table 80 on page 1165 to determine what book describes ACF/TAP.

Table 17 (Page 1 of 2). ACF/TAP Options

Specify:	To format and print:
LINE	Line trace
LINE	Scanner interface trace (3720, 3725, and 3745 only)
LINE	Transmission group trace
GPT	Generalized PIU trace
RNIO	I/O trace

3 This is not a complete list of programs that process external trace data.

4 Only scanner type 1, 2, and 3 records are processed. All others must be processed using ACF/TAP.

5 ACF/TAP must be used except for 3705 traces. See "Line Trace Operation" on page 386 for more information.

6 ACF/TAP allows you to specify some formatting parameters if the VIT is running in external mode.

Table 17 (Page 2 of 2). ACF/TAP Options

Specify:	To format and print:
BUFFER	Buffer contents trace
NETCTLR	Network controller line trace

Notes on using ACF/TAP:

1. The system services program (SSP) V3R8 must be installed before you can use ACF/TAP to format the VTAM full buffer contents trace.
2. ACF/TAP does not print a buffer contents trace that is traced at the API. Use IPCS GTFTRACE to print API (FF1) traces.
3. VM Do *not* use ACF/TAP to print VSCS trace output.

Using IPCS with GTF Trace Option (MVS)

To format and print the VTAM I/O trace, set RNIO on the GTFTRACE option. For all other VTAM traces, set USR(*symnum1[,symnum2]...[,symnum6]| ALL*) on the GTFTRACE option.

symnum can be either a symbolic name or a number representing the trace that you want formatted and printed. If you specify USR(ALL), IPCS formats and prints all user and subsystem traces recorded by GTF. For information on starting GTF, see "Starting the Generalized Trace Facility (GTF) (MVS)" on page 358.

Table 18 lists the valid symbols and numbers for the VTAM traces.

Table 18. Symbols and Numbers for Formatting and Printing VTAM Traces

Symbol	Number	Trace
INT1	FE1	VTAM internal trace
TPIO	FEF	VTAM buffer contents trace (TSC component) Trace output says "VTAM.."
CL01	FF1	VTAM buffer contents trace (API component) Trace output says "USER.."
CL02	FF0	SMS (buffer use) trace
LINE	FF2	NCP 37xx line or TG trace
APTH	FE2	TSO/VTAM TGET/TPUT trace
APTD	FE4	Line PIU, generalized PIU, or network controller line trace

Note: The symbol and the number can be used interchangeably, for example, USR(LINE) or USR(FF2); however, when starting GTF, use the number.

See Table 80 on page 1165 to determine what book describes how to use GTF and IPCS.

Using TRACERED (VM)

You can format and print trace output using TRACERED with the following commands:

`ACCESS 193 x`	Link to the minidisk that contains the GCS and VTAM formatting routines, usually 193. *x* is a user-specified file mode.
`TRACERED NAME traceid (ALL`	*traceid* is the collection of TRF files and is the same value as is specified in the TRSAVE and TRSOURCE commands. *traceid* will cause TRACERED to process all of the TRF files at once, producing a consolidated output file in chronological order.

See Table 80 on page 1165 to determine which book contains more information on TRACERED.

Using TPRINT (VSE)

Trace records in the external trace file can be printed, formatted or unformatted, by using the TPRINT utility program.

The trace print utility runs as a subtask under VTAM or as a job step under VSE. To run the trace print utility as a subtask, the operator initiates printing with the MODIFY SUBTASK command while VTAM is active. When the utility is run as a separate job step, printing can be done whether VTAM is active or inactive.

If TPRINT runs as a subtask, it is executed in the same address space in which VTAM runs and must be started in a static partition. VTAM JOB has the JCL for TPRINT. If TPRINT runs as a separate job, it can run in either a static or a dynamic partition, and SYS001 and SYS004 must be defined in the job, as shown in "Running TPRINT in a Separate Partition" on page 364.

Note: VTAM phases, macroinstructions, modules, and user-defined VTAM definitions reside in a group of sublibraries (library.sublibraries), which the system searches from left to right when loading is required. In the following examples, the system searches the user definition library and the VTAM library. Each of these sublibraries contains phases, macroinstructions, and modules. The system sublibrary default is IJSYSRS.SYSLIB, and it is searched last (by default).

Job Step Procedure: The trace print utility can be executed through a job step procedure that is defined when VTAM is installed. SYSLST is assigned to a tape, printer, or disk, and the trace file is defined in DLBL and EXTENT statements. For example:

```
// LIBDEF *,SEARCH=(PRD2.CONFIG,PRD1.BASE)                          1
// DLBL  TRFILE,'VTAM.TRACE.FILE'
// EXTENT SYS004
// ASSGN SYS004,cua
// ASSGN SYSLST,cua
   .
   .
   .
// EXEC TPRINT
```

SYSLST must be assigned to a printer, tape, or disk with the name IJSYSLS. If it is not assigned, TPRINT is canceled.

Running TPRINT in a Separate Partition: When TPRINT runs in a partition separate from VTAM, SYS001 and SYS004 must be defined for the partition in which TPRINT runs. The following is sample JCL defining SYS001 and SYS004 for the partition in which TPRINT runs. (In this sample, cua is the channel unit address for the device.)

```
// JOB TPRINT
// LIBDEF *,SEARCH=(PRD2.CONFIG,PRD1.BASE)                    1
// DLBL  TRFILE,'VTAM.TRACE.FILE'
// EXTENT SYS001
// ASSGN SYS001,cua
// ASSGN SYS004,cua
// ASSGN SYSLST,cua
// EXEC TPRINT
/&
```

MODIFY SUBTASK Command: To start the trace print facility with the MODIFY SUBTASK command, use the ID=TPRINT option. The operator is prompted to request a print of the TRFILE, using any of the print options available or a snapshot print. A snapshot prints all the I/O, BUF, LINE, and TNSTAT records in VTAM's main storage I/O buffers, without accessing records previously written to TRFILE. When the contents of TRFILE are being printed, all tracing is suspended until the printout is completed.

Note: When the trace print facility is run as a subtask, the trace output is not printed by VSE until VTAM is halted.

For more detailed information about initiating the trace print facility and the options available, see "Attaching the Trace Print Program as a VTAM Subtask (VSE)" in *VTAM Operation*.

All trace records contain either a time stamp or a formatted time and date field from which the order of recording can be determined. This information can be helpful in synchronizing buffer contents and I/O traces and, to a lesser extent, VTAM and non-VTAM traces.

The TPRINT User Exit Facility: An interface is provided in the trace print facility to allow the user to write an exit routine for the editing of VTAM trace records. The dummy user exit module, ISTRAEUE, can be replaced by an installation-supplied module. The dummy module is called by the trace writing module, ISTRACTO, with the following register contents:

Register	Contents
0	0 = first entry to module 1 = return for more output
1	Address of parameter list
13	Save area address
14	Caller's return address
15	ISTRAEUE entry point address

The parameter list pointed to by register 1 contains the following information:

- Word 1: Address of trace record (ISTTRAB)
- Word 2: Address of output buffer
- Word 3: Address of date and time stamp for this record.

Consider the following points when programming a user exit routine for the TPRINT facility:

- TPRINT provides support for reading TRFILE with VSE access methods DTFSD or DTFMT. The file format is standard variable blocked records with BLKSIZE=2048. Each physical record contains a dummy 16-byte logical record that provides control information for TPRINT and is not passed to the user exit routine. The block header and the dummy record are mapped by the TRB.
- The remaining logical records contain trace data. Each has a 32-byte header mapped by the TRAB. Each is passed to the user exit routine for processing. Contents of the TRAB header and data format and length vary according to the trace record type.
- TPRINT provides access method support for all SYSLST output. Line counting and page headers are provided automatically. DTFDI is used with ASA control characters. The output buffer is cleared to blanks before the user exit routine is called. Use of control characters is limited to the following:
  - Single space ( )
  - Double space (0)
  - Triple space (-)
  - Skip to channel 1 (1).

  If a skip is specified, TPRINT inserts its own page header and double spaces before printing the specified line.
- The TPRINT user exit routine can provide its own access-method interface to SYSLOG (console) or any other device. For example, you can require operator-specified, record-selection options to produce specific portions of trace records, such as line trace records or VIT entries.
- Buffer, I/O, and TNSTAT records are single entry data records. Line and internal trace records contain one or more entries following the TRAB header. The user exit routine must perform all de-blocking logic for multiple-entry records, using return code 8 to request re-entry from TPRINT for additional processing of the same trace record. See Figure 36 on page 366 for an example of user exit routine logic.

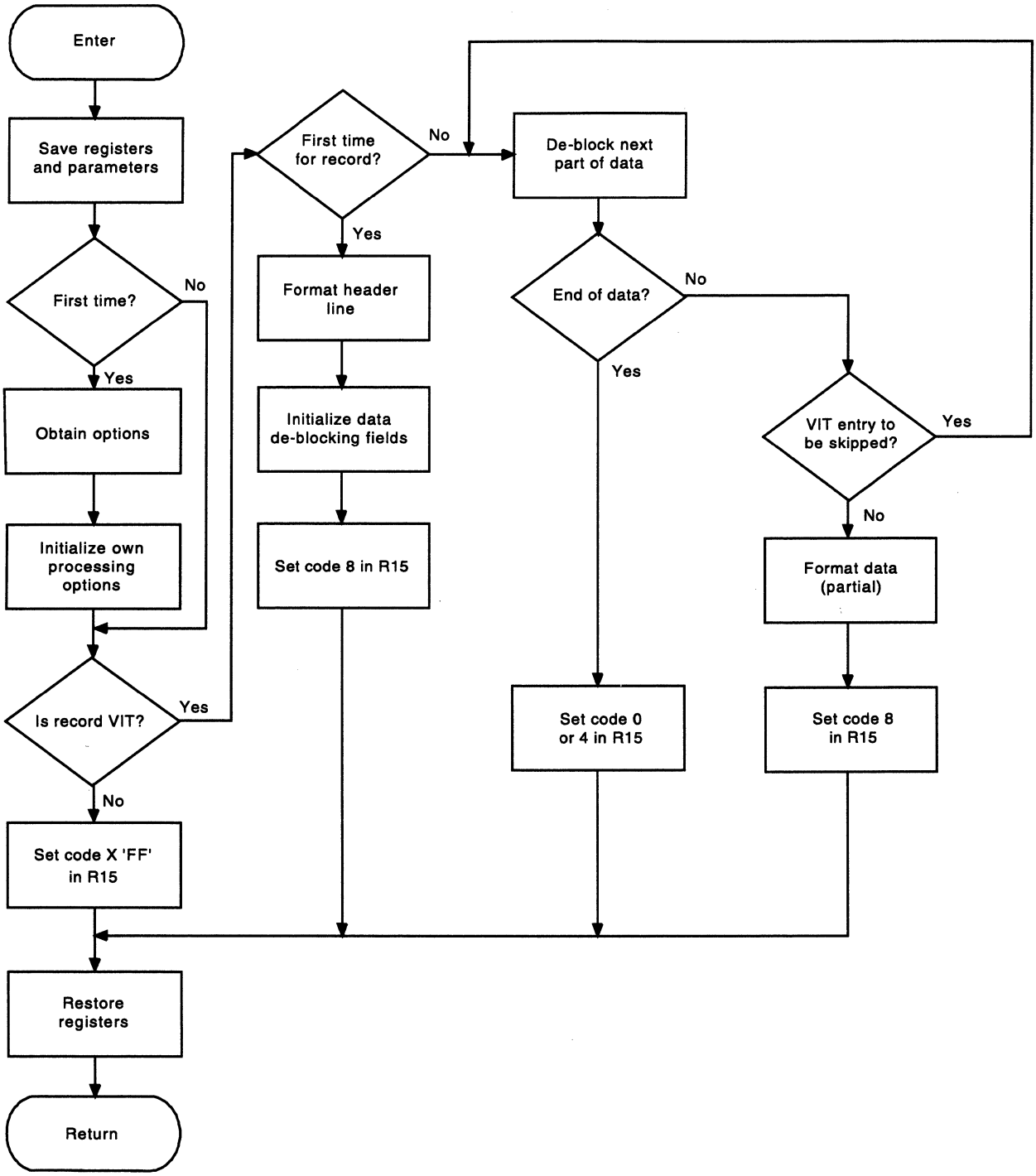


Figure 36. TPRINT User Exit Routine Logic VSE

Using DOSVSDMP

See Table 80 on page 1165 to determine what book describes DOSVSDMP.

Trace Output

In addition to the fields produced by VTAM, Table 19 contains generic fields that might appear in VTAM trace output.

Table 19. MVS *Fields in VTAM Trace Output*

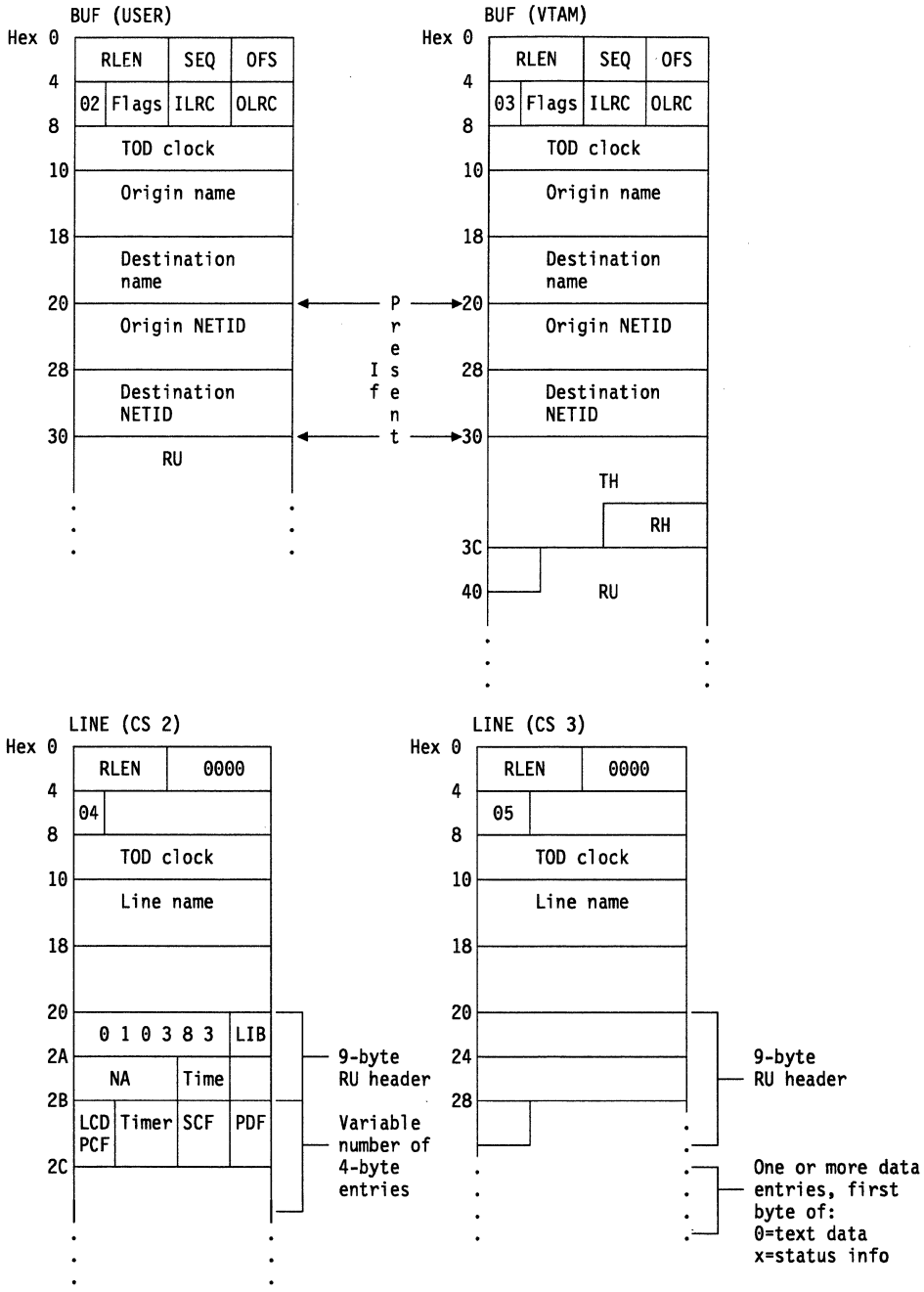
Field Header	Meaning
ASCB *nnnnnnnn*	The address of the ASCB for the address space that created the record.
CPU *nnnn*	The ID of the host processor in which the trace was run (applies only in a multiprocessor configuration).
JOBN *cccccccc*	The name of the job associated with the I/O operation (for an I/O trace).
RNIO	The trace record was created by GTF (with TRACE=RNIO).
USR*xx nnn*	The formatting routine (AMDUSR*xx*). VTAM's trace formatter is AMDUSRFD. *nnn* is the event ID that represents this trace.

In addition to the fields produced by VTAM, Table 20 contains VM-only fields that might appear in VTAM trace output.

Table 20. VM *Fields in VTAM Trace Output*

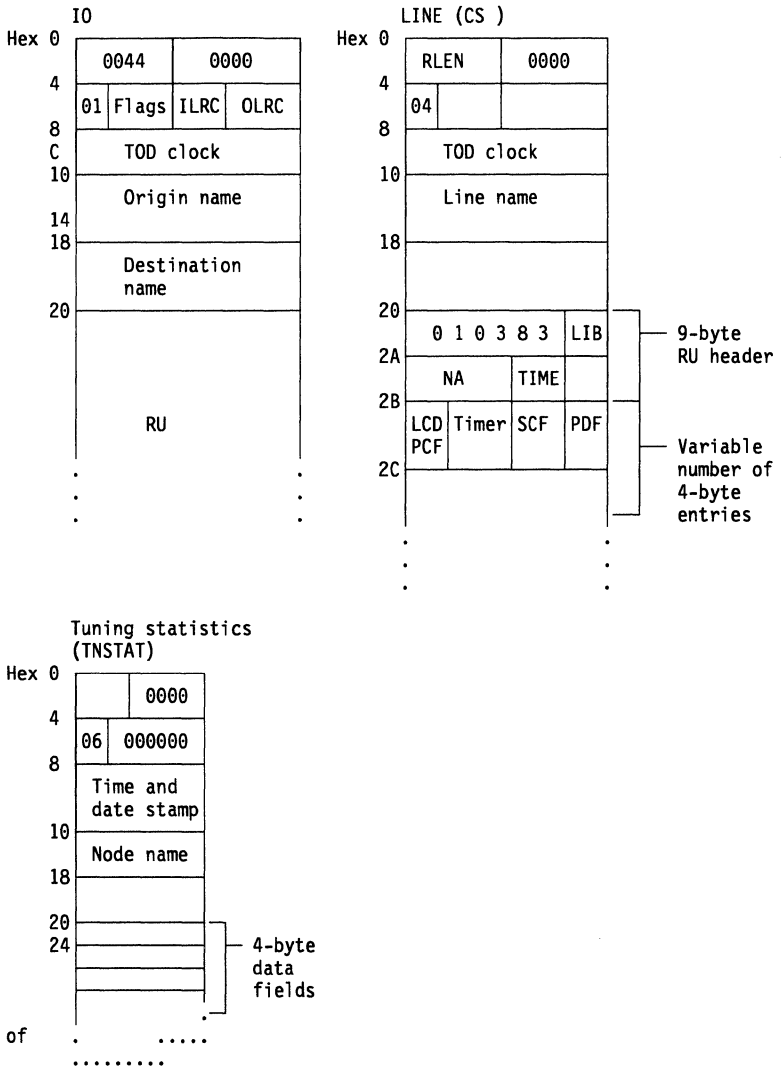
Field Header	Meaning	
Time-of-day clock	Time when trace record was created. To determine what book describes how to interpret time-of-day clock entries, see Table 80 on page 1165.	
Length of trace header and trace data	Indicates the length of this trace record.	
Format routine ID	Identifies formatting module used for this record.	
Event identification	Identifies the event being traced. ID parameter:	
	EFAA	VSCS trace entry
	EFEF	Buffer trace entry (VTAM)
	EFE1	VTAM internal trace entry
	EFE4	Line PIU, generalized PIU, or network controller line trace entry
	EFF0	SMS trace entry
	EFF1	Buffer trace entry (user)
	EFF2	NCP 3705 line, TG trace, or scanner interface trace entry
	E100	RNIO inbound I/O trace entry
	E200	RNIO outbound I/O trace entry

See Figure 37 on page 368 and Figure 38 on page 369 for the format of VTAM trace records.



Legend
RLEN = Total record length
SEQ = Sequence number when running Full Buffer Trace
OFS = Offset of the Network qualifiers (if present) or zero
Flag bit 0 (direction): 1 = Inbound
Flag bit 1: 1 = Confidential text
Flag bit 2: '10'B = First segment of full buffer trace record
Flag bit 3: '01'B = Last segment of full buffer trace record
ILRC = Inbound lost record count
OLRC = Outbound lost record count

Figure 37. VTAM Trace Record Format



Legend
RLEN = Total record length
Flag bit 0 (direction): 1 = Inbound
Flag bit 1: 1 = Confidential text
Flag bit 2: '10'B = First segment of full buffer trace record
Flag bit 3: '01'B = Last segment of full buffer trace record
ILRC = Inbound lost record count
OLRC = Outbound lost record count

Figure 38. VSE-Unique Trace Record Formats VSE

Buffer Contents Trace

The buffer contents trace shows the contents of message buffers in two places: the application program interface (API) and the transmission subsystem component (TSC). When data is sent by an application program (outbound), API is the first component of VTAM to process it, and TSC is the last component of VTAM to process it. When data is received from the network (inbound), TSC is the first component of VTAM to process it, and API is the last. To enable you to distinguish where in VTAM the trace data was recorded, the output specifies:

- Either USER (for data recorded in API) or VTAM (for data recorded in TSC).
- If the recorded data is inbound or outbound.

The API writes user buffer contents trace records while user data is still in the application program's buffers, before it is copied into VTAM buffers. Only user data is recorded.

TSC writes VTAM buffer contents trace records while the data is in VTAM fixed I/O buffers. The PIU — that is, the transmission header (TH), request/response header (RH), and user data — is recorded.

For a list of the resources for which you may request a buffer contents trace see "Resource and Trace Reference" in *VTAM Operation*.

You may request a partial buffer contents trace or a full buffer contents trace. For a partial buffer contents trace, VTAM writes trace records with a maximum size of 256 bytes. The partial buffer contents trace is the default.

For a full buffer contents trace, VTAM records all of the data transmitted in message buffers. The full buffer trace record has a maximum length of 8K bytes (VSE 2K bytes) including trace field headers, transmission headers, request/response headers, and data. If storage is not available to record a trace record in an 8K block (VSE 2K block) , VTAM will record the trace record in 256-byte blocks until the complete trace record is recorded.

Notes:

1. VTAM can start a buffer contents trace only for the resources that it owns. A data host, which does not own any NCPs, cannot start a buffer contents trace for an NCP or any of the NCP's subordinate resources.
2. If you want to trace a session between an LU and an application program, you must start the trace at the host where the application program resides.
3. If the buffer contents trace is active for a specific APPC application, the FMH5 is always traced at the user (API) level.

When to Use the Buffer Contents Trace

The buffer contents trace can help you determine whether a problem is in the host (VTAM or an application program) or in the network. For example, if an application program sends a message to a terminal, and the message is correct in VTAM buffer contents trace output, but the message does not appear correctly at the terminal, then the problem is probably in a system resource other than VTAM or the application program.

The buffer contents trace cannot always be used to distinguish an application program problem from a VTAM problem. However, it can confirm the order in which data is passed between an application program and a logical unit. It can also record all the data passing to and from an application program.

Partial buffer contents tracing is the default. Because of the possible effect on storage and performance, use the full buffer trace only when you need complete buffer data for problem determination. The full buffer contents trace may increase storage use because of the larger size of trace records and the possible need to

increase the size of the trace data set. Performance may be degraded because of the additional time needed to write the complete buffer trace records.

If you do not need to trace user data, use the I/O trace.

Buffer Contents Trace Operation

MVS Make sure that GTF with the TRACE=USR option is active before starting this trace. See "Starting the Generalized Trace Facility (GTF) (MVS)" on page 358.

VM VM V1R2.2 or higher is required to use the buffer contents trace.

Start the buffer contents trace with the MODIFY TRACE command or the TRACE start option.

Specify:	To Trace:
ID=*nodename*	Requests and responses flowing between VTAM and *nodename*.
ID=VTAM	Request and response units (RUs) for SSCP sessions.
ID=ISTPUS	Request and response units (RUs) for sessions between the host physical unit and another physical unit type 4 or 5 (these RUs include ER-ACT, ER-OP, and ER-TEST RUs).

If you use the SCOPE=ALL or EVERY operand when you start the trace, VTAM records messages to and from the specified node and all its valid subnodes.

Note: You cannot use the SCOPE=ALL or EVERY operands with ID=ISTPUS.

For more information on the MODIFY TRACE command, see "MODIFY TRACE Command" in *VTAM Operation*. For more information on the TRACE start option, see "TRACE for Buffer, I/O, NCP Line, SIT, or SMS Traces" in the *VTAM Resource Definition Reference*.

MVS To format and print buffer contents trace output, use ACF/TAP or IPCS.

VM To collect buffer contents trace information, use the TRSOURCE command. To format and print buffer contents trace output, use TRACERED or ACF/TAP.

VSE To print buffer contents trace output, use TPRINT or ACF/TAP.

For more information, see "Formatting and Printing Trace Records" on page 361.

Buffer Contents Trace Output

MVS Figure 39 shows an example of partial buffer contents trace output.
MVS VSE Figure 40 on page 372 shows an example of full buffer contents trace output. VM VSE Output is very similar to the examples shown.

```
BUFF  APPL12  /TERM200 LRC(000,000)   INBOUND
VTAM  TH=40000000 200000C2 0000 000C 00000004 IC000005 0028004 0040 RH=030080
        7DD8E811 D7F1F3C8 C1E3C140 C9E240C3       *'QY.P1THIS DATA IS C*
        D6D4D4C9 D5C740C9 D540D6D5 40E4E2C5       *OMING IN ON THE USE *

BUFF  TERM200 /APPL12  LRC(000,000)  OUTBOUND
USER    C5D5E3D9 40C4C1E3 C140E3D6 40C5C8D6       *ENTER DATA TO ECHO B*
        C5D3D6E6 4B5CF04B D7D9C5E2 E240C5D5       *ELOW.*0.PRESS ENTER *
```

Figure 39. Example of Partial Buffer Contents Trace Output MVS

```
USRFD FEF ASCB 00EBD380           JOBN ECHO42A
          BUFF    NETA.APPL1       /NETA.APPL2          LRC(000,000)    INBOUND     COMPLETE SEGMENT
          VTAM    TH=40000000 00000000 00000001 00000001 1C000014 001B08A7 0003  RH=838000
              GMT-11/02/93 17:32:48.795383    LOC-11/02/93 13:32:48.701175

USRFD FEF ASCB 00EBD380           JOBN ECHO42A
          BUFF    NETA.APPL1       /NETA.APPL2          LRC(000,000)    INBOUND     FIRST    SEGMENT  SEQ(001)
          VTAM    TH=40000000 00000000 00000001 00000001 1C000014 001B08A7 232B  RH=0380C0
                                       C1D7D7D3 F1404040 0000D5C5 E3C14040 40400000  *APPL1   ..NETA    ..*
                                       40D9C5C1 C4E840C6 D6D940C6 C9D9E2E3 40C9D5D7  * READY FOR FIRST INP*
                                       E4E34B40 D3D6C7D6 D540C4C1 E3C1407E 40E2C9D4  *UT. LOGON DATA = SIM*
                                       D3D6C7D6 D5150000 00000000 00000000 00000000  *LOGON.............*
                                       00000000 00000000 00000000 00000000 00000000  *..................*
                                       00000000 00000000 00000000 00000000 00000000  *..................*

:

                                       00000000 00000000 00000000 00000000 00000000  *..................*
                                       00000000 00000000 00000000 00000000 00000000  *..................*
                                       00000000 00000000 000000                      *...........       *
              GMT-11/02/93 17:32:48.802207    LOC-11/02/93 13:32:48.707999

USRFD FEF ASCB 00EBD380           JOBN ECHO42A
          BUFF    NETA.APPL1       /NETA.APPL2          LRC(000,000)    INBOUND     LAST     SEGMENT  SEQ(002)
          VTAM                             00000000 00000000 00000000 00000000 00000000  *..................*
                                           00000000 00000000 00000000 00000000 00000000  *..................*
                                           00000000 00000000 00000000 00000000 00000000  *..................*
                                           00000000 00000000 00000000 00000000 00000000  *..................*

:

                                           00000000 00000000 00000000 00000000 00000000  *..................*
                                           00000000 00000000 00000000 00000000 00000000  *..................*
                                           00000000 00000000 00000000 00000000 00000000  *..................*
                                           00000000 00000000 00000000 00000000 00000000  *..................*
                                           00000000 00000000 00000000 00000000 00000000  *..................*
                                           00000000 00000000 00                          *.........         *
              GMT-11/02/93 17:32:48.802423    LOC-11/02/93 13:32:48.708215
```

Figure 40. Example of Full Buffer Contents Trace Output MVS VSE

Table 21 explains the trace fields. In addition to these fields, operating-system-dependent fields may appear. For a description of these fields, see Table 19 on page 367 and Table 20 on page 367.

Table 21 (Page 1 of 2). Fields in the Buffer Contents Trace

Field Header	Meaning
BUFF *destname/origname*	Is the destination (*destname*) and origin (*origname*) node name. In Figure 39 on page 371 the destination is APPL12 and the origin is TERM200.
LRC(*xxx,yyy*)	The number of records lost since the last trace record was written because the trace facility could not get a VTAM buffer. *xxx* is the destination's lost record count, and *yyy* is the source's lost record count.
INBOUND or OUTBOUND	The direction of the traced data (inbound or outbound) with respect to this host subarea. By use of this field and the PIU sequence number in the TH, requests and corresponding responses can be matched.
position SEGMENT (full buffer trace only)	Indicates whether this trace record is FIRST, MIDDLE, or LAST in a series of trace records generated for one trace invocation. If only one trace record is needed, the value is COMPLETE. The segment indicator field appears only when full buffer contents tracing is in effect.

Table 21 (Page 2 of 2). Fields in the Buffer Contents Trace

Field Header	Meaning
SEQ(*xxx*) (full buffer trace only)	A sequence number indicating the sequence in which trace records were generated. The sequence number appears only when a series of trace records is generated for a single trace invocation. The sequence number does not appear when one trace record shows a complete buffer. The sequence number starts at 1, and upon reaching 255, wraps to 0. A gap in sequence numbers could indicate lost trace records. The sequence number field applies only when full buffer contents tracing is in effect.
VTAM or USER	Indicates where the message buffers were traced. VTAM means the buffers were traced in TSC (in which case the TH and the RH are included in the trace record). USER means the buffers were traced in API (in which case the TH and the RH are not included).
TH	The transmission header portion of the path information unit (PIU).
RH	The request/response header portion of the PIU.

Note: The rest of the trace record shows the contents of the buffer as displayed in Figure 39 on page 371.

Notes:

1. If the buffer trace information is out of sequence your trace may have wrapped.
2. User entries are not printed by ACF/TAP.
3. Confidential data is *not* recorded in trace records. When the trace facility detects confidential data (CONFTXT=YES is specified on the application program's NIB macroinstruction), the user data is replaced with the marker in the trace output.

   MVS VM The marker is "CONFIDENTIAL AND SUPPRESSED."
   VSE The marker is "CONFIDENTIAL DATA."

   MVS If you are using the VTAM encryption facility, data on a cryptographic session is handled in the same way as confidential data.

I/O Trace

The I/O trace shows requests and responses that flow between VTAM and network nodes. You can trace I/O activity for any of the following types of nodes:

- Application program
- Physical unit
- Logical unit
- SNA cluster controller
- NCP
- SSCP
- Host physical unit
- Host as an intermediate routing node
- Channel attachment major node
- Cross-domain resource

- Cross-domain resource manager.

The maximum I/O trace record length is 36 bytes.

Note: If you want to trace a session between an LU and an application program, you must start the trace at the host where the application program resides.

When to Use the I/O Trace

Use the I/O trace to record the order that PIUs flow between network nodes and VTAM. For example, you might use this trace to determine whether an application program receives all the responses that it should and whether VTAM forwards all the requests issued by the application program.

MVS If you are using the AnyNet* SNA over TCP/IP program product, the I/O trace provides tracing of all MPTN formats that are sent to and received from the TCP/IP network. See "AnyNet I/O Trace Format for SNA over TCP/IP (MVS)" on page 376 and *VTAM AnyNet Feature for V4R2: Guide to SNA over TCP/IP* for additional information.

The content of the I/O trace is similar to that of the buffer trace, except that the I/O trace collects less user data. If you do not need this user data, you can save trace file space by using the I/O trace.

I/O Trace Operation

MVS Before starting the I/O trace, make sure that GTF with the TRACE=RNIO option is active. See "Starting the Generalized Trace Facility (GTF) (MVS)" on page 358.

Start the I/O trace with the MODIFY TRACE command or the TRACE start option.

Specify:	To Trace:
ID=*nodename*	Requests and responses flowing between VTAM and *nodename.*
ID=VTAM	Request and response units (RUs) for SSCP sessions.
ID=ISTPUS	Request and response units (RUs) for sessions between the host physical unit and another physical unit type 4 or 5 (these RUs include ER-ACT, ER-OP, and ER-TEST RUs).
ID=ISTIRN	Request and response units (RUs) that flow through this host while this host is acting as an intermediate routing node.

Note:

1. If you use the SCOPE=ALL or EVERY operand when you start the trace, the trace contains I/O activity for the specified node and all its valid subnodes. You **must** specify SCOPE=ALL when tracing a channel-attachment major node or when tracing an APPN PU. You **cannot** use the SCOPE=ALL or EVERY operands with ID=ISTPUS or ID=ISTIRN.
2. You may trace a link in a channel-attachment major node, but not a link station.

See "MODIFY TRACE Command" in *VTAM Operation* for information on the MODIFY command. See "TRACE for Buffer, I/O, NCP Line, SIT, or SMS Traces" in the *VTAM Resource Definition Reference* for more information on the TRACE start option.

MVS If you are using the AnyNet SNA over TCP/IP program product, refer to the section on diagnosing SNA over TCP/IP problems in *VTAM AnyNet Feature for V4R2: Guide to SNA over TCP/IP* for information about activating the I/O trace.

MVS To format and print I/O trace output, use ACF/TAP or IPCS.

VM To collect I/O trace information, use the TRSOURCE command. To format and print I/O trace output, use TRACERED or ACF/TAP.

VSE To print I/O trace records, use TPRINT or ACF/TAP.

For more information, see "Formatting and Printing Trace Records" on page 361.

I/O Trace Output (MVS) (VM)

Figure 41 shows an example of I/O trace output.

The trace fields are explained after the figure. In addition to the fields described here, additional operating system-dependent fields may appear. For a description of these fields, see Table 19 on page 367 and Table 20 on page 367.

```
OUT..... 40000000 00000000 00000001 00000001 1D000092 0015001B 00256B80 00310102 02714020
GMT-11/02/xx 18:50:31.827470    LOC-11/02/xx 14:50:31.733262

IN...... 40000000 00000000 00000001 00000001 1D000015 0092001B 0004EB80 0031
GMT-11/02/xx 18:50:31.830107    LOC-11/02/xx 14:50:31.735899
```

Figure 41. Example of I/O Trace Output MVS VM

MVS VM The following fields appear in the I/O trace:

Field Header	Meaning
OUT or IN	The direction of the data shown in the trace record: OUT for outbound to the specified node and IN for inbound to VTAM. When ISTIRN is specified, each RU is traced twice — once when it leaves VTAM (OUT) and once when it enters VTAM (IN).

The rest of the trace record is some or all of the path information unit (PIU). It consists of a 26-byte transmission header (TH), a request/response header (RH), and as much of the request/response unit (RU) as will fit. The data length field in the TH indicates the length of the RH and RU in bytes. The RU may contain sense data.

I/O Trace Output (VSE)

Figure 42 shows an example of I/O trace output for VSE. The trace fields are explained after the figure.

```
IO  TERM12  /       87.153/10:31:02.13          OUTBOUND
TH=400000000000000000000000C0000000C1D000039000500 0D0028  RH=6B8000  RU=31010202714020

IO          /TERM12  87.153/10:31:02.15           INBOUND
TH=400000000000000000000000C0000000C1D000005003900 0D0004  RH=EB8000  RU=31
```

Figure 42. Example of I/O Trace Output VSE

VSE The following fields appear in the I/O trace:

Field Header	Meaning
IO *destname/origname*	Is the destination (*destname*) or origin (*origname*) node name. Usually, only one node name is resolved by the I/O trace. In Figure 42, the node name is TERM12 in both records.
yy.jjj/hh:mm:ss.hh	Is the year (yy), the Julian date (jjj), and the time in hundredths of a second (hh:mm:ss.hh) that the trace record was written. This field is "87.153/10:31:02.13" in the first trace record in Figure 42.
LRC(*xxx,yyy*)	Is the number of records lost since the last trace record was written because the trace facility could not obtain VTAM buffers. *xxx* is the destination's lost record count, and *yyy* is the source's lost record count. **Note:** This field is not present in the trace record unless records are lost.
INBOUND or OUTBOUND	The direction of the traced data (inbound or outbound) with respect to this host subarea. With this field and the PIU sequence number in the TH, requests and corresponding responses can be matched.
TH	Is the 26-byte transmission header portion of the path information unit (PIU). The data length field in the TH indicates the length of the RH and RU in bytes.
RH	Is the 3-byte request/response header portion of the PIU.
RU	Up to ten bytes of the request/response unit, which may contain sense data.

AnyNet I/O Trace Format for SNA over TCP/IP (MVS)

The format of the AnyNet trace header is described in Table 22 on page 377. Use this information to verify the MPTN formats that flow across the IP network. For additional information, refer to the *VTAM AnyNet Feature for V4R2: Guide to SNA over TCP/IP.*

Notes:

1. The AnyNet trace header is always complete.
2. The SNA request/response unit (RU) may be truncated.
3. The trace header identifier is in the same position as the identifier for the buffer contents trace.

Table 22. Format of the AnyNet Trace Header for SNA over TCP/IP

Byte	Field Length	Field Description
0	2	Total record length = *n*
2	2	Length of the header (always 32 bytes)
4	1	Trace type identifier (X'0D')
5	1	Trace flags Bit Meaning x... .... Direction flag **1** Inbound **0** Outbound .xxx xxxx Not used
6	2	Not used
8	4	Local internet (IP) address
12	4	Remote internet (IP) address
16	2	Local port number
18	2	Remote port number
20	8	Timestamp
28	4	Not used
32	*n*–32	MPTN format and SNA request/response unit (RU)

SMS (Buffer Use) Trace

The storage management services (SMS) trace records contain information on the use and availability of VTAM buffer pools. SMS trace records are written after a predetermined number of requests occur for VTAM buffers. An IBM-supplied threshold causes a trace record to be written after every 1000 requests.

Notes:

1. If the DISPLAY BFRUSE command is issued while this trace is running, the fields MAX TOTAL, MAX USED, and TIMES EXP in the output for DISPLAY BFRUSE reflect buffer usage only since the last trace record was written, because the SMS trace resets these fields.
2. The SMS trace is **not** the same thing as the VTAM internal trace with the SMS option specified. The SMS trace **is** similar to the DISPLAY BFRUSE command. The SMS trace displays in trace output the same information that the DISPLAY BFRUSE command displays on the screen.

When to Use the SMS Trace

Use the SMS trace during VTAM installation to evaluate VTAM use of buffer pools, to help estimate how many buffers VTAM needs for normal operation, and, with dynamic buffering, to limit buffer pool expansions to peak use periods. You can use the SMS trace in conjunction with tuning statistics.

SMS Trace Operation

Start the SMS trace with the MODIFY TRACE command, or use the TRACE start option with TYPE=SMS and ID=VTAMBUF specified. For more information about the MODIFY TRACE command, see "MODIFY TRACE Command" in *VTAM Operation*. For more information about the TRACE start option, see "TRACE for Buffer, I/O, NCP Line, SIT, or SMS Traces" in the *VTAM Resource Definition Reference*.

MVS Make sure that GTF with the TRACE=USR option is active before starting this trace. See "Starting the Generalized Trace Facility (GTF) (MVS)" on page 358. To format and print the data recorded by GTF, use IPCS and set USR(FF0) or USR(CL02) on the GTFTRACE option.

VM To collect SMS trace information, use the TRSOURCE command. To format and print SMS trace output, use the ACF/TAP or TRACERED command.

VSE Make sure that SDAID is active, and TRACE VTAMBU is specified. The trace output can be printed with the DOSVSDMP utility program. See Table 80 on page 1165 to determine what book describes SDAID and DOSVSDMP.

For more information on printing trace output, see "Formatting and Printing Trace Records" on page 361.

SMS Trace Record Output (MVS) (VM)

The SMS trace record format is different for MVS and VM. Figure 43 and Figure 44 show examples of SMS trace output. The trace fields are explained in Table 23 on page 379. In addition to the fields described here, other operating-system-dependent fields may appear. These fields are described in Table 19 on page 367 and Table 20 on page 367 .

```
VTAM BUFFERS       MAXU      MAXQ      AVNO      TEXP      MBUF      TOTL
      IO        00000039  00000000  0000016E  00000000  00000190  00000190
      BS        00000000  00000000  00000261  00000000  00000261  00000261
      LP        00000007  00000000  0000003D  00000000  00000040  00000040
      XD        00000000  00000000  0000000A  00000000  0000000A  0000000A
      LF        0000000D  00000000  0000006B  00000000  00000078  00000078
      CR        00000016  00000000  000000B7  00000000  000000C8  000000C8

VTAM BUFFERS       MAXU      MAXQ      AVNO      TEXP      MBUF      TOTL
      SF        00000008  00000000  00000038  00000000  00000040  00000040
      SP        00000001  00000000  00000032  00000000  00000032  00000032
      AP        00000000  00000000  00000038  00000000  00000038  00000038
VTAM CSAUSE     TOTAL 0030015B MAX 7FFFFFFF %FREE QUEUE  00000000
```

Figure 43. Example of SMS Trace Output MVS *. Two separate records will be printed, and they may be separated by another trace entry.*

```
VTAM BUFFERS       MAXU      MAXQ      AVNO      TEXP      MBUF      TOTL
      IO        00000005  00000000  00000129  00000000  0000012C  0000012C
      BS        00000002  00000000  0000001A  00000000  0000001C  0000001C
      LP        00000006  00000000  00000025  00000000  00000028  00000028
      XD        00000001  00000000  00000006  00000000  00000006  00000006
      LF        00000002  00000000  00000016  00000000  00000018  00000018

VTAM BUFFERS       MAXU      MAXQ      AVNO      TEXP      MBUF      TOTL
      CR        0000000A  00000000  0000001F  00000000  00000028  00000028
      UE        00000000  00000000  00000000  00000000  00000000  00000000
      SF        00000004  00000000  0000001C  00000000  00000020  00000020
      SP        00000000  00000000  00000002  00000000  00000002  00000002
VTAM CSAUSE     TOTAL 00000000 MAX 00000000 %FREE QUEUE  00000000
```

Figure 44. Example of SMS Trace Output VM *. Two separate records will be printed, and they may be separated by another trace entry.*

Table 23. MVS VM *Fields in the SMS Trace*

Field Header	Meaning
Pool ID	Is the first field in each record and identifies the buffer pool. Pool IDs and their corresponding buffer pool names are: AP Application program pageable pool (APBUF) MVS BS Boundary session block pool (BSBUF) CR Copied RPL pool (CRPLBUF) IO Fixed I/O pool (IOBUF) LF Large fixed pool (LFBUF) LP Large pageable pool (LPBUF) SF Small fixed pool (SFBUF) SP Small pageable pool (SPBUF) UE User exit control block (UECB) (obsolete) XD XID pool (XDBUF).
MAXU	The maximum number of buffers in the pool that are in use at any time since the last trace record is written.
MAXQ	The maximum number of requests for buffers that are queued waiting for storage at any time since the last trace record is written.
AVNO	The number of available buffers (those not in use at the time the trace record is written).
TEXP	The number of times the buffer pool is expanded since the last trace record is written.
MBUF	The maximum number of buffers that are in the pool at any time since the last trace record is written. This includes both used and unused buffers.
TOTL	The total number of buffers that are in the pool at the time this record is written. This includes both used and unused buffers.
TOTAL	The amount of CSA storage in use by VTAM at the time this record is written.
MAX	The largest amount of CSA storage used by VTAM since the last SMS buffer trace.
%FREE QUEUE	The amount of CSA storage allocated to VTAM that is waiting to be freed.

SMS Trace Record Output (VSE)

Table 24 on page 380 describes the fields that appear in the SMS trace.

Table 24. VSE Fields in the SMS Trace

Field Header	Meaning
Pool ID	Is the first field in each record. It identifies the buffer pool. Pool IDs and their corresponding buffer pool names are: BS Boundary session block pool (BSBUF) CR Copied RPL pool (CRPLBUF) IO Fixed I/O pool (IOBUF) LF Large fixed pool (LFBUF) LP Large pageable pool (LPBUF) SF Small fixed pool (SFBUF) SP Small pageable pool (SPBUF) XD XID pool (XDBUF). **Note:** Sometimes pool names are not listed. The pool names are not available when VTAM is running disabled.
IN USE	Indicates the number of buffers in the pool that are in use at any time during the time covered by the snapshot dump.
MAX ALLOC	Indicates the maximum number of buffers in the pool that are in use at any time during the time covered by the snapshot dump.
MAX WAIT	Indicates the maximum number of requests for buffers that are queued waiting for storage at any time during the time covered by the snapshot dump.
EXPAND	Indicates the number of times the buffer pool is expanded during the time covered by the snapshot dump.
MAX AVAIL	Indicates the maximum number of buffers that are in the pool at any time during the time covered by the snapshot dump.
CUR AVAIL	Indicates the number of buffers that are in the pool at the time the snapshot dump is taken.

TGET/TPUT Trace for TSO/VTAM (MVS)

The TGET/TPUT trace for TSO/VTAM writes a GTF trace record for each inbound and outbound message that uses the TGET/TPUT/TPG interface (SVC 93) between a TSO command processor and the VTIOC component of TSO/VTAM. Outbound messages are traced before being placed in the VTIOC queue manager output buffer. Inbound messages are traced before the data is sent to the TSO command processor.

Note: The TGET/TPUT trace does not trace address space ID TPUTs.

When to Use the TGET/TPUT Trace

Use this trace if the failure is restricted to TSO sessions. This trace can help you determine whether TSO/VTAM or your TSO command processor is causing the problem. For example, if outbound data is correct in the TGET/TPUT trace output, but incorrect in the buffer trace output, the problem is probably in TSO/VTAM or VTAM. Use the following tables as guidelines to determine where the error is occurring:

Table 25. Location of TPUT (Outbound) Error

Direction of Data	If TPUT Trace Data Is:	And Buffer Trace Data Is:	Then Possible Error Is In:
Outbound	Correct	Incorrect	• VTAM • TSO/VTAM • TPUT option • User edit exits
Outbound	Incorrect	Trace not required	TSO or the command processor
Outbound	Correct	Correct	Network

Table 26. Location of TGET (Inbound) Error

Direction of Data	If TGET Trace Data Is:	And Buffer Trace Data Is:	Then Possible Error Is In:
Inbound	Incorrect	Correct	• VTAM • TSO/VTAM • TGET option • User edit exits
Inbound	Correct	Trace not required	TSO or the command processor
Inbound	Incorrect	Incorrect	Network

TGET/TPUT Trace Operation

Start the TGET/TPUT trace with the MODIFY TRACE command and specify TYPE=TSO.

The trace output is a record of inbound and outbound messages for the specified TSO user ID.

Make sure that GTF with the TRACE=USR option is active before starting this trace. See "Starting the Generalized Trace Facility (GTF) (MVS)" on page 358.

To print these trace records, use IPCS and specify either USR(FE2) or USR(APTH) on the GTFTRACE option. For more information on printing trace output, see "Formatting and Printing Trace Records" on page 361.

TGET/TPUT Trace Output

The trace records created by the TGET/TPUT trace have a 12-byte GTF header and a 52-byte trace header followed by the data portion of the RU in unformatted hexadecimal. The entire RU is traced, but will span several trace entries if it is longer than 228 bytes.

Figure 45 on page 382 shows an example of TGET/TPUT trace output. The trace fields are explained after the figure.


```
1IPCS PRINT LOG FOR USER USER1                                          1  16:15:06 09/03/xx
+-------------------------------------------------------------------------------------------
0   **** GTFTRACE DISPLAY OPTIONS IN EFFECT ****
  USR=SEL
0 **** GTF DATA COLLECTION OPTIONS IN EFFECT: ****
  All GTRACE events requested
  RNIO events traced
0             **** GTF TRACING ENVIRONMENT ****
      Release: SP4.2.2   FMID: JBB4422   System name: MVS41D25
      CPU Model: 3090  Version: FF  Serial no. 373247
-HEXFORMAT AID FF FID 00 EID  EFE2
  +0000  00EBE380  E4E2C5D9  F1404040  E3E2D6D6  E4E30300  C9D5C9E3  E4E2C5D9  F1404040  | ..T.USER1   TSOOUT..INITUSER1    |
  +0020  0DD5C5E3  C14BC1F5  F0C9F0F7  F2F14040  40400000  C1114040  1D603CC1  50401D60  | .NETA.A50I0721    ..A.  .-.A& .- |
  +0040  3CC26040  1D603CC2  7E40C9C9  C9C94040  C2C2C2C2  40404040  D4404040  D43CC3F0  | .B- .-.B= IIII  BBBB    M   M.C0 |
  +0060  401D603C  C44F40C9  C9404040  40C24040  C2404040  D4D440D4  D43CC540  401D603C  |  .-.D| II    B  B   MM MM.E  .-.  |
  +0080  C55F40C9  C9404040  40C240C2  40404040  D440D440  D43CC650  401D603C  C66F40C9  | E¬ II    B B    M M M.F& .-.F? I |
  +00A0  C9404040  40C24040  C2404040  D4404040  D43CC760  401D603C  C77E40C9  C9C9C940  | I    B  B   M   M.G- .-.G= IIII  |
  +00C0  40C2C2C2  C2404040  40D44040  40D43CC8  F0401D60  3C4A4040  1D603C4B  50401D60  |  BBBB    M   M.H0 .-.¢  .-..& .-  |
  +00E0  3C4BF240  1DE8C9E2  D7C661D7  C4C63C4C  60401D60  3C4DF040  1D603C4F  40401DE8  | ..2 .YISPF/PDF.<- .-.(0 .-.|  .Y  |
  +0100  3C4FD340  D3898385  95A28584                                                    | .|L Licensed                        |
              GMT-09/03/xx 19:29:49.847841    LOC-09/03/xx 15:29:49.753633
0HEXFORMAT AID FF FID 00 EID  EFE2
  +0000  00EBE380  E4E2C5D9  F1404040  E3E2D6D6  E4E30300  40D4C9C4  E4E2C5D9  F1404040  | ..T.USER1   TSOOUT.. MIDUSER1    |
  +0020  0DD5C5E3  C14BC1F5  F0C9F0F7  F2F14040  40400000  40D481A3  85998981  93A24060  | .NETA.A50I0721    .. Materials - |
  +0040  40D79996  978599A3  A8409686  40C9C2D4  3C505040  1D603CD1  60401DE8  3CD1F340  |  Property of IBM.&& .-.J- .Y.J3  |
  +0060  F5F6F8F4  60F1F2F3  404DC35D  40C39697  A8998987  88A340C9  C2D440C3  9699974B  | 5684-123 (C) Copyright IBM Corp. |
  +0080  40F1F9F8  F06B40F1  F9F9F04B  3CD2F040  1DE83CD3  C340C193  93409989  8788A3A2  |  1980, 1990..K0 .Y.LC All rights |
  +00A0  409985A2  8599A585  844B3CD4  40401DE8  3CD4D340  E4E240C7  96A58599  95948595  |  reserved..M  .Y.ML US Governmen |
  +00C0  A340E4A2  8599A240  D985A2A3  998983A3  858440D9  898788A3  A240603C  D550401D  | t Users Restricted Rights -.N& . |
  +00E0  E83CD5E3  40E4A285  6B4084A4  97938983  81A38996  95409699  408489A2  839396A2  | Y.NT Use, duplication or disclos |
  +0100  A4998540  9985A2A3  998983A3                                                    | ure restrict                       |
              GMT-09/03/xx 19:29:49.847887    LOC-09/03/xx 15:29:49.753679
0HEXFORMAT AID FF FID 00 EID  EFE2
  +0000  00EBE380  E4E2C5D9  F1404040  E3E2D6D6  E4E30300  40D4C9C4  E4E2C5D9  F1404040  | ..T.USER1   TSOOUT.. MIDUSER1    |
  +0020  0DD5C5E3  C14BC1F5  F0C9F0F7  F2F14040  40400000  85843CD6  60401DE8  3CD6F340  | .NETA.A50I0721    ..ed.O- .Y.O3  |
  +0040  82A840C7  E2C140C1  C4D740E2  83888584  A4938540  C39695A3  998183A3  40A689A3  | by GSA ADP Schedule Contract wit |
  +0060  8840C9C2  D440C396  9997483C  D7F0401D  603CD940  401D603C  5A50401D  603C5A6E  | h IBM Corp..P0 .-.R  .-.!& .-.!> |
  +0080  401DE8C5  D5E3C5D9  1D60A396  40839695  A38995A4  853C5B60  401D6040  C6F17EC8  |  .YENTER.-to continue.$- .- F1=H |
  +00A0  C5D3D73C  5B6F40C6  F27EE2D7  D3C9E33C  5B7C40C6  F37EC5D5  C43C5CC9  40C6F47E  | ELP.$? F2=SPLIT.$@ F3=END.*I F4= |
  +00C0  D9C5E3E4  D9D54040  4040C6F5  7ED9C6C9  D5C43C5C  E340C6F6  7ED9C3C8  C1D5C7C5  | RETURN    F5=RFIND.*T F6=RCHANGE |
  +00E0  3C5CF240  C6F77EE4  D73C5C7F  40C6F87E  C4D6E6D5  3C5D4C40  C6F97EE2  E6C1D73C  | .*2 F7=UP.*" F8=DOWN.)< F9=SWAP. |
  +0100  5DD840C6  F1F07ED3  C5C6E33C                                                    | )Q F10=LEFT.                       |
              GMT-09/03/xx 19:29:49.892763    LOC-09/03/xx 15:29:49.798555
0HEXFORMAT AID FF FID 00 EID  EFE2
  +0000  00EBE380  E4E2C5D9  F1404040  E3E2D6D6  E4E30300  D3C1E2E3  E4E2C5D9  F1404040  | ..T.USER1   TSOOUT..LASTUSER1    |
  +0020  0DD5C5E3  C14BC1F5  F0C9F0F7  F2F14040  40400000  5DE540C6  F1F17ED9  C9C7C8E3  | .NETA.A50I0721    ..)V F11=RIGHT |
  +0040  40404040  C6F1F27E  D9C5E3D9  C9C5E5C5  40401140  4013                          |     F12=RETRIEVE  .  .             |
              GMT-09/03/xx 19:29:49.893415    LOC-09/03/xx 15:29:49.799207
0HEXFORMAT AID FF FID 00 EID  EFE2
  +0000  00EBE380  E4E2C5D9  F1404040  E3E2D6C9  D5408100  D3C1E2E3  E4E2C5D9  F1404040  | ..T.USER1   TSOIN a.LASTUSER1    |
  +0020  0DD5C5E3  C14BC1F5  F0C9F0F7  F2F14040  40400000  7D4040                        | .NETA.A50I0721    ..'              |
              GMT-09/03/xx 19:29:51.997746    LOC-09/03/xx 15:29:51.903538
1IPCS PRINT LOG FOR USER USER1                                          2  16:15:43 09/03/xx
+-------------------------------------------------------------------------------------------
```

Figure 45. Example of TGET/TPUT Trace Output

The following fields appear in the TGET/TPUT trace. The first two bytes in each row show the hex offset in storage. The data follows after that.

Byte (hex)	Meaning
00–03	ASCB address
04–0B	Job name
0C–0E	C'TSO'
0F–11	C'IN' for inbound data (TGET); C'OUT' for outbound data (TPUT)
12	TGET/TPUT option flags (See the TGET/TPUT option flags entry in Table 80 on page 1165 to determine what book describes these bit definitions.)
13	TGET: return code (See Table 80 on page 1165 to determine what book describes TGET return codes.) TPUT: X'00' EDIT, ASID, FULLSCREEN, or CONTROL options X'01' NOEDIT option X'02' TPG macro issued. X'03' NOEDIT option specified and TGP macro issued.

Byte (hex)	Meaning
14–17	C'INIT' for the first 228-byte section of a PIU; C'MID' for the middle sections of a PIU; C'LAST' for the last section of a PIU
18–1F	TSO user ID
20	Length of network-qualified name
21–31	Network-qualified name
32–33	Zero
34	Start of user data.

Traces Provided by NCP

NCP provides several kinds of traces to record the flow of network events. Each trace occurs at a different point in the network (see Figure 33 on page 348). This allows you to follow an RU through the network and determine where in the network the RU is incorrect. (The RU could be out of sequence or lost, the data in the RU could have been changed, and so forth.)

The NCP traces are:

- "Generalized PIU Trace"
- "Line Trace" on page 385
- "Network Controller Line Trace (3710 Only)" on page 389
- "Scanner Interface Trace (3720, 3725, and 3745 Only)" on page 390
- "Transmission Group Trace" on page 391.

To determine what book contains more information on NCP service aids, see Table 80 on page 1165.

Generalized PIU Trace

The generalized PIU trace collects PIU trace data collected by the NCP. The resulting trace output shows the flow of PIUs exchanged between the NCP and its attached resources. This trace is hierarchical when started for a physical unit or a line. That is, logical units associated with the physical unit are automatically traced when traffic flows to them. Likewise, when the generalized PIU trace is started for a line, physical units and logical units associated with the line are automatically traced when traffic flows to them. When the generalized PIU trace is stopped for a physical unit, the trace is reset for all logical units associated with the physical unit, regardless of how the generalized PIU trace was started.

When to Use the Generalized PIU Trace

Use the generalized PIU trace to trace PIU data at the NCP level and to determine whether the NCP has received or sent PIU data.

Note: VTAM can start a generalized PIU trace only for the resources that it owns. A data host, which does not own any NCPs, cannot start a generalized PIU trace for an NCP or any of the NCP's subordinate resources.

Generalized PIU Trace Operation

Start the generalized PIU trace with the MODIFY TRACE command. For more information on the MODIFY TRACE command for the generalized PIU trace, see "MODIFY TRACE Command" in *VTAM Operation*.

MVS Make sure that GTF with the TRACE=USR option is active before starting this trace. See "Starting the Generalized Trace Facility (GTF) (MVS)" on page 358.

VM To collect generalized PIU trace information, use the TRSOURCE command. To format and print generalized trace output, use ACF/TAP.

VSE Use ACF/TAP with INPUT=GPT to format and print the trace output.

For more information on printing trace output, see "Formatting and Printing Trace Records" on page 361.

Line Trace

The line trace, a joint function of VTAM and the NCP, records the status of a line each time the NCP receives data from or sends data to that line. Although the trace is controlled by VTAM, the information in the trace records is collected by the NCP. The NCP sends the data to VTAM in a PIU. A trace type indicator in the PIU indicates whether the trace is a byte line trace (type 2 scanner) or a block line trace (type 3 scanner).

Note: If the data is not from a type 1, 2, or 3 scanner, VTAM will not process the data. You must use ACF/TAP.

The line trace collects the operating parameters of a line each time a level 2 interruption occurs on that line.

- For a 3705 Communications Controller with a type 2 communication scanner, a level 2 interruption occurs each time a byte of data is sent or received across the line.
- For a 3705 Communications Controller with a type 3 communication scanner, a level 2 interruption occurs each time an NCP buffer is filled and the buffer data is sent or received across the line.
- For a 3720, 3725, or 3745 Communication Controller, a level 2 interruption occurs each time a message (an entire PIU) is sent or received.

You can use the line trace only for lines attached to a communication controller and operating in network control mode.

For each 3705 Communications Controller, as many as eight line traces can be active at a time. For each 3720, 3725, or 3745 Communication Controller, a combination of eight line or scanner interface traces can be active at one time.

The number of active line traces to be allowed is specified during NCP generation (default = 2). In a cross-domain network in which the communication controller is connected to more than one host processor, the number of active traces allowed is distributed among the connected host processors on a first-come, first-served basis.

Note: As the number of active line traces increases, the system becomes less efficient.

When to Use the Line Trace

You might use this trace if you suspect a problem with a device attached to a communication controller. If data appears correctly in the line trace but the terminal or printer does not react appropriately, the device itself is probably causing the failure.

Before using a line trace, you should use buffer and I/O traces to verify that the problem is not in VTAM or an application program. You may also want to use the scanner interface trace (for 3720, 3725, and 3745 Communication Controllers only), which traces data after it has been processed by the NCP and before it leaves the communication controller. Therefore, it can help determine whether the problem is in the NCP or in the line.

Note: VTAM can start a line trace only for the resources that it owns. A data host, which does not own any NCPs, cannot start a line trace for an NCP or any of the NCP's subordinate resources.

You might want to use a transmission group (TG) trace instead of or in addition to this trace. If there is more than one active line in a transmission group, and you do not know which line is causing the problem, use the transmission group trace. The transmission group trace also shows more data than the line trace.

A line trace can be active for any line in the transmission group. However, a line trace and a transmission group trace cannot be active for the same line, at the same time.

Line Trace Operation

Start the line trace with the MODIFY TRACE command or the TRACE start option. In either case, specify TYPE=LINE. For more information on the MODIFY TRACE command, see "MODIFY TRACE Command" in *VTAM Operation*. For more information on the TRACE start option, see "TRACE for Buffer, I/O, NCP Line, SIT, or SMS Traces" in the *VTAM Resource Definition Reference*.

MVS Make sure that GTF with the TRACE=USR option is active before starting this trace. See "Starting the Generalized Trace Facility (GTF) (MVS)" on page 358.

To format and print line trace data for a 3705, 3720, 3725, or 3745, use ACF/TAP and specify INPUT=LINE.

VM To format and print line trace data for a 3705, use ACF/TAP or TRACERED. For ACF/TAP, specify INPUT=LINE.

VM To format and print line trace data for a 3720, 3725, or 3745, use ACF/TAP and specify INPUT=LINE.

VSE To format and print line trace data for a 3705, use ACF/TAP or TPRINT. For TPRINT, specify LINE=*name* or LINE=ALL. For ACF/TAP, specify INPUT=LINE.

VSE To format and print line trace data for a 3720, 3725, or 3745, use ACF/TAP and specify INPUT=LINE.

For more information on printing trace output, see "Formatting and Printing Trace Records" on page 361.

Line Trace Output (CS Type 2)

Figure 46 shows an example of line trace output for a 3705 Communications Controller with a type 2 communication scanner.

The trace fields are explained after the figure. In addition to the fields described here, additional operating-system-dependent fields may appear. For a description of these fields, see Table 19 on page 367 and Table 20 on page 367.

```
LINE LINE01 LRC(000,000) INBOUND ACTIVE RNTIME=1D
  LCD C PCF A TIME 16 SCF 42 PDF D4   LCD C PCF A TIME 16 SCF 42 PDF 40
  LCD C PCF A TIME 16 SCF 42 PDF E5   LCD C PCF A TIME 16 SCF 42 PDF 40
  LCD C PCF A TIME 16 SCF 42 PDF E3   LCD C PCF A TIME 16 SCF 42 PDF 40
  LCD C PCF A TIME 16 SCF 42 PDF C1   LCD C PCF A TIME 16 SCF 42 PDF 40
```

Figure 46. Example of Line Trace Output (CS Type 2)

The following fields appear in the line trace for CS type 2.

The header portion of the line trace record contains these fields:

Field Header	Field Contents
LINE *linename*	The name of the node being traced. In Figure 46, the line name is LINE01.
LRC(*xxx,yyy*)	The number of records lost since the last trace record was written because the trace facility could not get a VTAM buffer. *xxx* is the destination's lost record count, and *yyy* is the source's lost record count.
INBOUND	MVS VM The direction of the data with respect to this host. It always says INBOUND because VTAM always receives the trace records from the NCP. VSE This field is replaced by a time stamp.
status	The status of the line being traced. In Figure 46, the line status is ACTIVE. The line status may also appear as DEACTIVATE or SLOWDOWN. • ACTIVE means that the line trace is active. • DEACTIVATE means that the line trace is not active. • SLOWDOWN means that the line trace is not active because the NCP is in slowdown mode. A status of DEACTIVATE or SLOWDOWN appears only in the last record to be sent for that line. It means that no more data will be sent until the trace is activated again for that line.
RNTIME=*hh*	Is a timer field. *hh* is a hexadecimal value indicating in tenths of a second the time at which the communication controller sent the completed line trace record to VTAM. This value is taken from a timer that is reset to 0 after 25.5 seconds. This value can be compared with the TIME value for the first level 2 interruption data contained in the second part of the line trace record. This comparison shows the elapsed time covered by this trace record. Comparing this with the number of level 2 interruptions contained in the trace record indicates the I/O activity during the elapsed time.

The data portion of the line trace record contains a timer value and values from the interface control word (ICW) and communication controller hardware registers at each level-2 interruption. Each printed line contains up to two sets of data, one for each of two level-2 interruptions.

To determine what book contains detailed descriptions of the fields in line trace records, see Table 80 on page 1165.

Field Header	Field Contents
LCD	The line control definer (LCD) indicates the type of protocol being used over the line. The LCD for a line is generally set when the NCP is loaded and remains unchanged throughout normal operation.
PCF	The primary control field (PCF) indicates the state of the line interface at any particular time.

Field Header	Field Contents
TIME	This hexadecimal value indicates, in tenths of a second, the time elapsed between level-2 interrupts as they occur on the line interface as data is received. Level-2 interrupts are represented by the data (PDF). The value of this timer is reset to 0 every 25.5 seconds.
SCF	The secondary control field (SCF) is used as a status and operation modifier by the communication scanner and the control program.
PDF	The parallel data field (PDF) is used as a character buffer. For a transmission, the program places the characters to be sent in the PDF, and transmits them to the line interface. For a receive operation, the character is assembled, then transferred by hardware to the PDF.

Line Trace Output (CS Type 3)

Figure 47 shows an example of line trace output for a 3705 Communications Controller with a type 3 communication scanner. This trace record shows the operating status of a line each time an NCP buffer is filled, instead of each time a character is transmitted.

The output in this example applies to all MVS, VM, and VSE utilities, except ACF/TAP. MVS, VM, and VSE utilities produce similar output for line traces; the differences are noted in the following text.

The trace fields are explained after the figure. Most fields in this record are similar to the example for the type 2 scanner, but the type 3 scanner has some additional fields. To determine what book contains detailed descriptions of the additional fields in CS type 3 output, see Table 80 on page 1165.

```
LINE LINE01 LRC(000,000) INBOUND ACTIVE RNTIME=1D
  STATUS SCF LCD PCF EPCF STAT1 STAT2 TIME ADDR CNTL IOBLXB DISP ICW CCBLV
         44  9   9   00    00   01    80   03   71    30    AA   11  1BB2
  TEXT C0B4
```

Figure 47. Example of Line Trace Output (CS Type 3)

In addition to several fields shown in the line trace for CS type 2, the following fields appear in the line trace for CS type 3:

Field Header	Field Contents
EPCF	The extended primary control field. These bits extend the meaning of the PCF.
STAT1	Status byte 14 of the ICW.
STAT2	Status byte 15 of the ICW.
TIME	This hexadecimal value indicates, in tenths of a second, the time elapsed between level-2 interrupts as they occur on the line interface as data is received. Level-two interrupts are represented by the data. The value of this timer is reset to 0 every 25.5 seconds.
ADDR	The SDLC frame address field.
CNTL	The SDLC frame control field.
IOBLXB	The 1-byte command field from the IOB/LXB control block.

Field Header	Field Contents
DISP or XDISP	MVS VM DISP or VSE XDISP is byte 0 from the communication scanner display register.
ICW	Byte 13 of the interface control word.
CCBLV or LVL2	MVS VM CBBLV or VSE LVL2 is the address of the level 2 interruption processor routine, taken from field CCBL2 in the NCP CCB control block.
TEXT	The contents of the NCP buffer being transmitted in both hexadecimal and EBCDIC.

Network Controller Line Trace (3710 Only)

The network controller line trace traces SDLC, BSC, and start/stop data link control frames sent or received by an IBM 3710 Network Controller.

The 3710 physical unit provides a network controller line trace on a physical unit type 2 node. The network controller line trace traces data link control (DLC) protocols, such as DLC frames exchanged between a network controller and an NCP, or DLC frames exchanged between a network controller and a control unit.

The network controller line trace facility also traces the synchronous data link control (SDLC), binary synchronous control (BSC), and start/stop DLC frames that are sent or received by a line adapter.

When tracing an SDLC or BSC line, you can choose to trace all the frames (control and data) or just the data frames. All frames are traced on a start/stop line.

Note: If the data is not from a type 1, 2, or 3 scanner, VTAM will not process the data. You must use ACF/TAP.

When to Use the Network Controller Line Trace

Run this trace if you have an incorrect output problem, a performance problem, or you suspect a network controller problem.

Network Controller Line Trace Operation

Start the network controller line trace with the MODIFY TRACE command. For more information on the MODIFY TRACE command for the network controller line trace, see "MODIFY TRACE Command" in *VTAM Operation*.

Active traces stop if:

- The network controller is reset (for example, during a failure or a configuration load)
- The network controller enters slowdown (shown in the header of the RECTRD RU)
- The line fails during the trace.

If the network controller recovers after a failure, and it is not the only one on the line, the trace will resume.

Make sure that GTF is active before starting this trace. See "Starting the Generalized Trace Facility (GTF) (MVS)" on page 358.

Use ACF/TAP with NETCTLR specified to format and print the output. For more information on printing trace output, see "Formatting and Printing Trace Records" on page 361.

Network Controller Line Trace Output

VTAM receives the trace data from a network controller in a series of RECTRD RUs, which are put into trace records. Use ACF/TAP to print the trace output.

Scanner Interface Trace (3720, 3725, and 3745 Only)

The scanner interface trace, a joint function of VTAM and the NCP, stores the operating parameters of a line whenever a 3720, 3725, or 3745 Communication Controller sends or receives a message. Although the trace is controlled by VTAM, the information in the trace records is collected by the NCP. The scanner interface trace collects the operating status of a line whenever the communication controller sends or receives a message (PIU).

When you start the trace, you can choose how many bytes of data you want to trace. The default is the entire PIU. You can only use the scanner interface trace on lines operating in network control mode.

For each communication controller, a combination of eight line traces or scanner interface traces can be active at a time. The number of active scanner interface traces to be allowed is specified during NCP generation (default = 2). In a multisystem network in which the communication controller is connected to more than one host processor, the number of active traces allowed is distributed among the connected host processors on a first-come, first-served basis.

Note: As the number of active scanner interface traces increases, the system becomes less efficient.

When to Use the Scanner Interface Trace

Use the scanner interface trace to determine whether the NCP or a line is causing a problem. The scanner interface trace collects inbound data before the NCP processes it, and collects outbound data after the NCP processes it. Therefore, if outbound data is correct in a scanner interface trace, but the device does not react properly, the problem is in either the line or the device.

Scanner Interface Trace Operation

Start the scanner interface trace with the MODIFY TRACE command or the TRACE start option. The COUNT option allows you to choose how many bytes of data you want traced (0 to 254 or ALL). The default is the entire PIU.

For more information on the MODIFY command for the scanner interface trace, see "MODIFY TRACE Command" in *VTAM Operation*. For more information on the TRACE start option, see the "TRACE for Buffer, I/O, NCP Line, SIT, or SMS Traces" in the *VTAM Resource Definition Reference*.

MVS Make sure that GTF with the TRACE=USR option is active before starting this trace. Use ACF/TAP with INPUT=LINE to format and print the output.

VM To collect scanner interface trace information, use the TRSOURCE command.

VM To format and print scanner interface trace output, use ACF/TAP.

VSE Use ACF/TAP to format and print the output.

For more information on printing trace output, see "Formatting and Printing Trace Records" on page 361.

Transmission Group Trace

The transmission group (TG) trace, a joint function of VTAM and the NCP, traces the sequence of PIUs being sent through a transmission group. The transmission group trace shows PIU traffic on a transmission group as though the transmission group were a single line. The sequence of PIUs traced is the sequence of their entry to and exit from the transmission group, not the sequence of actual transmission along the physical lines in the transmission group. The line trace shows the sequence of PIUs transmitted and received on a physical line.

When to Use the Transmission Group Trace

Use the transmission group trace instead of a line trace if your failure is restricted to sessions using a specific transmission group and you have more than one active line in a TG.

Note: VTAM can start a transmission group trace only for the resources that it owns. A data host, which does not own any NCPs, cannot start a transmission group trace.

Transmission Group Trace Operation

Start the transmission group trace with the MODIFY TRACE command. The name specified in the ID operand is any line "associated" with the transmission group trace. The trace is started for the transmission group in which the specified line resides. This line and its associated link station must be active before the transmission group trace is started. If a line trace is already active for the chosen line within a transmission group, you must stop it before starting a transmission group trace.

Once activated, the transmission group trace remains active until one of the following occurs:

- The operator stops the trace.
- The associated line or link station is deactivated or fails.
- The NCP goes through automatic network shutdown.
- The NCP goes into slowdown mode.

If the transmission group trace is ended because the associated line or link station fails or is deactivated, the operator can restart it by issuing the MODIFY command for another line in the TG.

For more information on the MODIFY TRACE command for the transmission group trace, see "MODIFY TRACE Command" in *VTAM Operation*.

Note: If the data is not from a type 1, 2, or 3 scanner, VTAM will not process the data. You must use ACF/TAP.

MVS Make sure that GTF with the TRACE=USR option is active before starting this trace. See "Starting the Generalized Trace Facility (GTF) (MVS)" on page 358.

Use IPCS or ACF/TAP to format and print these trace records. For IPCS, specify USR(LINE) or USR(FF2) on the GTFTRACE option. For ACF/TAP, specify INPUT=LINE.

VM To collect transmission group trace information, use the TRSOURCE command. To format and print trace output, use the ACF/TAP or TRACERED commands.

VSE To format and print transmission group trace output, use TPRINT or ACF/TAP.

For more information on printing trace output, see "Formatting and Printing Trace Records" on page 361.

Transmission Group Trace Output (MVS) (VM)

Figure 48 is an example of transmission group trace records

The trace fields are explained after the figure. In addition to the fields described here, additional operating system-dependent fields may appear. For a description of these fields, see Table 19 on page 367 and Table 20 on page 367.

```
LINE   LINE23    LRC(000,000)   INBOUND       ACTIVE       RNTIME=00
       TEXT 42000000 2000009A 00000004 0000000C 0E00000B
            0005808D 000ACB80 00A0
       TEXT 6E000000 F000009A 00000004 00000001 00000000
            00000032 20000000 00000823 00000000 00000000
            00000123 00000040
```

Figure 48. Example of Transmission Group Trace Output MVS VM

MVS VM Table 27 describes the fields in the transmission group trace.

Table 27. MVS VM Fields in the Transmission Group Trace

Field Header	Meaning
LINE *linename*	The name of the line "associated" with the transmission group trace. This line name was specified in the MODIFY TRACE command that started the transmission group trace. The line name is LINE23 in Figure 48 on page 392.
LRC(*xxx,yyy*)	The number of records lost since the last trace record was written because the trace facility could not get a VTAM buffer. *xxx* is the destination's lost record count, and *yyy* is the source's lost record count.
INBOUND	The direction of the data with respect to this host. It always says INBOUND because the data is always received from the NCP.
status	The status of the line being traced. In Figure 48, the line status is ACTIVE. The line status may also appear as DEACTIVATE or SLOWDOWN. • ACTIVE means that the line trace is active. • DEACTIVATE means that the line trace is not active. • SLOWDOWN means that the line trace is not active because the NCP is in slowdown mode. A status of DEACTIVATE or SLOWDOWN appears only in the last record to be sent for that line. It means that no more data will be sent until the trace is activated again for that line.
RNTIME=*hh*	Is a timer field, where *hh* is a hexadecimal value indicating, in tenths of a second, the time at which the communication controller sent the completed line trace record to VTAM. This value is taken from a timer that is reset to 0 every 25.5 seconds. This value can be compared with the TIME value for the first level 2 interruption data contained in the second part of the line trace record. This comparison shows the elapsed time covered by this trace record. Comparing this with the number of level 2 interruptions contained in the trace record indicates the I/O activity during the elapsed time.
TEXT	Shows the TH and RH for each PIU that traversed the data path through the transmission group. If the PIU is for data flow control, session control, or network control, the full RU portion of the PIU is also included in the trace record. The RU may contain sense data. If the PIU is function management data (FMD) and contains an FM header, six bytes of the RU are included in the transmission group trace record. If an FMD PIU without an FM header is traced, the RU is not included.

Transmission Group Trace Output (VSE)

Figure 49 is an example of transmission group trace records. The trace fields are explained after the figure.

```
LINE   LINE23    87.150/09:15:06.18                    ACTIVE      RNTIME=00
 TEXT 42000000 2000009A 00000004 0000000C 0E00000B 0005808D 000ACB80 00A00000
      00000032 20000000 00050101 0000000B 0040
 TEXT 6E000000 F000009A 00000004 00000001 00000000 0005808D 000ACB80 00A00000
      00000123 00000040 00050101 0000000B 0000
```

Figure 49. Example of Transmission Group Trace Output VSE

Table 28 on page 394 describes the fields that appear in the transmission group trace for VSE.

Table 28. VSE VM *Fields in the Transmission Group Trace*

Field Header	Meaning
LINE *linename*	Is the name of the line associated with the transmission group trace. This line name was specified in the MODIFY TRACE command that started the transmission group trace. The line name is LINE23 in Figure 49 on page 393.
yy.jjj/hh:mm:ss.hh	Is the year (yy), the Julian date (jjj), and the time in hundredths of a second (hh:mm:ss.hh) that the trace record was written. This field is "87.150/09:15:06.18" in the first trace record in Figure 49 on page 393.
LRC(*xxx,yyy*)	Is the number of records lost since the last trace record was written because of the inability of the trace facility to get VTAM buffers. *xxx* is the destination's lost record count, and *yyy* is the source's lost record count. This field is not present in the trace record unless records are lost.
TRANSMIT or RECEIVE	Indicates, for full-duplex lines, the side of the line used for the transmission.
status	The status of the line being traced. In Figure 49, the line status is ACTIVE. The line status may also appear as DEACTIVATE or SLOWDOWN. • ACTIVE means that the line trace is active. • DEACTIVATE means that the line trace is not active. • SLOWDOWN means that the line trace is not active because the NCP is in slowdown mode. A status of DEACTIVATE or SLOWDOWN appears only in the last record to be sent for that line. It means that no more data will be sent until the trace is activated again for that line.
RNTIME=*hh*	Is a timer field, where *hh* is a hexadecimal value indicating, in tenths of a second, the time at which the communication controller sent the completed line trace record to VTAM. This value is taken from a timer that wraps around to 0 after 25.5 seconds. This value can be compared with the TIME value for the first level 2 interruption data contained in the second part of the line trace record. This comparison will show the elapsed time covered by this trace record. Comparing this with the number of level 2 interruptions contained in the trace record indicates the I/O activity during the elapsed time.
TEXT	Shows the TH and RH for each PIU that traversed the data path through the transmission group. If the PIU is data flow control, session control, or network control, the full RU portion of the PIU is also included in the trace record. The RU may contain sense data. If the PIU is function management data (FMD) and contains an FM header, six bytes of the RU are included in the transmission group trace record. If an FMD PIU without an FM header is traced, the RU is not included.

Dynamic Trace Utility for Integrated Communication Adapters (VM)(VSE)

This trace facility of VSE applies to ESA-capable processors. It can record communication adapter line traces on tape while the line and the communication adapter are both active. It can also print selected trace entries after a line-related error has occurred.

The communication adapter line trace should be used when line-related errors are suspected on lines attached to the integrated communication adapter (ICA), for example:

- Frequent unexplained line outages
- Unusual sense or status in I/O error messages
- Device outages that cannot be explained with I/O or buffer traces.

See Table 80 on page 1165 to determine what book describes dynamic tracing of ICAs.

Chapter 9. Using the VTAM Internal Trace (VIT)

This chapter includes the following sections:

Most VTAM traces show the information flow between the VTAM program and other network components. However, the VTAM internal trace (VIT) provides a record of the sequence of events *within* VTAM. These internal events include the scheduling of processes (for example, POST, WAIT, and DISPATCH), the management of storage (for example, VTALLOC), and the flow of internal PIUs between VTAM components.

Together with the operator console listing and a dump, output from the VIT can help you reconstruct sequences of VTAM events and find internal VTAM problems more easily.

Activating the Internal Trace

Trace data for the following VIT options is always automatically recorded in the internal table:

- API
- MSG
- NRM
- PIU
- SSCP.

Use one of the following methods to start the VIT:

- You can use the TRACE start option, with TYPE=VTAM specified, to start the VIT when you first start VTAM. See "TRACE for VTAM Internal Trace" in the *VTAM Resource Definition Reference* for more information on how to use the TRACE start option.
- You can use the MODIFY TRACE command, with TYPE=VTAM specified, to start the VIT after you have started VTAM. See "MODIFY TRACE Command" in *VTAM Operation* for more information on how to use the MODIFY TRACE command.

Notes:

1. MVS VM To prevent the VIT table from being overwritten, VTAM disables the internal VIT when it issues SDUMP and when a FFST probe is tripped.
2. The minimum trace table size is 50 pages. Because the five trace option defaults are always running, the table may wrap many times.

Selecting Trace Options

Both the TRACE start option and the MODIFY TRACE command have an OPTION operand you can use to select VIT options. Select one or more of the following options to indicate the VTAM functions you want to trace.

Note: If you do not deactivate the VIT before you attempt to change an option, the options that are currently in effect will remain in effect. See "Deactivating the VTAM Internal Trace" on page 410 for more information on deactivating the VIT.

- **API** option (for application program interfaces)

  This option helps you determine whether an application program is causing a problem. API entries are written for RPL macros, RPL exit routines, user exit routines, and user posts.

  Trace data for this option is always automatically recorded in the internal table.

- **APPC** option (for LU 6.2 application program interfaces)

  This option helps you determine whether an LU 6.2 application is causing a problem. LU 6.2 entries are written for APPCCMD macro invocations, user posts and exit scheduling by LU 6.2 code, calls to a security manager for security processing, and message unit transmissions between LU 6.2 components.

- MVS **CFS** option (for coupling facility interfaces)

  This option helps you determine problems with VTAM's interface with the MVS coupling facility. CFS entries are written when VTAM issues MVS macros to request coupling facility related services.

- **CIO** option (for channel input and output)

  This option helps you isolate problems related to channel I/O. CIO entries are written for attentions, error recovery, interruptions, HALT I/O SVC, and START I/O SVC.

- **ESC** option (for execution sequence control)

  This option helps you track in detail the flow of requests for a given process.

- **LCS** option (for local area network (LAN) channel stations)

  This option helps you isolate problems occurring during activation of, deactivation of, and data transfer from an IBM 3172 Interconnect Controller. The LCS option enables tracing of data that VTAM receives from an IBM 3172 Interconnect Controller at four levels: LCSX (channel), LCSP (port or adapter), LCSS (SAP), and LCSL (line).

- **LOCK** option (for locking and unlocking)

  This option helps you determine when VTAM modules get and release locks.

- **MSG** option (for messages)

  This option helps you:

  – Correlate other VIT entries with the console messages even if you lose the console sheet. MSG entries are written for all messages to the VTAM operator.

  – Match the console log to a surge of activity shown in the VIT. OPER entries are written for all VTAM commands issued at an operator console.

  Trace data for this option is always automatically recorded in the internal table.

- **NRM** option (for network resource management)

  This option helps you follow the services of the network resource management component. These include the assignment of, references to, and the deletion of certain VTAM resources such as node names, network addresses, and control blocks. NRM entries are written for SRT macros issued by VTAM modules.

  Trace data for this option is always automatically recorded in the internal table.

- **PIU** option (for path information unit flows)

  This option, like the I/O and buffer contents traces, helps you isolate problems to hardware, to the NCP, or to VTAM. Unlike I/O and buffer contents traces, with this option PIU entries are written for all PIUs that flow internal and external to VTAM.

  Trace data for this option is always automatically recorded in the internal table.

- **PSS** option (for process scheduling services)

  This option helps you track the flow of requests through VTAM. PSS entries are written for the VTAM macros that invoke and control PSS, scheduling and dispatching VTAM routines.

- **SMS** option (for storage management services)

  This option helps you isolate problems caused by storage shortages. When used with the SSCP or PSS trace options, it can also help you isolate internal VTAM problems. SMS entries are written when SMS macros are used to request or free fixed- or variable-length buffers. SMS entries are also written when VTAM expands or attempts to expand a buffer pool.

- **SSCP** option (for system services control point request scheduling and response posting)

  This option helps you isolate a VTAM problem to a specific VTAM component or module. SSCP entries are written for the request/response units (RUs) sent between VTAM components. This option also records information for the APPN CP.

  Trace data for this option is always automatically recorded in the internal table.

- MVS **TCP** option (for use with the SNA over TCP/IP feature application program interfaces)

  This option helps you trace the communication between VTAM and TCP/IP. Events include:

  - Socket API calls and completions
  - IUCV calls and completions
  - Invocations of the SOCTREE macro, which controls updating a binary tree of socket control blocks
  - Tracing MPTN formats

- MVS VM **VCNS** option (for VCNS application program interfaces)

  This option helps you determine whether a VCNS application is causing a problem. VCNS entries are written for VCNSCMD macro invocations, user posts, and exit scheduling by VCNS code, and work element transmissions between VCNS components.

The VIT always traces the exception conditions listed in Table 29 and all the default VIT options listed under "Activating the Internal Trace" on page 398.

Table 29 (Page 1 of 2). Exception Conditions Always Traced by the VTAM Internal Trace

Option	Exception conditions traced
APPC option	• ACA and ACI entries when issuing the following commands: – SEND ERROR – DEALLOC ABNDxxxx – REJECT • ACRC and ACSN entries • Other entries with non-zero return codes (except RPL6RCSC)
CFS MVS	Entries with non-zero return codes
LCS option	LCSL, LCSP, LCSS, and LCSX entries with non-zero reason codes

Table 29 (Page 2 of 2). Exception Conditions Always Traced by the VTAM Internal Trace

Option	Exception conditions traced
SMS option	Entries with non-zero return codes and EXPN entries if a buffer pool expansion fails
(No option)	All SNAP entries.

Table 30 summarizes the VIT options and the records they create.

Table 30. VIT Options and the Records They Create

API	APPC	CFS MVS	CIO	ESC	LCS	LOCK	MSG	NRM	PIU	PSS	SMS	SSCP	TCP MVS	VCNS MVS VM
AIx	ACAx	CFCx	ADPx	ESC	LCSx	LKEX	**MSGx**	**BSPx**	**DSCx**	ADSP	AREL	**ALSx**	IPx	CNA
IOx	ACIx	CFDx	ATTx			LKSH	**OPEx**	**BSSx**	**NRSx**	ATSK	EXPN	**AP**	MIx	CNPx
RE	ACPx	CFEx	CONN			ULKA	**QRYL**	**BSXx**	**PIUx**	AXIT	FBLx	**A2**	MPTx	CNRx
UEx	ACRx	CFFC	CPTO			UNLK	**TRNx**	**PROx**	**RDSx**	BTSK	GBLx	**CCx**	NMPx	NSD
UP	ACSN	CFLx	DISC					**RCEx**		DSP	QREQ	**CIx**	SOCx	VCCx
	ACUx	CFNF	ERPx					**SRTx**		DTSK	RELS	**COx**		VCDQ
	MUx	CFRB	HIOx							ETSK	REQx	**CPPx**		VCPx
	RACR	CFUS	INTx							EXIT	VTAL	**CPRx**		
			LICx							IRBx	VTFR	**CPWx**		
			LNKx							POST		**CRx**		
			LSNx							QUEx		**DBx**		
			PIC							RESM		**DLTx**		
			PKTx							SCHD		**GNAx**		
			SIOx							SRBx		**SPTx**		
			XIDx							VPST		**TGMx**		
										VRSM		**TREx**		
										VWAI				
										WAIT				
										XPST				

Notes:

1. The **** (FFST), ABND, and LOST trace records are not activated by specific VIT options. They are activated as a result of exception conditions.
2. The entries in boldface type are forced, that is, always traced even if they are not specified by the user.

Internal and External Trace Recording

VTAM can write the VIT trace data to an internal table or an external device, such as a disk or tape. You specify internal or external with the MODE operand of the TRACE start option or the MODIFY TRACE command. The VIT record contains the same information regardless of the MODE selected.

MVS VM You can record data externally and internally at the same time, and if desired, you can have different sets of trace options active for each mode. The default trace options— API, MSG, NRM, PIU, and, SSCP—are always recorded internally.

Recording Traces in Internal Table (MODE=INT)

If you set MODE=INT on the MODIFY TRACE command or as a TRACE start option, or if you let MODE default to INT, VTAM writes the VIT trace records in an internal trace table. The table is allocated and initialized in a storage area:

- MVS The storage area is in extended common service area (CSA).
- VM The storage area is in common storage.

- VSE The storage area is in the extended system GETVIS area (SGA).

The SIZE operand of the TRACE start option specifies the number of pages (1–999) in storage to be allocated for the internal trace table. Each page is 4K. If you omit this option, the default size is 50 pages. If you specify fewer than 50 pages, VTAM uses 50. Because it is a wraparound table, specify enough pages to ensure that the VIT will not overwrite important trace records when the table fills and begins to wrap around.

The SIZE operand should also reflect the amount of storage available for the internal trace table. If there is not enough storage available for the number of pages specified, you will receive a message indicating that internal trace activation failed.

After the VIT is started, the size operand does not have a default. You can change the size of the internal trace table by issuing a MODIFY TRACE command with a new SIZE operand. However, if you change the table size while the VIT is running, the current internal trace table is freed, and VTAM starts a new one. Therefore, the trace information in the current table will be lost.

Figure 50 shows an example of VTAM internal trace records written with MODE=INT and printed in dump output. The five default trace options are always active.

```
E2C3C8C4 14202810 06308E88 00C17248  8652D180 00000000 C9D5E3D4 00000000  *  SCHD.......h.A..f.J.....INTM....  *
E2D9C2C4 14000000 06308E88 80000000  00000000 00000000 007FF158 00800000  *  SRBD.......h.............."1.....  *
E2D9C2E7 14000000 06308E88 00000000  00000000 00EBDE80 00EBDE80 824C4B82  *  SRBX.......h...............b<.b  *
C9D9C2C4 14000000 06308E88 80000000  00C17248 007DDB18 007FF158 00800000  *  IRBD.......h.....A...'..."1.....  *
D9C5D8E2 14170000 06308E88 06324810  81C23E1A 00010000 06300DA0 00000000  *  REQS.......h....aB..............  *
C4E2D740 1400A810 06308E88 00C17248  00000000 00000000 C9D5E3D4 06324810  *  DSP ..y....h.A.........INTM....  *
C3C3C900 14588000 06324A20 00000000  8652D13A 00000000 58588000 00000000  *  CCI.......¢.....f.J.............  *
D8E4C558 14482410 06308E88 00C17908  824EAA8C 0651C928 E3E2E6E4 06324810  *  QUE........h.A..b+....I.TSWU....  *
C5E7C9E3 14000010 06308E88 00C17248  8652CF74 80000000 C9D5E3D4 06324810  *  EXIT.......h.A..f.......INTM....  *
D9C5D3E2 14170000 06308E88 06324810  81C23E92 00000000 06300DA0 00000000  *  RELS.......h....aB.k............  *
C4E2D740 14582410 06308E88 00C17908  0651C928 0651C928 E3E2E6E4 0631D810  *  DSP .......h.A....I...I.TSWU..Q.  *
D8E4C558 14482810 06308E88 00C17248  86615658 0651C928 C9D5E3D4 0631D810  *  QUE........h.A..f/....I.INTM..Q.  *
C5E7C9E3 14000010 06308E88 00C17908  86615708 80000000 E3E2E6E4 0631D810  *  EXIT.......h.A..f/......TSWU..Q.  *
D9C5D8E2 14170000 06308E88 06324810  81C23E1A 00010000 06300DA0 00000000  *  REQS.......h....aB..............  *
C4E2D740 14582810 06308E88 00C17248  0651C928 0651C928 C9D5E3D4 06324810  *  DSP .......h.A....I...I.INTM....  *
C5E7C9E3 14000010 06308E88 00C17248  8652CF74 80000000 C9D5E3D4 06324810  *  EXIT.......h.A..f.......INTM....  *
```

Figure 50. Unformatted VIT Records in Dump Output (MODE=INT)

The first line of the internal trace table is the header. (The header is not included in the preceding example.) The header contains the following status information:

Byte (hex)	Contents
00–07	Zero
08–0F	Present wraparound time stamp
10–17	Last wraparound time stamp
18–1B	Address of the most recent entry in the table
1C–1F	Address of the last entry in the table.

Dump Analysis: MVS Several dump analysis tools are available for the VTAM internal trace:

- VITAL extracts an internal VIT from a dump for use with the VIT analysis tool. See "VITAL" on page 322 for a description.
- VTBASIC displays the VIT table. See "VTBASIC" on page 326 for a description.

- VTVIT displays the VIT options that were in effect at the time of the dump and whether the trace was running internally, externally, or both. See "VTVIT" on page 340 for a description.

Use the following procedure as an alternative to the dump analysis tools.

Examine the internal trace table by taking a dump of VTAM with the appropriate storage area. Find the internal trace table in the dump by first locating the ATCVT. The pointer to the ATCVT (ATCLCPTR) is at low-storage location MVS VSE X'408' and VM X'200'.

MVS If this low-storage location is not available in a dump, use the pointer in the MVS CVT control block extension (CVTATCVT) to find the VTAM control block AVT. Location hex 00 in the AVT points to the ATCVT. In the ATCVT, field ATCITTBL contains a pointer to the trace table. ATCITTBL is located at offset X'C'. The internal trace table contains a 32-byte header followed by 32-byte trace records.

Recording Traces in External File (MODE=EXT)

If you set MODE=EXT on the MODIFY TRACE command or as a TRACE start option, VTAM writes the VIT trace records in an external trace file. The external trace file contains the same information, except the first-line header record, as the internal trace table.

Use external mode when you must collect large amounts of trace data, to lessen the chance of wraparound and lost data. However, unless you are using a tape, the trace data can still fill the external DASD or data file and begin overwriting itself if you do not ensure that the external trace file is large enough.

Note: MVS VM When you specify MODE=EXT, information is still written to the internal trace table for the default options.

VSE When recording externally, the value specified for the SIZE operand on the MODIFY TRACE command or the TRACE start option is used in a formula to determine the number of 4K fixed internal buffers that can be placed in the area allocated for the trace tables. The formula is as follows:

```
((SIZE x ATCPGSIZE) - 16) / 2K
```

The 16 bytes that are subtracted represent the buffer header, where no data can be placed.

If the SIZE operand is changed while MODE=EXT is in effect, the current table contents are written to the trace file before the old table is freed.

Note: If the requested size is not available, VTAM returns to the default value of SIZE=50. When a low value for SIZE results in lost trace records, the lost record count (LRC) will be other than zero in the trace output.

MVS The external trace file is produced by GTF, and the default file name is SYS1.TRACE. You can print the internal trace data with IPCS or TAP. If you use IPCS to print the data, specify the GTFTRACE option, and set USR(FE1). For more information on using TAP and IPCS to print output, see "Using ACF/TAP" on page 361 and "Using IPCS with GTF Trace Option (MVS)" on page 362. See Table 80 on page 1165 to determine what books describe TAP and IPCS.

VM To produce the external trace file, use the TRSOURCE command. To format and print the external trace file, use the TRACERED command. Figure 51 on page 404 shows an example of CPTRAP output. For more information see "Collecting External Trace Data (VM)" on page 359.

```
3D 0E VTAM       GCS USER REQUESTED GTRACE
       TIME OF DAY CLOCK = 97F6F2B3CB2BAA40
       LENGTH OF GTF HEADER AND TRACE DATA = 0030
       FORMAT ROUTINE ID = FD
       EVENT IDENTIFICATION = EFE1
VTFR ISSR     0AD1D4   AREA 00105308 SP#  000000E5 LENGTH  00000368 RC     00000000 INIT  00000000 00000000
3D 0E VTAM       GCS USER REQUESTED GTRACE
       TIME OF DAY CLOCK = 97F6F2B3CEA29E00
       LENGTH OF GTF HEADER AND TRACE DATA = 0030
       FORMAT ROUTINE ID = FD
       EVENT IDENTIFICATION = EFE1
CCI  FLGS 00 ISSR 09D2EA CBID 54 SAV 106938 SVID INS1 CID  10011001 RU 00000000 00000000 00000000
3D 0E VTAM       GCS USER REQUESTED GTRACE
       TIME OF DAY CLOCK = 97F6F2B3D1131C80
       LENGTH OF GTF HEADER AND TRACE DATA = 0030
       FORMAT ROUTINE ID = FD
       EVENT IDENTIFICATION = EFE1
QUE  ISSR     A1289E   PST  5485C008 PAB  108436D8 WG 00 ELM 104F40 PABCHN 00000000 NAME NORT
    RPH      20000000
3D 0E VTAM       GCS USER REQUESTED GTRACE
       TIME OF DAY CLOCK = 97F6F2B3D3807860
       LENGTH OF GTF HEADER AND TRACE DATA = 0030
       FORMAT ROUTINE ID = FD
       EVENT IDENTIFICATION = EFE1
FBLK ISSR     A10EEA   AREA 00104E20 ANCH 00849928 LENGTH  00000060 RC     00000000 INIT  500AA664
           00000000
```

Figure 51. Formatted VTAM Internal Trace (VIT) Records in CPTRAP Output (MODE=EXT) VM

VSE The external trace file is produced by TPRINT, and the default file name is VTAM.TRACE.FILE. To print the trace data, use TPRINT or ACF/TAP. Figure 52 and Figure 53 on page 405 show examples of TPRINT output. For more information see "Using TPRINT (VSE)" on page 363. To determine the book that describes ACF/TAP, see Table 80 on page 1165.

```
 VTAM INTERNAL TRACE        85.295/18:35:54.01   LRC(62)
  VTAL  E5E3C1D3  34000000  0088C058  FFFFE5D7  60718532  00000078  00000000  00000000  VTAL.....H....VP-.E.............
  QUE   D8E4C540  340E0010  3297F010  00980BE0  009CD010  00000000  D9C1E3E6  00000001  QUE .....P0..Q..........RATW....
  VTAL  E5E3C1D3  34000000  0088C660  FFFFE5D7  607185B0  00000140  00000000  00000000  VTAL.....HF-..VP-.E.... ........
  QUE   D8E4C540  34490010  2497F9A8  0072BF18  0088C660  00000000  C3C6C6C5  00000000  QUE .....P9Y.....HF-....CFFE....
  VTFR  E5E3C6D9  34000000  0088C058  FFFFE5D7  60718AB0  00000078  00000000  00000000  VTFR.....H....VP-...............
  ADSP  C1C4E2D7  32000000  0097F010  00000000  00000000  00000000  80000000  00980BE0  ADSP.....P0.................Q..
  DSP   C4C9E2D7  320E0010  0097F010  00980BE0  809CD010  80000000  D9C1E3E6  00A31360  DSP......P0..Q..........RATW.T.-
  EXIT  C5E7C9E3  32000010  3297F010  00980BE0  50A439AC  80000000  00A43C00  00A31360  EXIT.....P0..Q..&U.......U...T.-
  AXIT  C1E7C9E3  32000000  3297F010  00000000  00000000  00000000  00000000  00000000  AXIT.....P0....................
```

Figure 52. Unformatted VTAM Internal Trace (VIT) Records in TPRINT Output (MODE=EXT) VSE

```
                              VTAM TRACE FILE UTILITY                          94.308  10:05:02    PAGE     1

MSG             TID  24   SAVE  015C53B1 MSGID 315I     ISSR  815C4D3A MODID RACTI    DEST  SYSTEM9C TYPE S
MSG2                     TEXT  EXT200
VTAL            TID  24   AREA  011E8308 SP#   FFFFD7E5 ISSR  81F9F6E6 LEN   00000150 INIT  00000000
                                                 RC    00000000                 FLAG  00
GBLK            TID  24   RC    00       ID    14       VTA   00       AREA  011AE020 ANCH  027C3088
                ISSR 81F9F994              LEN   00000098 INIT  00000000 RLEN  0000007E
VTFR            TID  24   AREA  011E8308 SP#   FFFFD7E5 ISSR  81F9FD6C LEN   00000150 INIT  00000000
MSG             TID  24   SAVE  015C53B1 MSGID 199I     ISSR  815C4EC8 MODID RACTI    DEST  SYSTEM9C TYPE S
⋮
WAIT            TID  24                    FLGS  21       FLGS1 00       PST   003FF228 PAB   00424800 ISSR  81F9AAA6
                                                 WEQ   80000000 NAME  CFR2     RPH   02799010
QUE             TID  24   CBID  0E       FLGS  400400   PST   003FF118 PAB   027C7988 ISSR  8236F72C
                                                 WEA   02676F78 NAME  RATW     RPH   00000001
REQS            TID  24   BPID  LP       PST   00000024 BUF   02797010 ISSR  803C55A2 BUF#  0001
                                                 REG1  02765D90 RC    00000000
⋮
```

Figure 53. Formatted VTAM Internal Trace (VIT) Records in TPRINT Output (MODE=EXT) VSE

Module Names in Internal Trace Records

Many VTAM internal trace records include the associated module names in EBCDIC, without the IST prefix and, for some types of trace records, without the sixth letter. For example, you would see TSSR for module ISTTSCSR. You can save time by scanning for these module names when you are following the logic flow through VTAM. You can sometimes isolate a VTAM problem to a specific component or module without even looking at a dump.

Module names can also be determined from the ISSR field in some VIT records. If the issuer is an MVS LPA module or VM DCSS or VSE SVA , the address can be found in the VTAM module list (which currently contains LPA modules).

The following names are exceptions to the naming convention:

CPSS	Session Services for CP-CP Sessions PAB
DIIO	Disk I/O PAB
DSME	Directory Services Management Exit PAB
DSVC	Directory Services PAB
LUSS	Session Services for LU-LU Sessions PAB
MSTI	MST Multiple Domain Server Router PAB
MSTR	MST Multiple Domain Server Router PAB
TRS	Topology Routing Services PAB
XPRT	Transaction Program PAB

VTAM component IDs are listed in Table 31.

Table 31 (Page 1 of 3). VTAM Component IDs

ID	Acronym	Component Name
@@		All or multiple components
AC	CONFIG	Activation
AD	APUNS	Adjacent PU network services
AI	API	Application program interface
AM	ADDR-MGT	Address management

Table 31 (Page 2 of 3). VTAM Component IDs

ID	Acronym	Component Name
AP	PSS	Process scheduling services
AS	APPC-SVC	APPC services
AT	APPNTGMT	APPN transmission group management
BS	BFSS	Boundary function session services
CD	CDRM	Cross domain resource manager
CF MVS	CFS	Coupling facility services
CI	OCI	Operator command interface
CN MVS VM	CNS	Common network services
CO	COPR	Control operator
CP	SSCP	System services control program functions
CS	CONFGSVC	Configuration services
DC	DLC	Data link control
DE	CONFIG	Deactivation
DL	DLUS	Dependent LU server
DP	DATACOMP	Data compression
DR	DS	Directory services
DS	DSS	Data space services
EN	ENVIRO	VTAM environment
EV	ERVR-MGR	Explicit route virtual route (ER/VR) manager
FD MVS VM	FMT-DMP	Formatted dump
FS MVS	CFS	Coupling facility services
FU	FUNCTION	SSCP functions
GN	SYSGEN	System generation
HS	HS	Half-session
IE	IEF	Inline exit facility
IM	INT-MAP	Internal mappings
IN	INIT/TRM	VTAM initialization/termination
IP MVS	SNAIP	IP network access
IT	VIT	VTAM internal trace
IX	INT-MAC	Internal macroinstructions
LM MVS	TPFLOGON	Transaction processing facility (TPF) logon manager
LR	DLR	Dump/Load/Restart
LS	LSA	Link services architecture
LU	LUS	Logical unit services
MA	MAINT-SV	Maintenance services
MG	MGMTSVC	Maintenance services

Table 31 (Page 3 of 3). VTAM Component IDs		
ID	**Acronym**	**Component Name**
MS	MGS	Messages
MT	MST	Management services transport
NA	NAM	Network address management
ND	NLDM	Network logical data manager
NO	NOS	Network operator services
NR	NRM	Network resource management
NS	LNS	LU network services
OC	O/C	Open/Close ACB
OI MVS VM	OSI-RPI	VTAM OSI RPI
OR	SMS	Storage management services
PS	PS	Presentation services
PU	PUNS	Physical unit services
PV	PVI	Primitive VTAM interface
RA	TRACE	Trace services
RM	LRM	LU resources manager
SA	SYS-ATTC	System attach
SC	SSC	Session services CP-CP
SD	SYSDEF	System definition
SL	SSL	Session services LU-LU
SS	SSCPSS	Session services
TB	TSC-BSC	TSC Binary Synchronous Communication
TD	TSC-DLC	TSC-DLC
TL	TSC-LAN	Token Ring Local Area Network
TO MVS	TSO/VTAM	TSO/VTAM
TR	TRS	Topology routing services
TS	TSC	Transmission subsystem
TX	TSX-X25	X.25
US	USS	Unformatted session services
VI	VIO	VSAM I/O
VX	VSAMMACS	VSAM user macroinstructions
XP	TPS	Transaction program services
62	APPC	APPC LU6.2

SNAP Trace Record

Besides recording the previously mentioned functions, you can use the VIT to record 32, 64, 96, or 128 bytes of user-generated information in a SNAP trace. This SNAP trace can place additional information into a VTAM module that might be useful in resolving a VTAM problem. The VIT traces the specified bytes and enters the data as a SNAP trace entry in the internal trace table or in the external trace file.

Unlike other trace entries, you do not specify the SNAP trace as an option of the MODIFY command. Instead, your code must follow these steps:

1. Verify that the pointer to the ATCVT (ATCLCPTR) is not 0. ATCLCPTR is at low-storage location MVS VSE X'408' or VM X'200'.
2. Verify that ATCRACTR is not 0.
3. Call the internal trace recording routine (ISTRACTR) in a VTAM module. Field ATCRACTR in the ATCVT points to the trace recording routine. For the hex offset, see *VTAM Data Areas for MVS/ESA*, *VTAM Data Areas for VM/ESA*, or *VTAM Data Areas for VSE/ESA*.

Note: SNAP records cannot be taken during the early stages of VTAM initialization, nor can they be taken during the late stages of termination.

The SNAP trace requires the following information in registers 1, 14, and 15 on entry to the trace routine:

Register 1
Points to the 32, 64, 96, or 128 bytes to be recorded.

Register 14
Points to a 2-byte input flag field containing:

- Hex 141C for a 32-byte SNAP entry
- Hex 142C for a 64-byte SNAP entry
- Hex 143C for a 96-byte SNAP entry
- Hex 144C for a 128-byte SNAP entry.

These flags tell the trace recording routine to record a SNAP trace.

Two Bytes Past the Address in Register 14
Is the address returned to after recording a SNAP trace.

Register 15
Points to the SNAP trace recording routine (ISTRACTR).

When the trace recording routine returns control to the calling module, the contents of registers 6 and 7 are unpredictable. The contents of the other registers, however, are not changed.

The SNAP trace record is unformatted. You should format the first 4 bytes of the SNAP record for a 1- to 4-character EBCDIC record identifier (for example, C'ABCD').

Notes:

1. The 1- to 4-character EBCDIC record identifier should not match any existing VIT record identifier.
2. For the VIT snap entry to be used by the VIT analysis tool, the VIT entry name must be the first 2 to 4 characters of the VIT entry.

Deactivating the VTAM Internal Trace

You can stop one or more of the active VIT options by issuing the MODIFY NOTRACE command with TYPE=VTAM specified. You can specify which traces are to be stopped using the OPTION operand.

If you specify OPTION=ALL, all active user selected VIT options are stopped, but the internal trace table is not freed. If you are only recording externally, a new internal trace table is started when this command is issued. Exceptions and default options are still recorded during this idle period.

If you specify OPTION=END, all active user selected internal VIT options are stopped, and the internal trace table is freed. The internal trace table is reallocated to 50 pages for the default options.

If you specify MODIFY NOTRACE, TYPE=VTAM, MODE=INT, internal trace recording is stopped. If you specify MODIFY NOTRACE, TYPE=VTAM, MODE=EXT, external tracing recording is stopped. If you do not specify MODE, all options are stopped, but the default options are immediately restarted and recorded internally.

If you attempt to stop the VIT using MODIFY NOTRACE, OPTION=END and the termination hangs, you can issue MODIFY NOTRACE, OPTION=FORCE to stop the VIT. If you issue MODIFY NOTRACE, OPTION=FORCE but the VIT is not waiting to terminate, VTAM issues message IST1038I stating that the NOTRACE command has been rejected because the VIT is not in the waiting (to terminate) state.

For more information see "MODIFY TRACE Command," "MODIFY NOTRACE Command," and "START Command (MVS, VM)" in *VTAM Operation*.

Chapter 10. Using the VIT Analysis Tool (MVS, VM)

This chapter describes the following:

You can use the VIT analysis tool to obtain information about a VTAM internal trace (VIT) that you have recorded on or transferred to an external device. The tool provides the following functions:

- Storage analysis
- Request and response unit (RU) counting
- VIT extraction.

You can choose to process only the VIT records that fall within a given time range in the trace record. In addition you can choose to:

- Add a title and a short description to the first page of each report
- Format the output
- Create a mini-report at a specified interval.

For information on required target data sets for the tool, see "Installing the VTAM Dump Analysis Enhancements and the VIT Analysis Tool" in the *VTAM Network Implementation Guide*. If you want a customized interface to be active to select the trace analysis commands of the VTAM program, see "Customizing the IPCS Interface" in the *VTAM Network Implementation Guide* for information.

If you experience problems that you suspect to be related to the VIT analysis tool, see "VTAM Internal Trace (VIT) Analysis Tool Problems (MVS) (VM)" on page 42 for help.

Setting up and Running the VIT Analysis Tool

Do these steps to set up and run the tool:

1. Record a VIT on an external device or transfer a previously recorded VIT to an external device.
2. Set up to run the tool.
3. Create the parameters for the job.
4. Run the job.
5. Check the output.

The following sections describe each step.

Step 1. Record a VIT

You must have a VIT on an external device, such as a disk or a tape before you can use the tool.

MVS A VIT that has been internally recorded can be copied to an external device using the VTAMMAP VITAL function. For instructions on the VTAMMAP VITAL function, see "VITAL" on page 322.

VM Convert the TRSOURCE trfile to a "GTF-like" CMS file, using the following command:

```
TRACERED NAME traceid GTRACE
```

traceid is the collection of TRF files and is the same value as is specified in the TRSAVE and TRSOURCE commands. *traceid* will cause TRACERED to process all of the TRF files at once, producing a consolidated output file in chronological order. See Table 80 on page 1165 to determine what book contains more information on TRACERED.

Step 2. Set up to Run the Tool

Create a Data Set (MVS)

Create a data set specifying the input and output data sets and the tool's program name. You may use JCL, a CLIST, or a REXX exec to create your data set.

Batch Mode: Use the sample JCL shown in Figure 54. Lower case indicates required variable information. **The actual JCL is determined by your installation**. For example, A has been defined as a printer in the sample installation, and SYSOUT=A directs output to it. Similarly, the sample JCL assumes that all input datasets have been catalogued.

Modify the JCL by including appropriate DD names. Even though all DD names shown are not required for all runs, you might want to list them to avoid changing your JCL when you change parameters. The record format for the output data sets can be variable or variable blocked (RECFM=V or RECFM=VB).

```
//jobname  JOB (account),'user name',etc.
//ISTRAFT1 EXEC PGM=ISTRAFT1,REGION=0K
//STEPLIB  DD DSN=SYS1.SISTDBUG,DISP=SHR
//SUMMARY  DD SYSOUT=A,DCB=(RECFM=V,LRECL=84)
//DETAILS  DD SYSOUT=A,DCB=(RECFM=V,LRECL=84)
//LOG      DD SYSOUT=A,DCB=(RECFM=V,LRECL=124)
//OUTSTAN  DD SYSOUT=A,DCB=(RECFM=V,LRECL=124)
//VITEXT   DD SYSOUT=A,DCB=(RECFM=V,LRECL=124)
//PARM     DD DSN=userid.run1.parm,DISP=SHR
//TRACE    DD DSN=userid.run1.trace,DISP=SHR
```

Figure 54. Sample JCL for VIT Analysis

As shown in Figure 54, the JCL contains the following DD names:

- SUMMARY is required and specifies where the output summarizing the trace is directed.
- DETAILS is required only for storage analysis and RU counting. It specifies where details of the trace analysis are directed.

- LOG is required. It specifies where VIT entries with possible errors are directed.
- OUTSTAN is required only for storage analysis of outstanding entries. It specifies where the list of outstanding GBLK, VTAL, and REQS entries is directed.
- VITEXT is required only for VIT extraction. It specifies where the VIT entries extracted from the trace are directed.

  Note: Only the FORMAT and NOFORMAT output options should be used when directing VITEXT output to a printer as shown in Figure 54 on page 412. VITEXT output using TRACEFORMAT should be directed to disk or tape.

  For VITEXT output using TRACEFORMAT, the record length must be the length of the TRACE record or 284, whichever is smaller.
- PARM is required and specifies the parameters to be passed to the VIT analysis tool. Parameters can be specified in-stream (in the JCL) or in a dataset; do not use the PARM parameter on the EXEC statement for this purpose because of size restrictions. The PARM dataset must have fixed records (can be blocked) with:
  - LRECL=80
  - RECFM=FB
- TRACE is required and specifies the input dataset containing the trace to be processed. The TRACE DCB information must match the actual data set characteristics. The record format can be V, VB, or VBA.

Interactive Mode: As an alternative to running in batch mode, you may invoke the following routine to run the VIT analysis tool interactively. If you choose this method for processing, your terminal will be unavailable until processing is completed.

```
1. /*REXX*/
2. /*********************************************************************/
3. /*  Run the VIT analysis tool interactively.                         */
4. /*                                                                   */
5. /*  Tailor the data set names and other ALLOC options as needed for*/
6. /*  each run.                                                        */
7. /*                                                                   */
8. /*********************************************************************/
9.
10.'ALLOC DD(PARM)     DSN(run1.parm) SHR'
11.'ALLOC DD(TRACE)    DSN(run1.trace) SHR'
12.'ALLOC DD(SUMMARY)  DSN(run1.summary) OLD'
13.'ALLOC DD(DETAILS)  DSN(run1.details) OLD'
14.'ALLOC DD(OUTSTAN)  DSN(run1.outstan) OLD'
15.'ALLOC DD(LOG)      DSN(run1.log)     OLD'
16.'ALLOC DD(VITEXT)   DSN(run1.vitext)  OLD'
17.
18."CALL 'SYS1.SISTDBUG(ISTRAFT1)'"
19.
20.'FREE DD(SUMMARY,DETAILS,OUTSTAN,LOG,VITEXT,PARM,TRACE)'
```

Figure 55. Sample VIT Analysis Tool Interactive Routine

The lines in Figure 55 are:

Line	Description
1	Required for a REXX EXEC
2–8	Comments
10 - 16	Data set allocations. The data sets must be pre-allocated.
18	Invokes the VIT analysis tool, assuming that it has been installed in the SYS1.SISTDBUG load library.
20	Frees the datasets allocated to the DD name statements to allow the exec to run again with different data set names.

Note: The DD parameters are required. The DSN parameters are optional and can be varied.

Warning

1. Return codes are not checked in this example.
2. Your user terminal will not be available while the tool is active.
3. This example is not shipped with the VTAM code and is included for information only.

Issue the appropriate FILEDEFs (VM)

Issue the FILEDEFs from the command line or use the sample REXX exec shown in Figure 56.

Note: MVS GTF trace tapes can be processed on VM as long as the VTAM trace records are generated from an MVS-equivalent VTAM.

```
1. /* REXX */
2.   'GLOBAL LOADLIB ISTDEBUG'    /* Define ISTRAFT1 for OSRUN     */
3. /*****************************************************************/
4. /* Define DD names.                                              */
5. /*****************************************************************/
6.   'FILEDEF PARM    DISK INPUT    PARM  A (PERM RECFM F LRECL 80'
7.   'FILEDEF SUMMARY DISK SUMMARY  OUT   A (PERM RECFM V LRECL 84'
8.   'FILEDEF DETAILS DISK DETAILS  OUT   A (PERM RECFM V LRECL 84'
9.   'FILEDEF OUTSTAN DISK OUTSTAN  OUT   A (PERM RECFM V LRECL 130'
10.  'FILEDEF LOG     DISK LOG      OUT   A (PERM RECFM V LRECL 124'
11.  'FILEDEF VITEXT  DISK VITEXT   OUT   A (PERM RECFM V LRECL 32756'
12.  'FILEDEF TRACE   DISK TRACEDAT GTOUT A (PERM RECFM V  LRECL 284'
```

Figure 56. Sample REXX Exec for VIT Analysis Tool VM

The lines in Figure 56 are:

Line	Description
1	Required for a REXX EXEC
2	Set up the loadlib containing ISTRAFT1.
3–5	Comments
6 - 12	CMS file definitions. For a description see page 412.

Warning

1. Return codes are not checked in this example.
2. This example is not shipped with the VTAM code and is included for information only.

Step 3. Create the Parameters for the Job

To create the parameters needed to analyze your VIT:

1. Use the panel interface. On the VTAM Internal Trace Analysis panel, specify the form of processing you want to use. Only one function may be used in a session. The panel interface then provides choices and help in specifying values for the parameters, and creates the PARM dataset.

   Pick option 1, 2, or, 3 from the Figure 57 ISTT0001 and follow the processing path until you return to the panel ISTT0001. Then choose option 4 to indicate that your input has been completed.

```
ISTT0001              VTAM Internal Trace Analysis

Select a choice, then press Enter.

 __ 1.  Storage Analysis
    2.  Request/response unit counting
    3.  VIT extraction
    4.  Input Complete









(C) Copyright IBM Corporation 1994.  All rights reserved.
 Command ===> ____________________________________________________
:
```

Figure 57. VTAM Internal Trace Analysis Option Panel

   Note: If an incorrect value is entered in a field, the cursor appears on the field where the error was made. For help about that field, press F1.

2. Use an editor to either:
   - Create a parameter data set
   - MVS Code the parameters in-stream in the JCL created in step 2.

   See the parameter syntax for a particular function, and "Creating Your Own Parameter Dataset" on page 447 for further details.

   See "How to Read the Syntax Diagrams" on page 238 for general information on how to code and read syntax diagrams.

Step 4. Run the Job

Submit the dataset for processing that you created in step 2. You may process it in batch mode MVS or interactively.

VM Issue the command

```
OSRUN ISTRAFT1
```

to invoke the VIT analysis tool, which should have previously been installed in the ISTDEBUG load library.

Step 5. Check the Output

After processing is complete, check the following data sets for your results:

For RU counting and storage analysis reports, check the DETAILS data set
For the extracted VIT entries, check the VITEXT data set
For unmatched storage allocation entries, check the OUTSTAN data set.

If you do not get the expected output, check the SUMMARY and LOG datasets for error messages or other information on what might have caused the problem. For example, the SUMMARY dataset contains the parameters used for the job, including the parameters specified and the defaults taken.

If the trace has wrapped, indicate this on the I/O Options panel or use the WRAP parameter. If the GTF trace tapes were specified in the wrong order, correct the order of the tapes in the JCL. In either case, submit the job again.

Return codes

If an error has occurred and the SUMMARY data set is available, a message will be written in the data set. The return codes are:

```
 0    No errors found
 4    Counter overflow     -   processing continues, if possible
 8    Storage unavailable  -   processing continues, if possible
10    I/O failure          -   processing continues, if possible
12    Unrecoverable error  -   processing stops
```

Environment

- The VTAM V4 formatted trace and the VIT analysis tool cannot process data created by earlier VTAM releases because of changes to output formats.
- Because of the way that MVS GTF and VM TRSOURCE handles entries that continue to multiple records, VTAM can only assume that the continued records are contiguous, and matching the continuation record to the prior record cannot be guaranteed.
- If you get trace information that is out of sequence, the trace may have wrapped. If the trace wrapped when it was recorded, specify WRAP on the VIT analysis tool.
- The existing VTAM formatted trace provides trace record formatting by splitting up all VIT entries into logical pieces and adding labels to indicate what the data represents. This function will not be replaced, and you can still format the VIT using the MVS IPCS subcommands VERBEXIT VTAMMAP or GTFTRACE and VM TRACERED .

Analyzing Storage

Use storage analysis to count storage allocated and freed, match related SMS entries, and report potential storage concerns found in a VIT. The main panel for storage analysis is shown in Figure 58. Select an option and follow the prompts.

```
ISTT006A              Storage Analysis - Panel 1 of 3

Select. Then press Enter.

Matching . . . . . 2   1.  Match allocates, frees
                       2.  None

Unmatched allocates  2   1.  List unmatched entries
                       2.  Do not list entries

Storage lengths  . . 2   1.  List storage lengths
                       2.  Do not list lengths









Command ===> _____________________________________________________________
```

Figure 58. VTAM Storage Analysis Option Panel

Matching

You can choose to match allocate entries with free entries. Depending upon which areas of storage are selected on subsequent panels, some or all of the following VIT entries are matched:

- GBLK and FBLK
- VTAL and VTFR
- REQS and QREQ
- REQS and RELS
- REQS and AREL.

The default is no matching.

Unmatched Allocates

You can choose to list all unmatched (outstanding) storage allocates found. By default, they are ignored. There may be a large number of unmatched storage allocates and the output may be very long. Unmatched allocates can occur if the VIT is not complete. Entries listed are not necessarily error conditions.

Storage Lengths

You can choose to list the storage lengths (number of bytes or buffers) requested, allocated, and freed. By default, storage lengths are not listed.

GBLK Pools, VTAL Pools, and REQS Buffer Pools

You can designate which SMS VIT entries to process. By default, all GBLK pools, VTAL pools, and REQS buffer pools are processed. Choices are available to process some or none of these pools. If you choose to process some, a panel is displayed from which you can choose the specific pools to process. For GBLK and VTAL, your choices include listing only the storage pools allocated from private storage or only the storage pools allocated from CSA (by default, both private and CSA storage pools are listed).

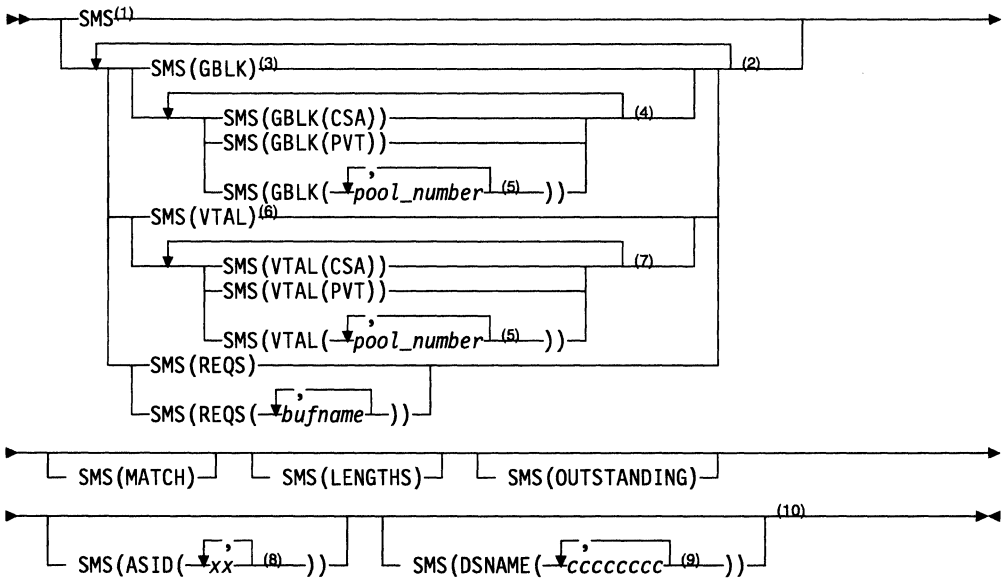
Address Space Identifiers (ASIDs) and Data Spaces

You can limit storage analysis to specific ASIDs (VM machine IDs and task IDs) and data spaces. By default, all ASIDs and data spaces are processed. If one or more ASIDs or data spaces are specified, only SMS entries associated with each of the specified ASIDs or data spaces are processed.

Note: MVS Data spaces only apply to GBLK entries.

Parameter Syntax

Use the following syntax only if you are using an editor to write your parameters as an alternative to the panel interface. See "How to Read the Syntax Diagrams" on page 238.



Notes:

1. Coding SMS has the same effect as coding SMS(GBLK) SMS(VTAL) SMS(REQS)
2. You can code one or more from each choice group below.
3. Coding SMS(GBLK) has the same effect as coding SMS(GBLK(PVT)) SMS(GBLK(CSA))
4. You can code one or more from each choice group below.
5. Code one or more *pool_number*s in the range 0–255. Separate each *pool_number* from the next with a comma.
6. Coding SMS(VTAL) has the same effect as coding SMS(VTAL(PVT)) SMS(VTAL(CSA))
7. You can code one or more from each choice group below.
8. Replace the two *x*s with two hex characters. You can code up to 5 ASIDs (VM machine IDs and task IDs) .
9. Replace the eight *c*s with eight alphanumeric characters. MVS You can code up to five DSNAMEs.
10. Data is not case sensitive.

Sample Output for Storage Analysis

In this example, the following PARM file was submitted:

```
sms sms(lengths)  sms(match)  sms(outstanding)
stoptime(02:04:10)  noformat
DESC This example shows storage analysis with a stop time.
```

The resulting Summary, Detailed, and Outstanding data sets follow.

Notes:

1. Ellipses indicate that part of the output has been omitted.
2. STOPTIME is described in "Using the Timing Options" on page 442.
3. NOFORMAT and DESC are described in "Using the I/O Options" on page 445.
4. General parameter coding information is described in "Creating Your Own Parameter Dataset" on page 447.

The following is an example of a Summary report for storage analysis, which is written to the data set name that you specify to receive the report. The DD name for the data set must be SUMMARY. The first line of all reports shows the VTAM level, the 20-character title (the default is "Trace Analysis "), and the date that the report was processed. The description, if specified, follows the title.

The next several lines of the report are the specified parameters and the defaults. These are followed by observations which are included to highlight important properties of the trace that were encountered during processing. The observations will vary depending on the trace content and the options specified. The option choices are storage analysis, RU counting, and VIT extraction.

The trace statistics section contains the first and last time stamps, the record type count, the VIT entry count, and the VIT option count. For example, there may be 6 PIU VIT entries and 18 PIU2 entries, for a total of 24 PIU VIT option occurrences. Some VIT entries are not associated with a VIT option. If entries of this type are found, the total is listed beside "N/A" in the VIT option counts. If SNAP entries are found, the total is listed beside "?" in the VIT option counts.

```
VTAM V4     Trace Analysis       Summary            92.325 11/20 11:28:47 LOC

This example shows storage analysis with a stop time.

Wrapped:     No
Formatting:  No
Interval:    None
Start time:  Beginning of trace
Stop time:   91.199 07/18 02:04:10.000 LOC (A43572C3 51A80000)
SMS:         Yes
   ASIDs:    All
   Options:  MATCH LENGTHS OUTSTANDING
   GBLK:     All
   DS Pools: All
   DSNAMEs:  All
   VTAL:     All
   REQS:     All
RU:          No
VITEXT:      No

*******************************************************************************
Observations

Only 27 GBLK entries were found in this VIT.  The findings should be verified
on a larger trace.

Only 7 VTAL entries were found in this VIT.  The findings should be verified on
a larger trace.

The high water mark is 392 bytes allocated by VTAL.  This occurred at
02:04:07.715 LOC (record 115).

Only 18 REQS entries were found in this VIT.  The findings should be verified
on a larger trace.
```

```
********************************************************************************
Trace Statistics

First GTF Timestamp: 91.199 07/18 02:03:13.266 LOC (A435728D 36BFAE01)
First VIT Timestamp: 91.199 07/18 02:03:13.266 LOC (A435728D 36BFAE01)

Last VIT Timestamp:  91.199 07/18 02:04:07.716 LOC (A43572C1 240D7002)
Last GTF Timestamp:  91.199 07/18 02:04:10.866 LOC (A43572C4 2529DC02)

Summary of GTF Record Types

                  1 Timestamp control records
                116 VIT records
                    ----------------
                117 Total GTF records

Count of VIT Entry and Option Occurrences

     VIT Entry Occurrences
                 33 FBLK              27 GBLK
                  6 PIU               18 PIU2
                 19 RELS              18 REQS
                  7 VTAL               5 VTFR

     VIT Option Occurrences
                 24 PIU
                109 SMS
                    -----
                133 Total
```

The following is an example of a Detailed Report for storage analysis, which is written to the data set name that you specify to receive the report. The DD name for the data set must be DETAILS. In this example, the GBLK pool totals follow the title and description. The totals are listed first for each get block (GBLK) pool in the "Home" data space, then for each pool in other data spaces, if any exist. For each pool, the pool number is shown in decimal and hexadecimal, followed by the pool name, an indication of allocation from private storage or CSA, and the data space name.

Following the counts of total entries, matches, bytes allocated and freed, and other entries, you will find a summary number for the allocate and freed entries in this pool for each storage size. This section only written if the LENGTHS option is specified. In this example, there were 9 GBLK entries, each of which allocated 152 bytes and requested 152 bytes. The largest number of GBLK entries not matched to a free block (FBLK) at one time was 5. There were 9 FBLK entries, each of which freed 152 bytes.

Following the GBLK pool counts for each pool are the totals for all GBLK pools. Next, the VTAL pools are listed like the GBLK pools and finally the REQS are listed.

```
VTAM V4     Trace Analysis       Detailed Report    92.194 07/12 18:51:44 LOC

This example shows storage analysis with a stop time.

GBLK Pool Totals:

Home Data Space:

GBLK Pool 0 (X'00'): RUPEPRIV (Private)  Data space: Home
                   9 GBLK entries (including failures, if any)
                   9 FBLK entries (including failures, if any)
                   5 GBLK and FBLK matches
                   5 Largest number of GBLK entries at one time
               1,368 Bytes allocated
               1,368 Bytes allocated above the 16M line
               1,368 Bytes requested
               1,368 Bytes requested above the 16M line
               1,368 Bytes freed
               1,368 Bytes freed above the 16M line
                 608 Bytes not freed
                 760 Most unfreed bytes

         Bytes          Bytes          GBLK       Maximum          FBLK
     Allocated      Requested       Entries      Requested       Entries
           152            152             9             5             9
⋮
GBLK Totals:
                  27 GBLK entries (including failures, if any)
                  33 FBLK entries (including failures, if any)
                  16 GBLK and FBLK matches
                  13 Largest number of GBLK entries at one time
                   5 Largest number of GBLK entries at one time in one pool
                     was in pool 0 (X'00') in Home data space
               4,192 Bytes allocated
               3,696 Bytes allocated in private storage
                 496 Bytes allocated in CSA
               4,192 Bytes allocated above the 16M line
               3,375 Bytes requested
               2,888 Bytes requested in private storage
                 487 Bytes requested in CSA
               3,375 Bytes requested above the 16M line
               3,672 Bytes freed
               3,304 Bytes freed in private storage
                 368 Bytes freed in CSA
               3,672 Bytes freed above the 16M line
               2,160 Bytes not freed
               1,880 Bytes not freed in private storage
                 280 Bytes not freed in CSA
               2,824 Most unfreed bytes
                     was at 02:04:07.715 LOC at record 116
               1,536 Most unfreed bytes in one pool
                     was in pool 32 (X'20') in Home data space

                      ****************************************
```

```
VTAL Subpool Totals:

Subpool 13 (X'0D'): Private
                    3 VTAL entries (including failures, if any)
                    2 VTFR entries (including failures, if any)
                    2 VTAL and VTFR matches
                    1 Largest number of VTAL entries at one time
                  288 Bytes allocated
                  288 Bytes allocated above the 16M line
                  192 Bytes freed
                  192 Bytes freed above the 16M line
                   96 Bytes not freed
                   96 Most unfreed bytes

         Bytes           VTAL      Most VTAL          VTFR
     Allocated        Entries        Entries       Entries
            96              3              1             2
⋮
VTAL Totals:
                    7 VTAL entries (including failures, if any)
                    5 VTFR entries (including failures, if any)
                    5 VTAL and VTFR matches
                    2 Largest number of VTAL entries at one time
                    1 Largest number of VTAL entries at one time in one pool
                      was in pool 13 (X'0D')
                1,344 Bytes allocated
                  768 Bytes allocated in private storage
                  576 Bytes allocated in CSA
                1,344 Bytes allocated above the 16M line
                  952 Bytes freed
                  672 Bytes freed in private storage
                  280 Bytes freed in CSA
                  952 Bytes freed above the 16M line
                  392 Bytes not freed
                   96 Bytes not freed in private storage
                  296 Bytes not freed in CSA
                  392 Most unfreed bytes
                      was at 02:04:07.715 LOC at record 115
                  368 Most unfreed bytes in private
                  296 Most unfreed bytes in CSA
                  368 Most unfreed bytes in one pool
                      was in pool 47 (X'2F')

                       ************************************
```

```
REQS Totals by Buffer Pool:

REQS Buffer Pool IOBUF
                  5 REQS entries (including failures, if any)
                  7 RELS entries (including failures, if any)
                  4 RELS entries were matched
                  5 Buffers allocated
                  5 Buffers allocated above the 16M line
                  4 Buffers freed
                  4 Buffers freed above the 16M line
                  1 Buffers not freed
                  2 Largest number of buffers at one time

    Buffers per          REQS      Most REQS
        Request       Entries        Entries
              1             5              2
⋮
REQS Totals:
                 18 REQS entries (including failures, if any)
                 19 RELS entries (including failures, if any)
                 15 RELS entries were matched
                 18 Buffers allocated
                 18 Buffers allocated above the 16M line
                 15 Buffers freed
                 15 Buffers freed above the 16M line
                  3 Buffers not freed
                  5 Largest number of buffers at one time
                    was at 02:03:44.609 LOC at record 71
                  3 Largest number of buffers at one time in one pool
                    was in the LPBUF pool
```

The following is an example of an Outstanding Report for storage analysis, which is written to the data set name that you specify to receive the report. The DD name for the data set must be OUTSTAN. Following the title and description, the unmatched GBLK entries are listed by pool number for each data space. Next the unmatched VTAL entries are listed by subpool number then the unmatched REQS and REQ2 entries are listed by buffer name.

Notes:

1. The REQ2 entries consist of the VIT entry name (REQ2) followed by 1 to 7 addresses. Instead of showing the actual REQ2 entry, which does not contain the data shown on the REQS entry, each buffer address is shown as if it had appeared in a REQS entry.
2. The queued REQS entries which have not been matched by a QREQ entry are also listed in the OUTSTAN data set. These REQS have a buffer address of zero.

```
VTAM V4     Trace Analysis       Outstanding Report 92.194 07/12 18:51:44 LOC

This example shows storage analysis with a stop time.

*******************************************************************************
* List of outstanding GBLK entries                                            *
*******************************************************************************

Home Data Space:

GBLK Pool 0 (X'00'): RUPEPRIV (Private)  Data space: Home

Outstanding GBLK at 02:03:52.888 LOC (record 89)
C7C2D3D2 0C000000 06638480 06357218 823D6340 00000098 864EBCF8 00000098 *GBLK......d.....b.. ...qf+.8...q*

Outstanding GBLK at 02:04:01.276 LOC (record 102)
C7C2D3D2 0C000000 066388E0 06357218 823D6340 00000098 864EBCF8 00000098 *GBLK......h\....b.. ...qf+.8...q*

Outstanding GBLK at 02:04:01.276 LOC (record 105)
C7C2D3D2 0C000000 066385C0 06357218 823D6340 00000098 8644B2B8 00000098 *GBLK......e ....b.. ...qf......q*

Outstanding GBLK at 02:04:03.374 LOC (record 108)
C7C2D3D2 0C000000 06638340 06357218 823D6340 00000098 864EBCF8 00000098 *GBLK......c ....b.. ...qf+.8...q*
⋮
*******************************************************************************
* List of outstanding VTAL entries                                            *
*******************************************************************************

Subpool 13 (X'0D'): Private

Outstanding VTAL at 02:04:05.615 LOC (record 112)
E5E3C1D3 0C000000 067A3FA0 0000000D 82478930 00000060 00000000 00000000 *VTAL.....:. ....b.i....-...

-------------------------------------------------------------

Subpool 15 (X'0F'): Private

Every valid VTAL in this pool was matched by a VTFR.

-------------------------------------------------------------
⋮
*******************************************************************************
* List of outstanding REQS entries                                            *
* (Note: Each buffer from a REQ2 entry is listed as a separate REQS)          *
*******************************************************************************

REQS Buffer Pool IOBUF

REQS waiting for QREQ or RELS at 02:03:25.703 LOC (record 35)
D9C5D8E2 0C000000 062EAE88 06321010 823B95F4 00010000 062DD648 00000000 *REQS.......h....b.n4......O.....*

-------------------------------------------------------------
⋮
```

Counting Request/Response Units (RUs)

Use RU counting to list the number of each kind of RU found in a PIU. Because there are so many RUs, RU counting lets you specify which RUs you are interested in. The main panel for counting request/response units is shown in Figure 59. Select an option and follow the prompts.

```
ISTT010A                Request/Response Unit Counting

Select, then press Enter.

Requests/Responses  3   1.  Request units only
                        2.  Response units only
                        3.  Both

RUs . . . . . . . . 1   1.  All
                        2.  Some - type codes
                        3.  Some - list names
                        4.  Some - list codes

Network addresses . _   All
                    _   From and/or to one address
                    _   Between two addresses
                    _   From one address to another

Sort order  . . . . 1   1.  Name
                        2.  Frequency
```

Figure 59. VTAM Request/Response Unit Counting

Requests/Responses

You can choose to process only request units, only response units, or both. Both are processed by default.

RUs

You can choose to process only specific RUs. Subsequent panels allow you to specify particular RUs by typing the codes, by picking the RUs from a list of RUs by name, or by picking the RUs from a list of RUs by code. By default, all RUs are processed.

Network Addresses

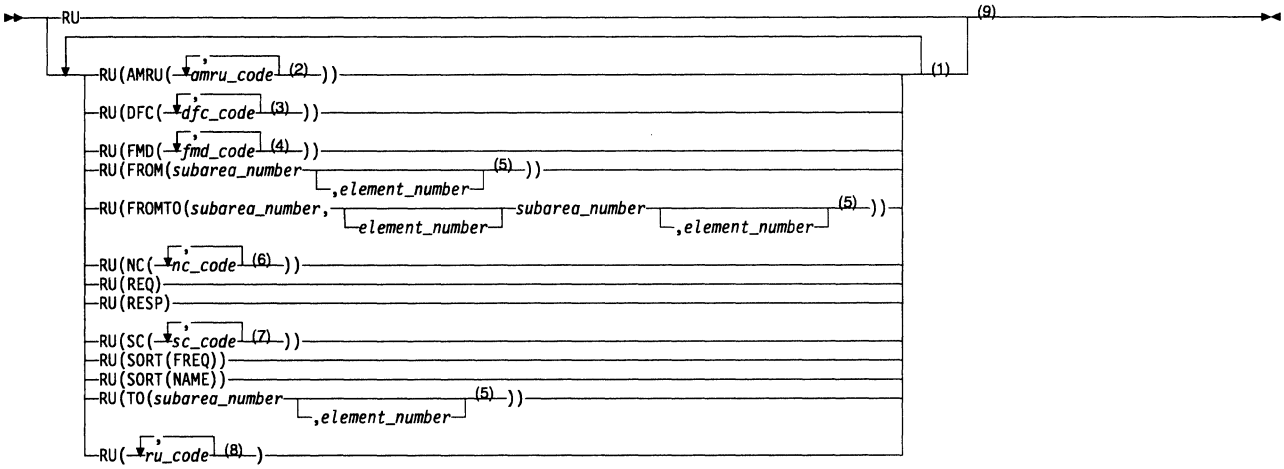
You can also specify processing of all PIUs found in the VIT regardless of the origin and destination, only PIUs from one network address to another in a single direction, all PIUs between two network addresses in both directions, or all PIUs to and from one network address.

Sorting Order

You can sort the RU counts either alphabetically (by RU name) or by frequency (in descending order of counts). The default is sorting by RU name.

Parameter Syntax

Use the following syntax only if you are using an editor to write your parameters as an alternative to the panel interface. See "How to Read the Syntax Diagrams" on page 238.



Notes:

1. You can code one or more of these options. Unless otherwise noted, code each option no more than once.
2. Each *amru_code* is four or eight hex characters. You can code up to 32 *amru_codes*.
3. Each *dfc_code* is two or four hex characters. You can code up to 32 *dfc_codes*.
4. Each *fmd_code* is six or eight hex characters. You can code up to 32 *fmd_codes*.
5. Each *subarea_number* is eight hex characters, and each *element number* is four hex characters.
6. Each *nc_code* is two or four hex characters. You can code up to 32 *nc_codes*.
7. Each *sc_code* is two or four hex characters. You can code up to 32 *sc_codes*.
8. Each *ru_code* is two, four, six, or eight hex characters. You can code up to 32 ru_codes.
9. Data is not case sensitive.

RU Parameter Coding

RU means count all request/response unit codes.

RU(NC(...)), **RU(SC(...))**, **RU(DFC(...))**, **RU(FMD(...))**, or **RU(AMRU(...))** means count only the RUs in categories which have the specified codes as follows:

- NC, SC, and DFC codes are 1 or 2 bytes[7]:
  - SNA codes are 1 byte
  - AMRU codes are 2 bytes—the first byte is X'FF'.
- FMD codes are 3 or 4 bytes[7]:
  - SNA codes are 3 bytes
  - AMRU codes are 4 bytes—the third byte is X'FF'.

You must fully specify all codes (the VIT analysis tool does not pad codes).

---

7 Two hexadecimal digits represent one byte. For example, X'FF' is one byte. For RU counting, you must specify all hexadecimal digits and you must omit the X and quotes. For example, RU(NC(X'C')) is not valid; RU(NC(0C)) is valid.

RU(...) means process only the RUs with those codes. Each code is 1, 2, 3, or 4 bytes. The VIT Analysis Tool counts these codes in any and all categories. For example, X'0D' is the NC code for NC-ACTVR and the SC code for ACTLU. If you specify **RU(0D)**, the VIT analysis tool will count both of these RU codes.

You can specify up to:

32 codes for **RU(NC(...))**,
32 codes for **RU(SC(...))**,
32 codes for **RU(DFC(...))**,
32 codes for **RU(FMD(...))**,
32 codes for **RU(AMRU(...))**, and
32 codes for **RU(...)**.

You can specify the RU codes individually or in lists. For example,

```
RU(SC)31,32,FF31,FF32))
```

is equivalent to

```
RU(SC(31)) RU(SC(32)) RU(SC(FF31)) RU(SC(FF32))
```

RU(REQ) means process requests. **RU(RESP)** means process responses. By default (if you specify neither **RU(REQ)** nor **RU(RESP)**), the VIT analysis tool counts both requests and responses.

RU(FROM(...)) means count RUs flowing from the specified network address to any network address. **RU(TO(...))** means count RUs flowing to the specified network address from any network address. **RU(FROMTO(...))** means count RUs flowing from the first network address to the second network address. You can specify up to:

24 **RU(FROM(...))** options,

24 **RU(TO(...))** options, and

24 **RU(FROMTO(...))** options.

Each subarea field (*subarea_number*) is exactly 4 bytes [7]. Each element field (*element_number*) is exactly 2 bytes. The element fields are optional. For the **RU(FROMTO(...))** option, if you omit the first element address, keep its comma as a place-holder.

Note: You must specify the **RU(FROM(...))**, **RU(TO(...))**, and **RU(FROMTO(...))** options individually. (You cannot combine them in lists like RU codes.)

By default (if you specify neither **RU(FROM(...))**, **RU(TO(...))**, nor **RU(FROMTO(...))**), the VIT analysis tool ignores the origin and destination fields in the PIUs.

RU(SORT(NAME)) means sort the request and response counts by RU name (the default). **RU(SORT(FREQ))** means sort the counts by frequency (highest count first).

Combinations of RU options

You can combine any or all of the RU options. When considering combinations of options, you might find it helpful to think of the RU options as two groups, as shown in Figure 60 and Figure 61.

```
RU
RU(NC(...))
RU(SC(...))
RU(DFC(...))
RU(FMD(...))
RU(AMRU(...))
RU(...)
```

Figure 60. RU Code Options. These specify which RU codes to count.

```
RU(REQ)
RU(RESP)
RU(FROM(...))
RU(TO(...))
RU(FROMTO(...))
RU(SORT(NAME))
RU(SORT(FREQ))
```

Figure 61. RU Modify Options. These options indicate whether to count request or response units, which origins and destinations to count, and the sort order for the count.

The options in Figure 61 modify the options in Figure 60. For example, given the following combination of options,

```
RU(REQ) RU
```

the VIT analysis tool will count all requests for all RU codes.

If you specify any modify option and no code option, the VIT analysis tool uses the default, which is the **RU** option. In other words, the VIT analysis tool counts all RU codes which match the modify options. The VIT Analysis Tool prints a message in the SUMMARY data set so you will know the **RU** option is in effect.

The **RU** option (which counts all RUs) overrides the other code options (Figure 60). The VIT Analysis Tool prints a message in the SUMMARY data set so you will know the override is in effect. For example, given the following combination of options

```
RU RU(NC(04))
```

the VIT analysis tool counts all RUs and does not check whether the specified RU, NC(04) in the example is found in the VIT.

If any of the **RU(FROM(...))**, **RU(TO(...))**, or **RU(FROMTO(...))** options match the origin or destination of an RU, the VIT analysis tool will count the RU. For example, you can ask for the counts of the following:

- RUs from subarea X'00000012'
- RUs to subarea X'00000012'
- RUs flowing between subarea X'0000004A' element X'000C' and subarea X'00000002'.

by coding

```
RU(FROM(00000012)) RU(TO(00000012))
RU(FROMTO(0000004A,000C,00000002)) RU(FROMTO(00000002,,0000004A,000C))
```

If you list specific RUs (i.e., you specify any options in Figure 60 except **RU**), the VIT analysis tool does not count:

- User RUs
- FMH RUs
- Unknown RUs.

(Although the VIT analysis tool reports the first occurrence of an unknown RU in the LOG data set.)

Sample Output for RU Counting

In this example, the following PARM file was submitted:

```
Desc This example shows request/response unit counting
Desc with a start time and a 30-second interval.
RU STARTIME(02:04:30) interval(00:30)
```

The resulting DETAILS, LOG, and SUMMARY datasets follow.

Note: Ellipses indicate that part of the output has been omitted. The following is an example of a Summary report for RU counting, which is written to the data set name that you specify to receive the report. The DD name for the data set must be SUMMARY.

This report is similar to the Summary report for storage analysis. See "Sample Output for Storage Analysis" on page 419 for details.

```
VTAM V4     Trace Analysis       Summary           92.325 11/20 11:47:26 LOC

This example shows request/response unit counting
with a start time and a 30-second interval.

Wrapped:     No
Formatting:  Yes
Interval:    00:30
Start time:  91.199 07/18 02:04:30.000 LOC (A43572D6 64780000)
Stop time:   End of trace
SMS:         No
RU:          Yes
   Options:  Requests Responses
   Sort:     Name
   Codes:    All
   From:     All
   To:       All
   From/To:  All
VITEXT:      No

********************************************************************************
Observations

There are 7 messages in the LOG file.

Only 250 RUs were found in this trace.  The findings should be verified on a
larger trace.

7 responses had sense data included.  See the LOG data set.

********************************************************************************
Trace Statistics

First GTF Timestamp: 91.199 07/18 02:03:13.266 LOC (A435728D 36BFAE01)
First VIT Timestamp: 91.199 07/18 02:04:32.920 LOC (A43572D9 2D8DDE02)

Last VIT Timestamp:  91.199 07/18 02:06:00.891 LOC (A435732D 12B57F02)
Last GTF Timestamp:  91.199 07/18 02:06:00.891 LOC (A435732D 12B57F02)
```

Sample Output for RU Counting

```
Summary of GTF Record Types

                15 Timestamp control records
             6,930 VIT records
                   -----------------
             6,945 Total GTF records

Count of VIT Entry and Option Occurrences

     VIT Entry Occurrences
             1,390 FBLK              6 FBL2
             1,385 GBLK              6 GBL2
               104 MSG             104 MSGS
               217 MSG2            262 PIU
               746 PIU2          1,427 RELS
             1,414 REQS              6 REQ2
               415 VTAL            411 VTFR

     VIT Option Occurrences
               425 MSG
             1,008 PIU
             6,460 SMS
                   -----
             7,893 Total
```

The following is an example of a Detailed Report for RU counting, which is written to the data set name that you specify to receive the report. The DD name for the data set must be DETAILS. The RU counts for each interval are written following the title and description. The total count for all intervals is written at the end of the report.

Under Requests, only the first GDS variable in each RU is counted. Under Responses, all RUs with sense data included are grouped together. Each RU with sense data included is printed in the LOG data set. For example, an UNBIND response with sense data included is counted only as a response with sense data included, not as an UNBIND response.

```
VTAM V4     Trace Analysis       Detailed Report    92.194 07/12 18:51:46 LOC

This example shows request/response unit counting
with a start time and a 30-second interval.

********************************************************************************
Interval 1

First VIT timestamp in this interval:
91.199 07/18 02:04:32.920 LOC (A43572D9 2D8DDE02) (record 390)
********************************************************************************
RU Totals for Interval 1:

Requests:
               1 RNAA
                              --------------
                            1 Total requests

Responses:
               1 FNA                                  1 RNAA
                              ---------------
                            2 Total responses

********************************************************************************
Last VIT timestamp in interval 1
91.199 07/18 02:05:01.277 LOC (A43572F4 3877DE01) (record 429)
********************************************************************************
```

```
Interval 2
:
RU Totals:

Requests:
                2 ACTLINK                          7 BFCINIT
                4 BFCLEANUP                        7 BFINIT
                2 BFSESSEND                        5 BFSESSST
                5 BFTERM                          18 BIND
                1 BINDF                            4 CINIT
                5 CLEANUP                          2 CONNOUT
                2 CONTACT                          2 CONTACTED
                1 DACTLINK                         1 DISCONTACT
                5 FMH-5 Attach-CP Capab            6 FMH-5 Attach-TDU
                9 FNA                              4 GBIND BIND
                4 GDS CP Capabilities              7 GUNBIND
                3 INIT-OTHER                       1 INOP
                1 NOTIFY (SSCP<-->LU)              3 RECMS
                2 REQCONT                         11 RNAA
                5 SESSEND                          5 SESSST
               23 UNBIND
                              --------------
                          157 Total requests

Responses:
                2 ACTLINK                          7 BFCINIT
                1 BFCLEANUP                        5 BIND
                4 CINIT                            1 CLEANUP
                2 CONNOUT                          2 CONTACT
                1 DACTLINK                         1 DISCONTACT
               10 FNA                              4 INIT-OTHER
                1 NOTIFY (SSCP<-->LU)             11 RNAA
                7 Sense Data Included             13 User
               21 UNBIND
                              --------------
                           93 Total responses
```

Sample Output for RU Counting

The following is an example of an RU counting log, which is written to the data set name that you specify to receive the report. The DD name for the data set must be LOG. The Log contains important details found in the VIT during processing. For RU processing, the Log contains all RUs with included sense data.

```
VTAM V4     Trace Analysis       Log              92.194 07/12 18:51:46 LOC

This example shows request/response unit counting
with a start time and a 30-second interval.

Sense data included at 02:05:56.653 LOC (record 1,990)
    Origin:           00000004 0073
    Destination:      00000001 0008
    Response Header:  EF9000   Session Control
    Sense Data:       80050000 Path error
                               No session
    Rejected RU code: 32
    Rejected Command: UNBIND

Sense data included at 02:05:56.655 LOC (record 2,000)
    Origin:           00000004 0073
    Destination:      00000001 0008
    Response Header:  EF9000   Session Control
    Sense Data:       80050000 Path error
                               No session
    Rejected RU code: 32
    Rejected Command: UNBIND

Sense data included at 02:06:00.372 LOC (record 6,706)
    Origin:           00000001 0008
    Destination:      00000001 0001
    Response Header:  8F9000   Function Management Data
    Sense Data:       08160000 Request reject
                               Function already inactive
    Rejected RU code: 810629
    Rejected Command: CLEANUP

Sense data included at 02:06:00.399 LOC (record 6,725)
    Origin:           00000004 0000
    Destination:      00000001 0001
    Response Header:  8F9000   Function Management Data
    Sense Data:       081E0001 Request reject
                               Session reference error
    Rejected RU code: 812629
    Rejected Command: BFCLEANUP

Sense data included at 02:06:00.456 LOC (record 6,774)
    Origin:           00000001 0008
    Destination:      00000001 0001
    Response Header:  8F9000   Function Management Data
    Sense Data:       08160000 Request reject
                               Function already inactive
    Rejected RU code: 810629
    Rejected Command: CLEANUP
⋮
```

Extracting Information from the VIT

Use VIT extraction to extract entries from a VIT. VIT entries extracted from a VIT can be formatted, displayed in hex with the eye-catcher, or copied in the same format as the input.

Upon selecting VIT Extraction, the VIT Extraction Boolean Expression panel is displayed as shown in Figure 62.

```
ISTT0018              VIT Extraction Boolean Expression

Type a Boolean expression or press F4 to use the template.

_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________

 Operands                    Description                Operators
 CCcc or E'CCcc'             Option or entry name       () Delimiters
 A'xxxxxxxX':nn              Address:offset             ¬  Not
 C'Ccccc...':nn              Char String:offset         -  Through
 X'XXxxx...':nn              Hex String:offset          &  And
 O'xxxx...' or D'xxxx...'    Origin or Destination      |  Or
Command ===> _____________________________________________________________
:
```

Figure 62. VIT Extraction Boolean Expression panel

The first time VIT extraction is invoked, the VIT extraction Boolean expression panel is blank. After the first time, the panel is displayed with the previously entered Boolean expression.

You may specify the VIT entries you want extracted by entering a Boolean expression on this panel, or by filling in a template (one or more times). See "Using the Template" for information on how to use the template. Both methods result in a Boolean expression that specifies the criteria used to select VIT entries. VIT entries which contain the data specified by the Boolean expression are extracted. Extracted VIT entries may be formatted, displayed in hex with the eye-catcher or copied as is. See "Using the I/O Options" on page 445 for information on how to code these options.

Using the Template

To use the template, press F4. The VIT Extraction Template is displayed as shown in Figure 63 on page 435.

All fields are optional. Any explicitly specified VIT entries, and VIT entries created by a specified option, are eligible for extraction if found in the trace. Only those eligible entries that meet all other specified selection criteria are extracted.

```
ISTT0014                VIT Extraction Template

Type information in one or more fields, then press Enter. This information
will be appended to the full expression.

VIT options/entries _____ _____ _____ _____ _____ +
                    _____ _____ _____ _____ _____

Address . . . . . __________ (Hexadecimal)
Offset  . . . . . ____ (Decimal or Hexadecimal)

Character string  . ____________________________
Offset  . . . . . ____ (Decimal or Hexadecimal)

Hexadecimal string ____________________________
                   ____________________________
Offset  . . . . . ____ (Decimal or Hexadecimal)



Command ===> ____________________________________________________________
:
```

Figure 63. VIT Extraction Template

The fields on the template are described below. Fill in the template and press Enter.

The template is checked for proper data type and length of data and saved. The resulting Boolean expression is then added to the VIT Extraction Boolean Expression panel. You can append multiple instances of the template, and a VIT entry that matches any of the templates will be extracted (when VIT extraction is invoked). The length of the resulting expression is limited to the input area on the VIT Extraction Boolean Expression panel.

Press F3 to exit, and you are given the option to save the expression you have created.

The fields on the extraction template are described as follows:

VIT options/entries

You may specify VIT options or entries to limit extraction to particular VIT entries. If you specify a VIT option, VIT entries created when the designated options are active are eligible for extraction if found in the trace. For example, the LOCK option generates the LKEX, LKSH, ULKA, and UNLK trace entries. If you specify LOCK and the LOCK option was used when the VIT was started, any LKEX, LKSH, ULKA, and UNLK entries found in the trace are eligible for extraction.

Note: User-defined (SNAP) entries are allowed. See "SNAP Trace Record" on page 408.

You may also specify particular VIT entries (for example, LKEX or LKSH), and an asterisk may be used in a VIT entry name to match any character (for example, LK* matches VIT entries LKEX and LKSH). If an entry with multiple parts is specified, all parts are extracted (for example, if AI2 is specified, AI1, AI2, and AI3 are extracted).

If no VIT options or entries are specified, all VIT entries are eligible for extraction.

Address Entries with this address are eligible for extraction. Specify up to 8 hex characters for an address. If fewer than 8 digits are specified, the address is padded on the left with zeros. The low-order 31 bits of the address are then used to compare on all word boundaries if an offset is not specified. If an offset is specified, then only the offset is checked for a match. Address X'00000000' is allowed only when an offset for the address is also provided.

Address offset
The offset for an address is a word boundary offset into a trace record where a comparison should be made for the address. The offset must be one of the following:

X'04', 4
X'08', 8
X'0C', 12
X'10', 16
X'14', 20
X'18', 24
X'1C', 28

Character string
Entries containing this character string are eligible for extraction. Enter a search string of 1–30 alphanumeric characters, which includes special characters (except a single quote). Data entered is not converted to uppercase. Data is case sensitive. By default, a comparison is made at all offsets.

Character string offset
You may include a byte offset into the VIT entries where comparisons should be made. The offset can be decimal or hex. The offset range is 2–31 or X'02'–X'1F'.

Note: The length of the character string determines the offset allowed. For example, if the character string entered consists of 8 characters, the valid offset range is 2–24. The string could not possibly start after byte 24, since a VIT entry is 32 bytes long (byte 0 through byte 31) and 8 bytes are needed to represent the string (bytes 24–31).

Hexadecimal string
Entries containing this hex string are eligible for extraction. Enter a search string of 2–60 hex characters representing 1–30 bytes of data. By default, a comparison is made at all offsets.

If you code an odd number of hex digits, they are padded to the left with a zero to make one byte. For example; X'D' is equivalent to X'0D'.

Hexadecimal string offset
You may include an offset into the VIT entries where comparisons should be made. The offset can be decimal or hex. The offset range is 2–31 or X'02'–X'1F'.

Note: The length of the hex string divided by 2 determines the offset allowed, since two hex digits represent one byte. For example, if the hex string entered consists of 8 hex digits, 4 bytes are needed to store

the string and the valid offset range is 2–28. The string could not possibly start after byte 28, since a VIT entry is 32 bytes long (byte 0 through byte 31) and 4 bytes are needed to represent the string (bytes 28–31).

Important Restriction	Character and hex strings will not be found if they cross VIT entry boundaries. If a PIU is represented in the VIT as a PIU entry plus a series of PIU2 entries and a string is requested that spans the end of one PIU2 and the beginning of a second PIU2, it will not be found.

Creating a Boolean Expression without the Template

The template allows you to create many Boolean expressions, but there is no way to:

- Negate an operand
- Group operands
- Specify AND or OR
- Specify an operand multiple times on one template
- Specify a range.

To do any of this, you will need to enter an expression directly on the VIT Extraction Boolean Expression panel. Blanks are recommended between all operands and operators. Again, the length of the expression is limited to the input area provided. No syntax error checking is performed.

A sample free-form expression follows:

```
AI* | (LOCK | E'PIU' & X'31':15 |
GBL* & A'476C' - A'4800')
```

This example selects each VIT entry that is either AI1, AI2, or AI3, or fulfills at least one of the following criteria:

- LOCK option group (LKEX, LKSH, ULKA, UNLK)
- PIU entry with value hex 31 at offset 15
- GBLK or GBL2 entry with any address from X'0000476C' through X'00004800'.

Operands: Following are the operands used in creating a Boolean expression. For further information on these operands, see "Using the Template" on page 434.

VIT options or entries
VIT entries with names identical to option names must be prefaced with an E and enclosed in single quotes (for example, E'MSG'). This distinguishes the name as an entry rather than an option. A VIT option or unique entry name requires no preface.

Address Preface an address with an A and enclose it in single quotes (for example, A'xxxxxxxx', where xxxxxxxx is the hex address).

Address offset
Specify the offset for an address after the address string, and separate the address string and the offset with a colon (for example, A'xxxxxxxx':nn, where xxxxxxxx is the hex address and nn is the offset).

Note: Only decimal offsets are allowed.

Character string
Preface a character string with a C and enclose it in single quotes (for example, C'ccccc...', where ccccc... represents the character string). Data is case sensitive.

Character string offset
Specify the offset for a character string after the character string, and separate the character string and the offset with a colon (for example, C'ccccc...':nn, where ccccc... is the character string and nn is the offset).

Note: Only decimal offsets are allowed.

Hexadecimal string
Preface a hex string with an X and enclose it in single quotes (for example, X'xxxxx...', where xxxxx... represents the hex string).

Hex string offset
Specify the offset for a hex string after the hex string, and separate the hex string and the offset with a colon (for example, X'xxxxx...':nn, where xxxxx... is the hex string and nn is the offset).

Note: Only decimal offsets are allowed.

Origin To extract PIU entries originating at a designated network address, enter 1–12 hex characters (representing the 6-byte network address) in the form O'xx...'.

Note: The address is right aligned and padded with zeros on the left. For example, O'3001A' means subarea 3 element 1A.

Destination
To extract PIU entries destined for a particular network address, enter 1–12 hex characters (representing the 6-byte network address) in the form D'xx...'.

Note: The address is right aligned and padded with zeros on the left. For example, D'4E' means subarea 0 element 4E.

All Specify ALL to extract all VIT entries. ALL is not valid with any other operand or operator.

Note: You may also extract all VIT entries without entering the ALL operand. For example, 'gblk | ¬ gblk' will extract all VIT entries.

Operators: The operators used in creating a Boolean expression are shown in Figure 64 on page 439.

Operator	Description
()	Parentheses
¬	Not
-	Through
&	And
|	Or

Figure 64. Boolean Expression Operators in Order of Precedence

Parentheses have the highest precedence and can be used to change the normal order of evaluation. The maximum nesting level is 15. The *through* operator (a hyphen) specifies a range and can be used for addresses or a hex string in the following combinations:

- address-address
- address:offset-address
- hex string-hex string
- hex string:offset-hex string

If an offset is specified and you are using the *through* operator, the offset on the first operand is used for both operands.

Parameter Syntax

Use the following syntax only if you are using an editor to write your parameters as an alternative to the panel interface.

```
►►—VITEXT Boolean_expression————————————————————————————————►◄
```

Note: Up to 15 VITEXT parameters may be coded for longer Boolean expressions.

VITEXT must be the first six characters and must be followed by a blank. The rest of the line is assumed to be the expression.

Sample Output for VIT Extraction

In this example, the following PARM file was submitted:

```
desc This example shows VIT extraction.
desc All PIU VIT entries to or from network address 000000040073
desc and all MSGs with the string ACTIVE will be extracted.
vitext o'40073' | d'40073' | (MSG & c'ACTIVE')
noformat
```

The resulting VITEXT and SUMMARY datasets follow.

Note: Ellipses indicate that part of the output has been omitted. The following is an example of a Summary report for VIT extraction, which is written to the data set name that you specify to receive the report. The DD name for the data set must be SUMMARY.

This report is similar to the Summary report for storage analysis. See "Sample Output for Storage Analysis" on page 419 for details.

```
VTAM V4     Trace Analysis       Summary           92.325 11/20 11:47:45 LOC

This example shows VIT extraction.
All PIU VIT entries to or from network address 000000040073
and all MSGs with the string ACTIVE will be extracted.

Wrapped:     No
Formatting:  No
Interval:    None
Start time:  Beginning of trace
Stop time:   End of trace
SMS:         No
RU:          No
VITEXT:      Yes
       o'40073' | d'40073' | (MSG & c'ACTIVE')

********************************************************************************
Observations

25 GTF VIT records were extracted and written to VITEXT.

********************************************************************************
Trace Statistics

First GTF Timestamp: 91.199 07/18 02:03:13.266 LOC (A435728D 36BFAE01)
First VIT Timestamp: 91.199 07/18 02:03:13.266 LOC (A435728D 36BFAE01)

Last VIT Timestamp:  91.199 07/18 02:06:00.891 LOC (A435732D 12B57F02)
Last GTF Timestamp:  91.199 07/18 02:06:00.891 LOC (A435732D 12B57F02)

Summary of GTF Record Types

               16 Timestamp control records
            7,318 VIT records
                  ----------------
            7,334 Total GTF records

Count of VIT Entry and Option Occurrences

     VIT Entry Occurrences
              1,504 FBLK              6 FBL2
              1,498 GBLK              6 GBL2
                110 MSG             110 MSGS
                230 MSG2            279 PIU
                795 PIU2          1,464 RELS
              1,451 REQS              6 REQ2
                443 VTAL            439 VTFR

     VIT Option Occurrences
                450 MSG
              1,074 PIU
              6,817 SMS
                    -----
              8,341 Total
```

Sample Output for VIT Extraction

The following is an example of a VIT Selections report, which is written to the data set name that you specify to receive the report. The DD name for the data set must be VITEXT. This report contains the VIT entries selected by the Boolean expression.

The 'date' line containing the date, time, and record number for the first record extracted is written following the title and description. The title and description are not written if the TRACEFORMAT option is selected. The 'date' line is written at the beginning of the report and when the date changes. The time is written to the left of each record.

In this example, each VIT entry is written on one line and each line contains the time, the entry in hexadecimal, and the EBCDIC translation of the entry.

```
VTAM V4     Trace Analysis       VIT Selections     92.194 07/12 18:51:50 LOC

This example shows VIT extraction.
All PIU VIT entries to or from network address 000000040073
and all MSGs with the string ACTIVE will be extracted.

*** DATE *** 91.199 07/18 02:05:56.137 LOC (A4357328 8A0BB802) (record 881)
02:05:56.137 D7C9E440 0C990000 06321010 40007870 20000037 00000001 00000004 1D000008 *PIU .r..... ...............*
02:05:56.137 D7C9E4F2 00730018 00816B80 00310013 07B0B050 B33F8797 97870706 02000000 *PIU2.....a,........&..gppg......*
02:05:56.137 D7C9E4F2 00000000 00230000 04C1F0F2 D51F0008 02C3D7E2 E5C3D4C7 090300E3 *PIU2.........A02N....CPSVCMG...T*
02:05:56.137 D7C9E4F2 F9560FFA BBA50A04 D5C5E3C1 4BC1F0F2 D50004C1 F0F1D50E 0AF3D5C5 *PIU29....v..NETA.A02N..A01N..3NE*
02:05:56.137 D7C9E4F2 E3C14BC1 F0F2D50E 0AF4D5C5 E3C14BC1 F0F2D52C 0A010840 40404040 *PIU2TA.A02N..4NETA.A02N....     *
02:05:56.137 D7C9E4F2 40404060 12E7E3F9 560FFABB A509D5C5 E3C14BC1 F0F2D500 00000000 *PIU2   -.XT9....v.NETA.A02N.....*
02:05:56.137 D7C9E440 0C990000 06321010 40007870 20000037 00000001 00000004 1D000008 *PIU .r..... ...............*
02:05:56.137 D7C9E4F2 00730018 00816B80 00310013 07B0B050 B33F8797 97870706 02000000 *PIU2.....a,........&..gppg......*
02:05:56.137 D7C9E4F2 00000000 00230000 04C1F0F2 D51F0008 02C3D7E2 E5C3D4C7 090300E3 *PIU2.........A02N....CPSVCMG...T*
02:05:56.137 D7C9E4F2 F9560FFA BBA50A04 D5C5E3C1 4BC1F0F2 D50004C1 F0F1D50E 0AF3D5C5 *PIU29....v..NETA.A02N..A01N..3NE*
02:05:56.137 D7C9E4F2 E3C14BC1 F0F2D50E 0AF4D5C5 E3C14BC1 F0F2D52C 0A010840 40404040 *PIU2TA.A02N..4NETA.A02N....     *
02:05:56.137 D7C9E4F2 40404060 12E7E3F9 560FFABB A509D5C5 E3C14BC1 F0F2D500 00000000 *PIU2   -.XT9....v.NETA.A02N.....*
⋮
02:05:56.712 D4E2C7E2 0C000000 00000000 0000F0F1 0039E000 C9E2E3F1 F0F5C940 C1F0F4D7 *MSGS.........01..\.IST105I A04P*
02:05:56.712 D4E2C7F2 E4C3C1F3 40D5D6C4 C540D5D6 E640C9D5 C1C3E3C9 E5C56B40 D5D6C4C5 *MSG2UCA3 NODE NOW INACTIVE, NODE*
02:05:56.722 D4E2C7E2 0C000000 00000000 0000F0F1 0038E000 C9E2E3F1 F0F5C940 C1F0F4D3 *MSGS.........01..\.IST105I A04L*
02:05:56.722 D4E2C7F2 D5C3C1F3 40D5D6C4 C540D5D6 E640C9D5 C1C3E3C9 E5C56B40 D5D6C4C5 *MSG2NCA3 NODE NOW INACTIVE, NODE*
02:05:56.804 D4E2C7E2 0C000000 00000000 0000F0F1 002BE000 C9E2E3F0 F9F3C940 C1F0F4E2 *MSGS.........01..\.IST093I A04S*
02:05:56.804 D4E2C7F2 F1F640C1 C3E3C9E5 C56B40D5 D6C4C540 E3E8D7C5 407E40D3 C9D5C508 *MSG216 ACTIVE, NODE TYPE = LINE.*
02:05:56.883 D4E2C7E2 0C000000 00000000 0000F0F1 002FE000 C9E2E3F0 F9F3C940 C1F0F4D7 *MSGS.........01..\.IST093I A04P*
02:05:56.883 D4E2C7F2 F1F6F140 C1C3E3C9 E5C56B40 D5D6C4C5 40E3E8D7 C5407E40 D7E46DE3 *MSG2161 ACTIVE, NODE TYPE = PU_T*
02:05:57.239 D4E2C7E2 0C000000 00000000 0000F0F1 002DE000 C9E2E3F0 F9F3C940 C1F0F4D3 *MSGS.........01..\.IST093I A04L*
02:05:57.239 D4E2C7F2 D5C3C1F3 40C1C3E3 C9E5C56B 40D5D6C4 C540E3E8 D7C5407E 40D3C9D5 *MSG2NCA3 ACTIVE, NODE TYPE = LIN*
02:05:57.306 D4E2C7E2 0C000000 00000000 0000F0F1 0030E000 C9E2E3F0 F9F3C940 C1F0F4D7 *MSGS.........01..\.IST093I A04P*
02:05:57.306 D4E2C7F2 E4C3C1F3 40C1C3E3 C9E5C56B 40D5D6C4 C540E3E8 D7C5407E 40D7E46D *MSG2UCA3 ACTIVE, NODE TYPE = PU_*
⋮
```

Using the Timing Options

After completing storage analysis, RU count, or VIT extraction, the timing options panel is automatically displayed. Use the timing options to report at certain intervals in the VIT or to process only the VIT records within a certain time range. By default, the entire VIT is processed. All time values, including time stamps, are local (LOC) time. The main panel for storage analysis is shown in Figure 65. Select an option and follow the prompts.

```
ISTT0190                        Timing Options

Select and type. Then press Enter.

Start 1   1.  Beginning of trace
          2.  At timestamp (hex TOD clock)
          3.  At date and time


Stop  1   1.  End of trace
          2.  At timestamp (hex TOD clock)
          3.  At date and time


Report interval _____  (MM:SS) (Storage Analysis and RU Counting only)



Command ===> ____________________________________________________________
```

Figure 65. VTAM Timing Options Panel

Note: Report interval, start times, and stop times might not produce the expected output if the VIT was extracted from a dump using the VITAL option. VIT entries in a dump do not have individual time stamps.

Start and Stop Time

Start and stop times can be at a time stamp within the trace, at a date and time within the trace, or at the beginning of the trace for start and at the end for stop. If you do not specify a start or stop time, the entire VIT is processed.

If you select 'At timestamp', another panel appears on which you may enter a System 370 time-of-day (TOD) time stamp. To ensure that the time stamp reflects your local time, you must add the time zone value to the high-order word. The time zone can be obtained from a dump, if the trace being used has been extracted from it, or by browsing the first time-stamp record from the trace in hexadecimal format. For example:

```
  A905470D237491E4   GMT TOD on a GTF trace record
+ FFFFBCF100000000   Time zone (padded with zeros)
  ----------------
  A90503FE237491E4   Local TOD
```

You can enter either the high-order 4 bytes of the time stamp, such as `X'A90503FE'` or all 8 bytes, such as `X'A90503FE237491E4'`.

Note: All hexadecimal time stamps reported by the VIT analysis tool are local time stamps. (The time zone has already been added.)

If you select 'At date and time', another panel appears on which you can enter both date and time selections.

The time can be

hh:mm:ss

or

hh:mm:ss.ddd.

The date can be

Calendar format (*mm/dd/yy*)

or

Julian format (*yy.ddd*).

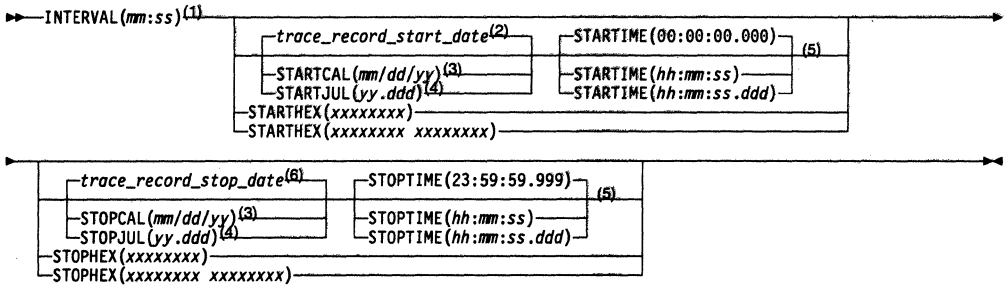
Report Interval

If you select report interval, a report is written for each time interval determined from your selection. For example , if you have a storage analysis trace that was recorded for a 2 hour period and you specify an interval of 5 minutes, you get a report of the storage allocated and freed for every 5 minute period of the trace. A total for the whole 2 hours is printed regardless of the interval specified.

Note: Interval is only valid for storage analysis and RU counting.

Parameter Syntax

Use the following syntax only if you are using an editor to write your parameters as an alternative to the panel interface. See "How to Read the Syntax Diagrams" on page 238.



Notes:

1. INTERVAL cannot be used with VIT extraction. INTERVAL can be used with any start or stop time specification. Intervals (*mm:ss*) are interpreted as 00 through 99 for minutes, and 00 through 59 for seconds.
2. If you do not code a start date, the VIT analysis tool uses the date on the first trace record.
3. The calendar date (*mm/dd/yy*) is 01 through 12 for month, 01 through 31 for day, and 00 through 99 for year. For dates, *yy* values 43–99 are interpreted as 1943–1999, and *yy* values 00–42 are interpreted as 2000–2042.
4. The Julian date (*yy.ddd*) is 00 through 99 for year and 001 through 366 for day. For dates, *yy* values 43–99 are interpreted as 1943–1999, and *yy* values 00–42 are interpreted as 2000–2042.
5. Calendar time (*hh:mm:ss.ddd*) is 00 through 23 for hour, 00 through 59 for minutes, 00 through 59 for seconds, and 000 through 999 for milliseconds.
6. If you do not code a stop date, the VIT analysis tool uses the date on the first trace record.

Using the I/O Options

The I/O Options panel is displayed automatically after the Timing Options panel. Use the I/O options to designate a wrapped trace, whether or not you want formatted output, or to specify a title or description for the first page of a report. The main panel for the I/O options is shown in Figure 66. Select an option and follow the prompts.

```
ISTT0022                    Input/Output Options

Type, then press Enter.

Trace wrapped? 2   1.  Trace wrapped
                   2.  Trace did not wrap

Format output? 1   1.  Format the VIT entries
                   2.  Do not format
                   3.  Create trace data set

Title . . . . . Trace Analysis

                              Description

   ____________________________________________________________
   ____________________________________________________________
   ____________________________________________________________
   ____________________________________________________________

Command ===> ____________________________________________________
```

Figure 66. VTAM I/O Options Panel

Trace Wrapped

Select *Trace wrapped* if the trace wrapped when it was recorded. Wrapped means that the trace ran out of space on the specified device and began to write over previously recorded data. A trace recorded on a DASD device is usually large enough to wrap. A trace recorded on a tape will not wrap.

Format Output

VIT entries can be written formatted, unformatted or as a hexadecimal string. Here is a GBLK entry in each format:

- **Do not format** (parameter syntax: **NOFORMAT**). Shows the 32 byte VIT entry as 8 words in hexadecimal format, followed by the same 32 bytes as the EBCDIC eye-catcher. For example,

  ```
  C7C2D3D2 0C000000 06638480 06357218 823D6340 00000098 864EBCF8 00000098 *GBLK.....d.....b.. ...qf+.8...q*
  ```

- **Format the VIT entries** (parameter syntax **FORMAT**). Shows the 32 byte VIT entry with labels for each field. For example,

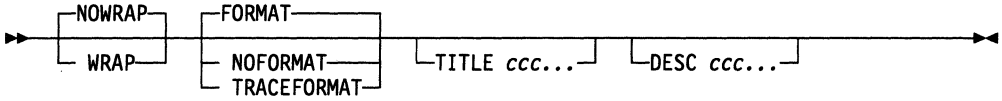
  ```
  GBLK        ASID 0C   RC   00      ID   00      VTA  00      AREA 06638480 ANCH 06357218
              ISSR 823D6340          LEN  00000098 INIT 864EBCF8 RLEN 00000098
  ```

- **Create trace data set** (parameter syntax **TRACEFORMAT**). The 24-byte GTF header followed by the 32-byte entry in hexadecimal format. This output is not suitable for printing, but is usable as input to the VIT analysis tool or other tools. For example,

  ```
  ..u..e5et....7..VTAMTST GBLK.....d.....b.. ...qf+.8...q
  ```

Parameter Syntax

Use the following syntax only if you are using an editor to write your parameters as an alternative to the panel interface.



Operands

Notes:

1. MVS If FORMAT is specified and the VTAM format module is not found or is unusable, a message is issued and the job is stopped. To recover, do either of the following and rerun the job.
   - Find the current version of AMDUSRFD and add a STEPLIB DD statement to your application
   - Specify NOFORMAT to print the VIT entries in hexadecimal with an eye-catcher.
2. The TRACEFORMAT option is valid only for VIT extraction, and results in output in the same format as that of the TRACE input dataset.
3. If you specify the TRACEFORMAT option, the output from the VIT analysis may be processed by:
   - The VIT analysis tool
   - ACF/TAP
   - IPCS GTFTRACE, unless the VIT was recorded internally and was extracted from a dump by the MVS VTAMMAP VITAL function.
4. The 20 characters following the word TITLE are used as the title. The rest of the input line is ignored.
5. DESC may contain up to 75 additional characters, and you can code up to 4 DESC parameters. DESC must be the only option on the line and must be the first 4 characters on the line followed by at least one blank.

Creating Your Own Parameter Dataset

The PARM dataset must have fixed records (can be blocked) with:

- LRECL=80
- RECFM=FB

The parameters can be coded in any order, and in lowercase, uppercase, or mixed case. Code only *one* function parameter (SMS, RU, or VITEXT) per job or execution. See the parameter syntax for each trace function for a list of the possible parameters that may be coded.

Restriction	The parameters shown in Figure 67 can be coded in parts to avoid exceeding the maximum line length of 80 characters, but a single parameter cannot be continued on the next line.

```
SMS(GBLK(0,1,...255))
SMS(VTAL(0,1,...255))
SMS(REQS(bufname,...))
DESC ccc...
RU(NC(xx,xx,...xx))
RU(SC(xx,xx,...xx))
RU(DFC(xx,xx,...xx))
RU(FMD(xxxxxx,xxxxxx,...xxxxxx))
RU(AMRU(xxxx,...xxxxxxxx))
RU(xx,...xxxxxxxx))
VITEXT Boolean expression
```

Figure 67. Parameters Coded on Multiple Lines

For example, SMS(GBLK(0,1,2,3)) can be split onto multiple lines as follows:

```
SMS(GBLK(0))
SMS(GBLK(1))
SMS(GBLK(2))
SMS(GBLK(3))
```

Notes:

1. If you use an editor, you may include comments in your parameter data set. An asterisk in column one identifies a line as a comment line. If you use the panel interface, you cannot enter comment lines.
2. You may use the DEBUG option to gather information to solve problems with the tool itself. To use it, enter DEBUG with one of the VIT analysis options.
3. Lines can not be continued. Each parameter must be fully specified on one line.

Sample Parameter Data Set

Following is a sample parameter data set.

```
SMS(VTAL) sms(match) SMS(Lengths)
INTERVAL(00:15) STARTIME(12:42:14) STOPTIME(14:00:00)
DESC This is an analysis of the VTAL and VTFR VIT entries.
DESC This will tell the high-water mark (the most storage used)
DESC from the start time to the stop time.
DESC This job will also report the storage used in 15-second intervals.
```

See "Analyzing Storage" on page 417, "Counting Request/Response Units (RUs)" on page 426, or "Extracting Information from the VIT" on page 434 for additional explanation of the parameters.

Also see "Using the Timing Options" on page 442 and "Using the I/O Options" on page 445.

Chapter 11. Using VSCS Service Aids (VM)

The following tools can help you diagnose VSCS problems:

- Messages, see "Message and Module Conventions"
- Operator console listing, see "Operator Console Listing" on page 450
- Operator commands: BFRFIFO, CHANGE, QUERY, DISPLAY, FORCE, and STORAGE, see "Operator Commands" on page 450
- VSCS dump, see "VSCS Dump" on page 453
- VSCS trace facility, see "VSCS Trace Facility" on page 458.

These service aids, along with service aids for VTAM, can help you isolate problems to a single component and identify problems within VSCS.

Message and Module Conventions

For most messages, the module name appears as the first word in the message text. For example, in the following message DTIC01I, DTICINIT is the module name.

```
DTIC01I DTICINIT VSCS IUCV INITIALIZATION IS IN PROGRESS
```

All VSCS message and module identifiers begin with DTI. Module names are in the form DTI*caaaa*, where *aaaa* is the unique module identifier and *c* represents the issuing subcomponent, as follows:

C Communication services for inter-user communication vehicle (IUCV)
I Initialization and termination services
P Presentation services
  In this subcomponent (DTIP*aaaa*), the fifth letter is one of the following:
  D Device manager
  F Force processing
  I Input manager
  O Output manager
  R Request services
  S State manager
S System services
V VTAM services
  In this subcomponent, a module with the last character of X (DTIV*aaa*X) indicates a VTAM exit routine.
X System services (messages only)

Message identifiers are in the form DTI*cnni*, where *c* indicates the issuing component (as above), *nn* is the two-digit serial number, and *i* is the action indicator:

A Action
E Eventual action
I Information

For internal VSCS errors, DTIMSGE macro requests issue the following message, where *modname* indicates the name of the module that issued the message, and *n* identifies the *n*th occurrence of the DTIMSGE macroinstruction within the module:

```
DTIS99I INTERNAL VSCS ERROR modname n [netid.luname]
```


The number *n* can be used to identify the reason the message was issued. See "Message Problems: Message DTIS99I Indicates a VSCS Internal Error" on page 176 and *VTAM Messages and Codes* for more information.

Operator Console Listing

The operator console listing is a printout of the messages that appear at the VSCS operator console. The listing contains everything that appears on the operator's console during the entire session. It can help you find out the sequence of events when a problem occurs. If VSCS is running in the same virtual machine as VTAM, the console listing includes VTAM messages as well as VSCS messages.

To obtain a console listing, issue the CP command SPOOL CONSOLE START. If you are using a hard-copy terminal, such as a 3767, this is not necessary because all input and output already comes out on paper.

Note: If the operator console is in CONMODE 3270, the CP SPOOL CONSOLE command produces no output.

To determine what book describes CP operator commands for your operating system, see Table 80 on page 1165.

Operator Commands

The following commands are described in more detail in Chapter 4, "VSCS Operator Commands (VM)" in *VTAM Operation*. For a description of the messages in the output examples, see *VTAM Messages and Codes*.

BFRFIFO Command

BFRFIFO affects the dequeueing and dispatching structure for storage. Generally, storage is used from the top in a last-in-first-out fashion (BFRLIFO). This is the default. For diagnostic purposes, however, you can turn on BFRFIFO. This first-in-first-out method of storage preserves the storage element queue long enough for you to take a dump and see the storage elements that have just been processed.

Note: The VSCS STORAGE command will show the current queuing method in use.

To start BFRFIFO, use the VSCS BFRFIFO command or the DTIGEN macroinstruction. For more information on the DTIGEN macroinstruction see Chapter 5, "VM/SNA Console Support (VSCS)" in the *VTAM Resource Definition Reference*.

Note: Using BFRFIFO can seriously degrade system performance, especially in a large network, and prevents the storage manager check and release functions from executing or functioning.

CHANGE Command

The CHANGE command enables you to modify various DTIGEN start options without having to halt and restart VSCS. When you use the CHANGE command to change DTIGEN start options, the changes are in effect only while VSCS is active. With the CHANGE command, you can affect VSCS' use of:

- Storage when sending or receiving data for a logical unit
- System dump facilities during repetitive dump situations

- User exit control.

For more information on the CHANGE command, see "VSCS CHANGE Command" in *VTAM Operation*. For information on the DTIGEN start options, see "VSCS Start Options" in the *VTAM Network Implementation Guide*.

DISPLAY Command

- The VSCS command DISPLAY ID=*netid.luname* shows the status of a single logical unit.
- The VSCS command DISPLAY ID=*luname* shows the status of all the logical units with *luname*, regardless of their network IDs.

Figure 68 shows example output from the VSCS DISPLAY command with *netid* specified.

```
vscs d id=a01a721
Ready;
DTIS60I DISPLAY COMMAND ACCEPTED
DTIS61I LU NAME = NETA.A01A721 DEVICE TYPE = 3278 MODEL = 3
DTIS62I LU TYPE = 0 DEVICE FEATURES = EDS APL( NEW ) COLOR
DTIS63I SCREEN SIZE = 24 BY 80 ALTERNATE SIZE = 32 BY 80 PACE = 32
DTIS65I IUCV PATHIDS = 2 5 CURRENT RPL COMMAND = SEND RPL IS FREE
DTIS66I CURRENT STATES: PS = NMH VS = NORM DR
DTIS67I LOGON TIME = 15 : 25 : 16
DTIS68I SEND COUNT = 00000005 RECEIVE COUNT = 00000001
DTIS74I TRACE TABLE STARTS AT 002C8010 END 002C8FEF , CURRENT 002C8010 , SIZE 126
DTIS77I VSCS COMMAND PROCESSING COMPLETE
```

Figure 68. Example of the VSCS DISPLAY Command Output

DUMPLU Command

The VSCS DUMPLU command enables the operator to obtain documentation for one or more VSCS logical units without having to dump the entire VSCS virtual machine. A partial dump is taken of the VSCS control blocks and work elements associated with each logical unit specified. If *netid* is not specified on the command, then all logical units with the specified name will be dumped. See "VSCS DUMPLU Command" in *VTAM Operation* for information on DUMPLU.

FORCE Command

The VSCS command FORCE ID=*[netid.]luname* terminates the CP session and VTAM session for a logical unit. If *netid* is not specified on the command, it will default to the host network ID. This command may be used when the CP FORCE and VTAM VARY INACT commands do not terminate a logical unit.

QUERY Command

The VSCS QUERY command displays the following information:

- The location of the VSCS trace table (if it has been allocated)
- The status of VSCS traces
- The status of a VSCS dump
- The status of buffer queues.

Figure 69 on page 452 shows an example of the messages issued in response to the VSCS QUERY command.

```
vscs query
Ready;
DTIS74I TRACE TABLE STARTS AT 00299020 END 002A0D5F , CURRENT 00299880 , SIZE 2000
DTIS82I NON-SELECTIVE CCS, VTAM TRACE ACTIVE
DTIS97I INDIVIDUAL LU TRACE TABLES ENABLED, SIZE = 120
DTIS93I VSCS EXTERNAL TRACE IS DISABLED
DTIS78I BUFFER DEQUEUE IS IN NORMAL MODE(BFRLIFO)
DTIS77I VSCS COMMAND PROCESSING COMPLETE
```

Figure 69. Example of the VSCS QUERY Command Output

STORAGE Command

The VSCS STORAGE command shows the status of VSCS dynamic storage and static storage and some DTIGEN options that are involved with VSCS dynamic storage and static storage usage. Figure 70 shows an example of the output that results from the VSCS STORAGE command when both STCHKTM and STRELTM are zero. For more information on the DTIGEN macroinstruction, see Chapter 5, "VM/SNA Console Support (VSCS)" in the *VTAM Resource Definition Reference*.

```
vscs storage
Ready;
DTIS15I DYNAMIC STORAGE UTILIZATION
DTIS16I RCVBFRL -    284 DPXMTL   -   1268  KPXMTL  -    284
DTIS17I RPLNUM  -    255 VSAMLM   -     10  VEIBS   -     15
DTIS18I BLKMULT -      1 STCHKTM  -      0  STRELTM -      0
DTIS19I GETMAIN -     43 FREEMAIN -      1  SWAP    -      0
DTIS20I VTAMSES -      1 CPCONNS  -      1  CPMAX   -      1
DTIS11I SPECIAL -      0 SPGET    -      0  SPFREE  -      0
DTIS12I BADSIZE -      0 BADCOUNT -      0
DTIS50I STATIC STORAGE UTILIZATION
DTIS23I PATHTAB -   1608 LOGO     -   2520  VTAMWA  -  29040
DTIS24I PRINTER -      0 K                  TRACE   -   2000 K
DTIS77I VSCS COMMAND PROCESSING COMPLETE
```

Figure 70. Example of the VSCS STORAGE Command, STCHKTM=0 and STRELTM=0

Figure 71 shows an example of the VSCS STORAGE command when either STCHKTM or STRELTM is greater than zero.

```
vscs storage
Ready;
DTIS15I DYNAMIC STORAGE UTILIZATION
DTIS16I RCVBFRL -    284 DPXMTL   -   1268  KPXMTL  -    284
DTIS17I RPLNUM  -    255 VSAMLM   -     10  VEIBS   -     15
DTIS18I BLKMULT -      1 STCHKTM  -      1  STRELTM -     10
DTIS19I GETMAIN -     43 FREEMAIN -      1  SWAP    -      0
DTIS20I VTAMSES -      1 CPCONNS  -      1  CPMAX   -      1
DTIS11I SPECIAL -      0 SPGET    -      0  SPFREE  -      0
DTIS12I BADSIZE -      0 BADCOUNT -      0
DTIS14I DYNAMIC STORAGE IN USE
DTIS21I POOL   HIGHWATER/VTAMSES    CURRENT    AVAILABLE     LIFO
DTIS22I  PS          10 /      1         10          8  TOTAL=     56 K
DTIS22I  VS          26 /      1         26          1  TOTAL=    104 K
DTIS22I  VX           5 /      1          5          3  TOTAL=     20 K
DTIS14I DYNAMIC STORAGE NOT IN USE
DTIS21I POOL   HIGHWATER/VTAMSES    CURRENT    AVAILABLE     LIFO
DTIS22I  S1           0 /      0          0          0  TOTAL=      0 K
DTIS22I  S2           0 /      0          0          0  TOTAL=      0 K
DTIS22I  S3           0 /      0          0          0  TOTAL=      0 K
DTIS50I STATIC STORAGE UTILIZATION
DTIS23I PATHTAB -   1608 LOGO     -   2520  VTAMWA  -  29040
DTIS24I PRINTER -      0 K                  TRACE   -   2000 K
DTIS77I VSCS COMMAND PROCESSING COMPLETE
```

Figure 71. Example of the VSCS STORAGE Command, STCHKTM>0 or STRELTM>0

VSCS Dump

The VSCS dump contains all of the storage for VSCS. It also contains common, group control system (GCS), and VTAM storage if they are in the same virtual machine as VSCS.

While the dump is being taken, all network activity temporarily stops in order to keep the contents of the dump intact. When the dump is requested, VSCS issues a message containing the name of the module requesting the dump and a dump identifier number. VSCS dumps are numbered to correlate to the console messages.

Note: If you get a VSCS dump using the DUMPLU command, only LU data is complete. The dump is not numbered.

Use the link-edit listing to locate key modules or work areas in the dump. The addresses of CSECTs DTIISDA1 and DTIISDA2 also appear in the trace table header record, as shown in Figure 74 on page 461. DTIISDA1 and DTIISDA2 contain the global control blocks, which can be identified in the dump by their unique identifiers. For more information on control block identifiers, see "VSCS Storage and Control Block ID Codes (VM)" on page 1139.

For more information on controlling dumps, see "Controlling VSCS Dump Points" in *VTAM Customization*.

Dump Formatting Routine

VSCS provides a dump formatter to format VSCS control blocks and the VSCS internal trace table in a dump. The routine runs as part of PRTDUMP. To produce formatted dump output, do the following:

1. Use the GCS GDUMP or CP VMDUMP command to dump VSCS storage.
2. Use DUMPLOAD to process the reader file produced by the dump.
3. Use PRTDUMP to format the VSCS control blocks and internal trace table.

   Note: You cannot print an entire dump using PRTDUMP. Use DUMPLOAD to print the entire dump.

See Table 80 on page 1165 to determine what book contains more information on dump collecting and dump formatting for your operating system.

Use the VSCS QUERY command to find the location of the VSCS internal trace table. Record the address. If you cannot find the proper control blocks, you can still use the table's address to find them using the dump formatter.

The general format of control blocks in a formatted dump is:

```
Control block name and address
Control block information
Offset .........data in dump format......... *..printable..*
```

The five options shown in Table 32 on page 454 allow you to choose what portion of the dump you want to format.

Note: For partial dumps created by DUMPLU, or requested for LU-related internal errors, the VSCS formatter does not allow you to specify any options. The title of the dump is displayed and the dump is processed without prompting the user. The output is similar to the output produced by the "Specific LU control blocks" option.

Table 32. Dump Formatting Options for VSCS

1	2	3	4	5
DTICGB	DTIPDBs	DTICLB	Global trace table	Module list
DTICIA	DTISDBs	DTICMT		
DTICLBs	DTISCB summary	DTIPIBs		
DTICMDs		DTIPLB		
DTIDCES		DTITAB		
DTIDMP		DTIVEIB		
DTIDSBs		DTIVLB		
DTIGTD		DTIWEB		
DTIPACT		DTIXMT		
DTIPGB		Logical unit		
DTIPIBs		trace entries		
DTIPLBs		ISTNIB		
DTIPRM		Receive RPL		
DTISAB		Send RPL		
DTISCB		ILU trace table		
DTISWBs				
DTITABs				
DTIUGB				
DTIVGB				
DTIVLBs				
Path table				
IFGACB				
Receive RPLs				
DTISCB summary				
Send RPL				
Global trace table				
VTAM work area				
IHASDWA				

Formatting may terminate if any of the following conditions occur:

- You are trying to process a back-level dump for either VTAM or VSCS.
- VSCS cannot find the control blocks in the dump. Early in initialization and late in termination, the appropriate pointers and exits may not exist. If VSCS cannot find the control blocks, it issues a message asking you for the VSCS trace table address. From this, it can locate the control blocks.
- Validity checking shows that a control block has been overlaid with data.
- A page of storage that VSCS needs is unavailable.

Figure 72 on page 455 shows how to use the dump formatting routine. This example is taken from an operator console listing.

```
prtdump prb00001
PROCESSING FILE 'PRB00001 DUMP'
CSIIPR550R DO YOU WISH TO FORMAT VTAM CONTROL BLOCKS?  REPLY YES OR NO
y
FORMAT VSCS CONTROL BLOCKS? (YES OR NO)
y
DTIFT04 - VSCS V4R2 DUMP FORMATTER
DTIFT05 - VSCS CONTROL BLOCKS LOCATED
DTIFT01 - VSCS DUMP FORMATTING OPTIONS
1. TRACE, GLOBAL AND LU RELATED CONTROL BLOCKS
2. STORAGE CONTROL BLOCKS
3. SPECIFIC LU CONTROL BLOCKS
4. VSCS TRACE TABLE ONLY
5. VSCS MODULE CROSS REFERENCE LISTING
T. USER PROVIDED TITLE
Q. EXIT

ENTER OPTION(1-5), T-TITLE OR Q-EXIT:

t
DTIFT11 - ENTER DESIRED TITLE(UP TO 120 CHARACTERS)
Prb8997a - Printer hang situation
DTIFT01 - VSCS DUMP FORMATTING OPTIONS
1. TRACE, GLOBAL AND LU RELATED CONTROL BLOCKS
2. STORAGE CONTROL BLOCKS
3. SPECIFIC LU CONTROL BLOCKS
4. VSCS TRACE TABLE ONLY
5. VSCS MODULE CROSS REFERENCE LISTING
T. USER PROVIDED TITLE
Q. EXIT

ENTER OPTION(1-5), T-TITLE OR Q-EXIT:

1
DTIF101 - SHOULD THE VSCS INTERNAL TRACE TABLE BE PROCESSED? Y|N

y
DTIF102 - ENTER TRACE OPTION:
F. FORMATTED
U. UNFORMATTED
ENTER OPTION(F OR U):

f
DTIF203 - SORTING    04 LU'S BY NAME
DTIF203 - SORTING    04 LU'S BY ADDRESS
DTIFT01 - VSCS DUMP FORMATTING OPTIONS
1. TRACE, GLOBAL AND LU RELATED CONTROL BLOCKS
2. STORAGE CONTROL BLOCKS
3. SPECIFIC LU CONTROL BLOCKS
4. VSCS TRACE TABLE ONLY
5. VSCS MODULE CROSS REFERENCE LISTING
T. USER PROVIDED TITLE
Q. EXIT

ENTER OPTION(1-5), T-TITLE OR Q-EXIT:
```

Figure 72 (Part 1 of 3). Using the VSCS Dump Formatting Routine

```
2
DTIFT01 - VSCS DUMP FORMATTING OPTIONS
1. TRACE, GLOBAL AND LU RELATED CONTROL BLOCKS
2. STORAGE CONTROL BLOCKS
3. SPECIFIC LU CONTROL BLOCKS
4. VSCS TRACE TABLE ONLY
5. VSCS MODULE CROSS REFERENCE LISTING
T. USER PROVIDED TITLE
Q. EXIT
ENTER OPTION(1-5), T-TITLE OR Q-EXIT:

3
ENTER LUNAME(1-8 CHARACTERS):

vtam
DTIFT03 - FORMATTING VTAM     LU CONTROL BLOCKS
DTIF204 - SEARCHING FOR LU....
DTIF205 - EXTRACTING LU TRACE ENTRIES...
DTIF201 - LU NAME NETA.VTAM     PROCESSED
DTIFT01 - VSCS DUMP FORMATTING OPTIONS
1. TRACE, GLOBAL AND LU RELATED CONTROL BLOCKS
2. STORAGE CONTROL BLOCKS
3. SPECIFIC LU CONTROL BLOCKS
4. VSCS TRACE TABLE ONLY
5. VSCS MODULE CROSS REFERENCE LISTING
T. USER PROVIDED TITLE
Q. EXIT

ENTER OPTION(1-5), T-TITLE OR Q-EXIT:
```

Figure 72 (Part 2 of 3). Using the VSCS Dump Formatting Routine

```
4
DTIFT02 - ENTER TRACE OPTION:
F. FORMATTED
U. UNFORMATTED
ENTER OPTION(F OR U):

u
DTIFT01 - VSCS DUMP FORMATTING OPTIONS
1. TRACE, GLOBAL AND LU RELATED CONTROL BLOCKS
2. STORAGE CONTROL BLOCKS
3. SPECIFIC LU CONTROL BLOCKS
4. VSCS TRACE TABLE ONLY
5. VSCS MODULE CROSS REFERENCE LISTING
T. USER PROVIDED TITLE
Q. EXIT

ENTER OPTION(1-5), T-TITLE OR Q-EXIT:

5
DTIFT09 - SEARCHING FOR MODULE NAMES
DTIFT10 - SORTING MODULE NAMES BY NAME
DTIFT10 - SORTING MODULE NAMES BY ADDRESS
DTIFT01 - VSCS DUMP FORMATTING OPTIONS
1. TRACE, GLOBAL AND LU RELATED CONTROL BLOCKS
2. STORAGE CONTROL BLOCKS
3. SPECIFIC LU CONTROL BLOCKS
4. VSCS TRACE TABLE ONLY
5. VSCS MODULE CROSS REFERENCE LISTING
T. USER PROVIDED TITLE
Q. EXIT

ENTER OPTION(1-5), T-TITLE OR Q-EXIT:

q
FORMAT VTAM CONTROL BLOCKS? (YES OR NO)

n
```

Figure 72 (Part 3 of 3). Using the VSCS Dump Formatting Routine

VSCS Trace Facility

The VSCS trace facility offers the following trace options:

Console communication services (CCS) trace
: Creates one entry for each IUCV request or response sent or received by VSCS.

Cleanup trace
: Creates entries for LU cleanup and global error situations that can occur, like abends, internal errors, or requests from CP that are not valid. This trace is activated automatically when one or more of the VSCS trace options are activated. It cannot run by itself.

Data trace (DAT)
: Creates one entry for logical unit-bound data (output), and one entry for CCS-bound logical unit data (input).

Dispatcher trace (DISP)
: Creates one entry each time DTIPSTAM or DTIVSTAM gives control to another module to process a work element block. This can occur more than once for a single work element block.

FRE trace
: Creates one entry for each request to free a block of storage.

GET trace
: Creates one entry for each request for a block of storage.

Operator trace (OPER)
: Creates one entry for each command processed and message issued by VSCS.

Storage statistics trace (STAT)
: Creates one entry, which is recorded externally, each time the STCHKTM time interval expires.

VTAM exit trace (VTAM)
: Creates one entry each time VTAM gives control to a VSCS VTAM exit.

Note: Each trace option causes some system degradation, but it can be controlled to some extent. The cleanup, CCS, operator, and storage statistics traces cause the least system degradation. The VTAM exit trace causes a little more degradation, and the data, dispatcher, FRE, and GET traces cause the most degradation. Consequently, the CCS and VTAM exit traces are the defaults.

Figure 73 on page 459 shows where each trace occurs.

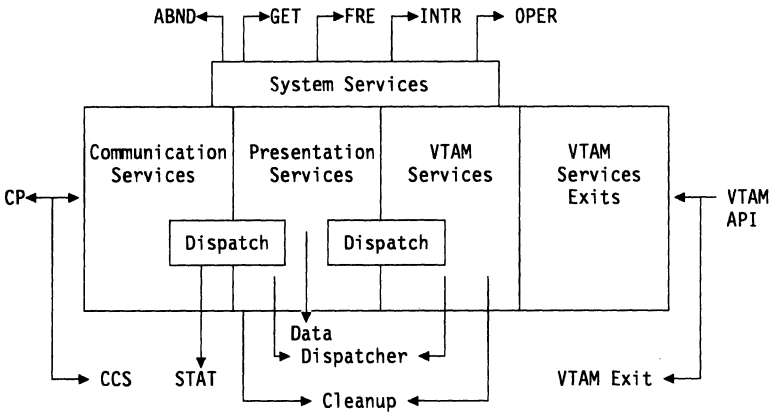


Figure 73. Where VSCS Traces Gather Data

Activating VSCS Traces

All of the traces except cleanup are started with operator commands. (For the CCS and VTAM exit trace options, the default is active.) The cleanup trace starts automatically with any of the other traces. See "VSCS Tracing Commands" in *VTAM Operation* for more information on starting the VSCS traces.

A trace can be started for one logical unit, up to 64 specific logical units, or all logical units at any time. If a trace is running for all logical units and the TRACEON command is specified for a single logical unit, the trace only records data for the logical unit specified. In the same manner, if you are tracing a specific logical unit and issue VSCS TRACEON (ALL, the trace then records data for all logical units.

If one or very few logical units are experiencing a problem, and you are re-creating the problem to get documentation, only run the traces on the logical units causing problems, and specify the ILU and SIZE options. By focusing on individual logical units, you can reduce extraneous trace data, trace table wrapping, and negative impacts on system performance.

When you specify the ILU option, VSCS allocates an individual trace table for each LU that is traced. Use the SIZE option with the ILU option to set the size of each LU table. VSCS still records trace entries to the global VSCS trace table when the ILU option is active. See "VSCS Tracing Commands" in *VTAM Operation* for instructions on using the ILU and SIZE operands.

The external trace option prevents trace information from being overlaid when the internal table wraps. If MODE=EXT, VSCS maintains both the internal trace table and the external trace table. This ensures that a dump taken for an unrecoverable error will contain the relevant data from the internal trace table. The external VSCS trace entries are sent to the external recording facilities of CP. See "Using the External VSCS Traces" on page 460 for information on obtaining the external trace output.

You can use the VSCS QUERY command to display:

- What traces are active
- Which logical units are being traced (if selective tracing is used)
- Modes of tracing, for example, external and individual LU trace tables
- The location of the internal trace table (if it is allocated).

Using the External VSCS Traces

To collect trace data from the VSCS trace running in external mode, you must:

1. Set up operating system utilities to collect the trace information.
2. Run the VSCS trace in external mode.
3. Stop the VSCS trace to quit collecting the information.
4. Stop running the operating system utilities that collect the information.
5. Format and print the trace information.

This section describes how to run and stop the VSCS trace in external mode.

To run the VSCS traces externally, issue the following commands from the VSCS console:

`VSCS TRACEON (EXT ALL` EXT means run the trace externally. ALL means turn on the CCS, data, dispatcher, FRE, GET, and VTAM exit trace options.

`VSCS TRACEON` *netid.luname*
This is required only for selective LU tracing.

VSCS trace output is sent to a spool file.

When you have collected enough data and you want to stop VSCS tracing, issue the following commands from the VSCS console:

`VSCS TRACEOFF (EXT` Turn off external tracing.
`VSCS TRACEOFF (ALL` Turn off all VSCS trace options.

To start or change the current VSCS trace options after issuing the TRACEOFF command or to switch from selective to non-selective tracing, use the following:

`VSCS TRACEON *` Turn on non-selective tracing. This command is needed only if you used selective tracing earlier.

`VSCS TRACEON (CCS VTAM` Activate the CCS and VTAM trace options, and use the internal global trace table. You should use this minimum trace at all times.

See "Collecting External Trace Data (VM)" on page 359 for information on using operating system utilities to collect, format, and print VSCS trace information.

VSCS Trace Table

VSCS records the data in a global internal table that wraps around when full. The size of the trace table, up to 64000 entries, is specified during VSCS initialization. The default size is 2000 records of 32 bytes each.

Storage for the global VSCS trace table is not allocated until the trace is activated and tracing begins. When the trace facility is turned off, storage for the global VSCS trace table is preserved and can be dumped by the operator using the CP VMDUMP FORMAT command.

The VSCS QUERY command gives the starting address, ending address, and size of the global VSCS trace table. In addition, the global VSCS trace table can be found through a pointer at SABTRHDR in the DTISAB.

Trace Table Header

The VSCS trace table contains a header record (DTITHDR) followed by the trace records (DTITREC). Figure 74 shows the trace table header.

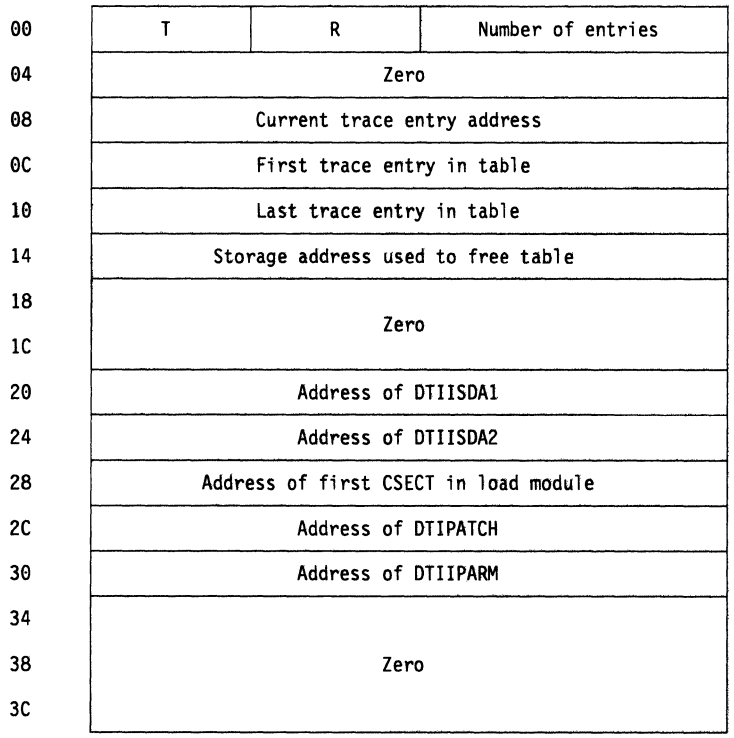


Figure 74. VSCS Trace Header Format

The header shows this information:

Byte (hex)	Contents
00–01	DTITHDR element header (TR)
02–03	Number of entries in table
04–08	Zero
08–0B	Pointer to current entry in table
0C–0F	Pointer to first entry in table
10–13	Pointer to last entry in table
14–17	Address to be used in freeing the trace table storage
18–1F	Zero
20–23	Address of DTIISDA1
24–27	Address of DTIISDA2
28–2B	Address of DTIISTRT
2C–2F	Address of DTIIPATCH
30–33	Address of DTIIPARM
34–3F	Zero.

Examples of VSCS Trace Output

Following are three ways to look at the VSCS global trace table:

- Online at the VSCS operator console
- In dump output
- If produced in external mode, formatted with the TRACERED utility.

To view the global trace table online, issue the VM DISPLAY command from the VSCS operator console. Figure 75 on page 463 is an example of DISPLAY output. (The trace record titles do not appear in actual output; they are in the example to identify the different trace entry types.)

Figure 76 on page 464 shows what the trace table looks like in dump output. (The trace record titles and the frames around the output do not appear in actual output. They are added to the examples to identify the different trace entry types.)

Figure 77 on page 465 shows an example of a VSCS trace running externally and formatted with TRACERED.

To determine what book describes the VM DISPLAY command and the TRACERED utility, see Table 80 on page 1165.

```
Storage
Address
         Trace header
265020   E3D97F58  00000000  00265580  00265040     *TR"..........& *
265030   00363B40  00265008  00000000  00000000     *... ..&.........*
265040   002221F8  0025E588  0021F000  00263BA8     *...8..V...0.....*
265050   00220688  00000000  00000000  00000000     *................*

         VSCS CCS data trace entry
265120   C3D5C3E3  00012000  00060000  5CC3C3E2     *CNCT........*CCS*
265130   40404040  05010000  00000006  40404040     *    ........    *

         VSCS CCS global trace entries
265200   00213988  00000001  C7D3C2D3  D9010228     *........GLBLR...*
265210   0C000000  00110000  00000000  C8C9D5C5     *............HINE*
265220   0000CE98  00000001  C7D3C2D3  F1000008     *........GLBL1...*
265230   0E000000  00010000  00000000  E5E3C1D4     *............VTAM*

         VSCS VTAM exit trace entries
265320   E5D3D6C7  00367A98  00367C58  00000000     *VLOG..:...@.....*
265330   00000000  00000000  10000000  00000005     *................*
265340   E7D3D6C7  80000000  00000000  00000000     *XLOG............*
265350   00000000  00000000  00000000  00000000     *................*

         VSCS dispatcher trace entries
2653E0   D7020000  00367A98  0036FF18  07040000     *P.....:.........*
2653F0   80000000  00000000  00000000  E2C9D5E3     *............SINT*
265400   E9000000  00100000  00000000  01000002     *Z...............*
265410   00000000  00000000  00000000  00000000     *................*
265480   E5030000  00367A98  00375A40  10040000     *V.....:.... ....*
265490   81000080  00000000  00000000  E2C5D5C4     *............SEND*
2654A0   E9000000  00100000  00000000  01000002     *Z...............*
2654B0   00000000  00000000  00000000  00000000     *................*

         VSCS CCS LU trace entries
265860   00210E10  00020000  01000002  F2830009     *............2...*
265870   01000001  00020000  00000000  40020040     *............ .. * 
2658A0   00210E68  00020000  01000002  C3830002     *............C...*
2658B0   0E000000  00040000  00000000  00070000     *................*

         VSCS Storage trace entries
265140   C6D9C5C2  002290D0  0000CEC0  E2C4C9D5     *FREB........SDIN*
265150   01000000  6024F41C  50250828  00000000     *....-.4.&.......*
265160   C7C5E3C2  00229300  0021A940  0005E000     *GETB....... ....*
265170   004C0100  502384A0  60236A92  C5D5C1C2     *.<..&...-...ENAB*

         VSCS input and output data records
266060   D6010000  00219B68  00275DA8  00000000     *O.........).....*
266070   00000000  00000000  00000000  00000000     *................*
266240   C9040400  00219B68  0021AD18  00067D5B     *I.............'$*
266250   60115B60  00000000  00000000  00000000     *-.$-............*

         VSCS cleanup trace entry
266BE0   D7D9C2D2  00219B68  00840000  80000000     *PRBK............*
266BF0   00000200  20000000  01000002  00270DF0     *.............0*
```

Figure 75. VSCS Trace Records Displayed at the Operator Console

Trace header

```
STORAGE
 ADDR
265020  E3D97F58 00000000 00265580 00265040 00363B40 00265008 00000000 00000000  *TR*.........&  ...  ..&........*
265040  002221F8 0025E588 0021F000 00263BA8 00220688 00000000 00000000 00000000  *...8..V...0.....................*
```

VSCS CCS data trace entry

```
265120  C3D5C3E3 00012000 00060000 5CC3C3E2 40404040 05010000 00000006 40404040  *CNCT........*CCS     ........    *
```

VSCS CCS global trace entries

```
265200  00213988 00000001 C7D3C2D3 D9010288 0C000000 00110000 00000000 C8C9D5C5  *........GLBLR..............HINE*
265220  0000CE98 00000001 C7D3C2D3 F1000008 0E000000 00010000 00000000 E5E3C1D4  *........GLBL1..............VTAM*
```

VSCS VTAM exit trace entries

```
265320  E5D3D6C7 00367A98 000367C58 00000000 00000000 00000000 10000000 00000005  *VLOG..:...@.....................*
265340  E7D3D6C7 80000000 00000000 00000000 00000000 00000000 00000000 00000000  *XLOG.. .........................*
```

VSCS dispatcher trace entries

```
2653E0  D7020000 00367A98 0036FF18 07040000 80000000 00000000 00000000 E2C9D5E3  *P.....:.....................SINT*
265400  E9000000 00100000 00000000 01000002 00000000 00000000 00000000 00000000  *Z...............................*
265480  E5030000 00367A98 00375A40 10040000 81000080 00000000 00000000 E2C5D5C4  *V.....:.....................SEND*
2654A0  E9000000 00100000 00000000 01000002 00000000 00000000 00000000 00000000  *Z...............................*
```

VSCS CCS lu trace entries

```
265860  00210E10 00020000 01000002 F2830009 01000001 00020000 00000000 40020040  *.............2.................*
2658A0  00210E68 00020000 01000002 CE830002 0E000000 00040000 00000000 00070000  *.............C.................*
```

VSCS Storage trace entries

```
265140  C6D9C5C2 002290D0 0000CEC0 E2C4C9D5 01000000 6024F41C 50250828 00000000  *FREB........SDIN....-.4.&.......*
265160  C7C5E3C2 00229300 0021A940 0005E000 004C0100 502384A0 60236A92 C5D5C1C2  *GETB.......  .....<..&...-...ENAB*
```

VSCS input and output data entries

```
266060  D6010000 00219B68 00275DA8 00000000 00000000 00000000 00000000 00000000  *O.........).....................*
266240  C9040400 00219B68 0021AD18 00067D5B 60115B60 00000000 00000000 00000000  *I.............'$-.$-............*
```

VSCS cleanup trace entry

```
266BE0  D7D9C2D2 00219B68 00840000 80000000 00000200 20000000 01000002 00270DF0  *PRBK........................0*
```

Figure 76. VSCS Trace Records in Dump Output

```
3D 0E VSCS3      VM/GCS USER REQUESTED GTRACE
      TIME OF DAY CLOCK = A1F762AA98968A01
      LENGTH OF GTF HEADER AND TRACE DATA = 0058
      FORMAT ROUTINE ID = F5
      EVENT IDENTIFICATION = EFAA
  VOPN   TAB 0154F028  RPL 0154F1E8  RPLREQ 17  RTN CODE 00  NAU 0002  RPLFDBK 0000  RPLSENS 0000
         VS STATE 00  VLBERFLG 00  PATHID 0000  VLBSWAP 10 00 00 00  VLBLRCD 00  EXIT DATA 17 07 02
  XOPN   PS STATE 80  PLB- SM 00 00 00 00  OFLG1 00  NOM 00 00  EDIT 00  IAFLG 00  DFLG1 00
         RSVD 00 00000000 00000000 00000000 00000000
3D 0E VSCS3      VM/GCS USER REQUESTED GTRACE
      TIME OF DAY CLOCK = A1F762AA98AA5E01
      LENGTH OF GTF HEADER AND TRACE DATA = 0058
      FORMAT ROUTINE ID = F5
      EVENT IDENTIFICATION = EFAA
  PDISP  LBCSTTUS 02  RSVD 0000  TAB 0154F028  WEB 015540D8  WEBFUN 07  WEBMODE 04  WEBVSRPS 00  RSVD 00
         PS STATE 80  PLB- SM 00 00 00 00  OFLG1 00  NOM 00 00  EDIT 00  IAFLG 00  DFLG1 00  RSVD 00  MODULE SINT
  ZDISP  RSVD 00  PATHID 0000  VS STATE 00  VLBSWAP 10 00 00 00  VLBERFLG 00  VLBLRCD 00  VLBERCVY 00
         CID 01000002  RSVD 00000000 00000000 00000000 00000000
3D 0E VSCS3      VM/GCS USER REQUESTED GTRACE
      TIME OF DAY CLOCK = A1F762B0647F1001
      LENGTH OF GTF HEADER AND TRACE DATA = 0038
      FORMAT ROUTINE ID = F5
      EVENT IDENTIFICATION = EFAA
  CCS    WEB 01553028  PATHID 0002 0000  CID 01000002  TYPE 2  PS STATE 83  WEBCOUNT 0009  WEBFUN 01  WEBMODE 00
         WEB- CHAR 00  EDIT 01  FLAGS 00 02  LINE 0000  CURSR 0000  TABCH 00  RSVSA 00  DATA 40020050  * .. *
```

Figure 77. VSCS Trace Records Formatted with TRACERED

Note: To determine what book describes how to interpret "time of day clock" entries in trace output, see Table 80 on page 1165.

VSCS Trace Record Formats

The following list contains all VSCS trace entries, as well as LBCSTATE values for PLBIN*xxx* and WEB function codes. For detailed descriptions, see the page indicated.

VSCS CCS Trace Record Formats

When the CCS trace option is active, VSCS traces all traffic to and from CCS. However, you can limit the CCS trace to a selected set of logical units when you start it. The CCS trace produces three types of record formats:

- *Logical unit-related* for traffic associated with logical units
- *Global* for traffic not associated with a logical unit
- *Data* for traffic associated with logical units or global traffic.

CCS Trace Entry for a Logical Unit

00	DTIWEB address			
04	PATH IDs			
08	VTAM CID			
0C	Flow ind.	LBCSTATE	WEBCOUNT	
10	WEBFUN	WEBMODE	WEBCHAR	WEBEDIT
14	WEBFLAGS	WEBCPFLG	WEBLINE	
18	WEBCURSR		WEBTABCH	WEBRSVSA
1C	First four characters of data			

Displacement:

00–03 Address of DTIWEB associated with this request or response

04–07 Path ID from communication services local block (CLB)

- 04–05 VSCS path ID
- 06–07 CCS path ID

08–0B VTAM communication identifier (CID), either OAF or DAF

0C

Flow indicator	**Flow direction**
R (Receive)	CCS => VSCS (outbound)
Y (REPLY)	VSCS => CCS (inbound)
1 (1-way SEND)	VSCS => CCS (inbound)
2 (2-way SEND)	VSCS => CCS (inbound)
T (Reject)	VSCS => CCS (inbound)

C (Message complete) CCS => VSCS (outbound)

Note: In this book, whenever a flow is referred to as *inbound* or *outbound*, you should take the console communications services (CCS) as the reference point. That is, flows *toward* the CCS are *inbound*; flows *away from* the CCS are *outbound*.

0D	LBCSTATE from PLB		
	1...	LBCINT	Internal mode if on (on=1). If LBCINT is on, see "LBCSTATE Values for PLBINxxx" on page 535. If LBCINT is off, the following values apply:
	.11.	LBCFSSMF	Full-screen support mode flags
	.1..	LBCFSAV	Screen available
	..1.	LBCFSII	Full-screen input inhibited
			Description of settings for LBCFSSMF: B'00' CP (console) mode B'01' Virtual machine mode state between ATTN and READ, queue CP messages. B'10' Virtual machine mode queue CP messages. B'11' Virtual machine mode available for switch.
	...1	LBCCOPY	Copy mode: on=1
	.... 1...	LBCNAL	NAL flag: on=1
	.... .11.	LBCHOLD	HOLDING flag: on=B'11'
	.... .1..	LBCM10	MORE 10 flag: on=1
	.... ..1.	LBCM50	MORE 50 flag: on=1
	1	LBCCMSD	Screen written with a CMS DIAGNOSE: on=1
	....		No MORE/HOLDING state
0E–0F	WEBCOUNT	Length of actual data in WEBDATA	
10	WEBFUN	WEB function code. This is explained in "WEB Function Codes" on page 535.	
11	WEBMODE	Mode: 00 Console 01 CMS 02 Full-screen 04 Internal	
12	WEBCHAR	Character set (set by CCS for console and CMS modes) 00 EBCDIC character set 01 APL character set 02 TEXT character set 04 DBCS character set 05 DBCS and APL character set 06 DBCS and TEXT character set	
13	WEBEDIT	Editing characteristics	
	1...	WEBIMMED	Mode switch immediately

	.1..	WEBPETCM	'PRESS ENTER TO CONTINUE' message
	..1.	WEBCLRSC	CP clear screen
	...x		Zero
	.... 1...	WEBHIEKO	Highlight input display
	.... .1..	WEBEDPTI	Do not print or do not display
	.... ..1.	WEBNOEKO	Do not redisplay input on next output line
	1	WEBEDCUP	Convert to uppercase
14	WEBFLAGS		
	1...	WEBTTYWT	TTY/TWX WRITE when a READ is pending. Reject READ.
	.1..	WEBCLVMI	Cancel all VM Reads and Writes for a display
	.1..	WEBCONMD	Request for display in CONMODE 3270
	..1.	WEBLED	Limited edit diagnose
	...1	WEBANFSW	Allow any full-screen WRITE; first full-screen WRITE after reset.
	.... 1...	WEBDIAL	SNA dialed logical unit
	.... .1..	WEBNOMOR	CMS ERASE WRITE no "MORE" requested
	.... ..1.	WEBPRMPT	WEBDATA contains pre-logon prompt
	1	WEBPASPA	Pass PA1 (FSSM) as data to application
15	WEBCPFLG	CCS status flags for VSCS	
15	WEBSAFLG	VSCS status flags	
	1...	WEBCPNCR	No carriage return (hard-copy terminal)
	.1..	WEBCPPRI	Priority flag
	..1.	WEBCPALM	Ring the alarm
	..1.	WEBSAHLD	HOLDING
	...1	WEBCPMDE	CP-generated bit = 1
	...1	WEBSAMOR	MORE VM-generated bit = 0
	.... 1...	WEBCPHLW	Highlight WRITE requested
	.... .1..	WEBCPRSP	Indicates CCS response
	.... .1..	WEBSARSP	Indicates VSCS response
	.... ..1.	WEBCPRRR	CCS request, requires response
	.... ..1.	WEBSARRR	VSCS request, requires response
	1	WEBCPRNR	CCS request, requires no response
	1	WEBSARNR	VSCS request, requires no response
16–17	WEBLINE	Line number for CMS WEBCMWRT	
16	WEBLAID	Logical attention identifier	
17	WEBNLLOS	Number of lines left on screen	
18–19	WEBCURSR	Cursor position may be a relative address or a buffer address	
1A	WEBTABCH	Tab (not DTITAB) character	
1B	WEBRSVSA	Reserved	
	1...	WEBCPRDS	DTIWEB is a redisplay DTIWEB
	.1..	WEBNOVCK	Do not validity check data
	..1.	WEBATTNW	Attention WRITE (VSCS only) yes = 1
	...1	WEBINLIP	Logon in progress

	.... x...		Zero
	.... .1..	WEBCHDIR	Change Direction required
	.... ..1.	WEBMASKW	WRITE inhibit mask
	x		Zero
1C–1F	WEBDATA	First four bytes of data field	

Note: This field is valid only if WEBCOUNT > 0.

Global CCS Trace Entry

A global entry occurs if CCS receives a request for a path ID that has been severed.

00	DTIWEB address			
04	PATH IDs			
08	G	L	B	L
0C	Flow ind.	WEBCAN	WEBCOUNT	
10	WEBFUN	WEBMODE	WEBCHAR	WEBEDIT
14	WEBFLAGS	WEBCPFLG	WEBLINE	
18	WEBCURSR		WEBTABCH	WEBRSVSA
1C	First four characters of data			

Displacement:

00–03 Address of DTIWEB associated with this request or response

04–07 Path ID (from CLB)

04–05 VSCS path ID
06–07 CCS path ID

08–0B Characters 'GLBL'

0C

Flow indicator	**Flow direction**
R (Receive)	CCS => VSCS
Y (REPLY)	VSCS => CCS
1 (1-way SEND)	VSCS => CCS
2 (2-way SEND)	VSCS => CCS
T (Reject)	VSCS => CCS
C (Message complete)	CCS => VSCS

0D WEBCAN Number of control areas

0E–0F WEBCOUNT Length of actual data in WEBDATA

10 WEBFUN WEB function code. This is explained in "WEB Function Codes" on page 535.

11 WEBMODE Mode:

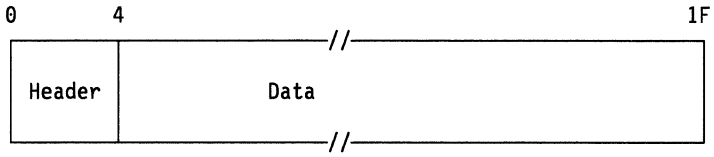
00 Console
01 CMS
02 Full-screen
04 Internal

12	WEBCHAR	Character set (set by CCS for console and CMS modes)	
		00 EBCDIC character set	
		01 APL character set	
		02 TEXT character set	
		04 DBCS character set	
		05 DBCS and APL character set	
		06 DBCS and TEXT character set	
13	WEBEDIT	Editing characteristics	
	1...	WEBIMMED	Mode switch immediately
	.1..	WEBPETCM	'PRESS ENTER TO CONTINUE' message
	..1.	WEBCLRSC	CP clear screen
	...x		Zero
	.... 1...	WEBHIEKO	Highlight input display
	.... .1..	WEBEDPTI	Do not print or do not display
	.... ..1.	WEBNOEKO	Do not redisplay input on next output line
	1	WEBEDCUP	Convert to upper case
14	WEBFLAGS		
	1...	WEBTTYWT	TTY/TWX WRITE when a READ is pending. Reject READ.
	1...	WEBCLVMI	Cancel all VM Reads and Writes for a display
	.1..	WEBCONMD	Request for display in CONMODE 3270
	..1.	WEBLED	Limited edit diagnose
	...1	WEBANFSW	Allow any full-screen WRITE; first full-screen WRITE after reset.
	.... 1...	WEBDIAL	SNA dialed logical unit
	.... .1..	WEBNOMOR	CMS ERASE or WRITE no "MORE" requested
	.... ..1.	WEBPRMPT	WEBDATA contains pre-logon prompt
	1	WEBPASPA	PA1 (FSSM) pass as data to application
15	WEBSAFLG	VSCS status flags	
	1...	WEBCPNCR	No carriage return (hard-copy terminal)
	.1..	WEBCPPRI	Priority flag
	..1.	WEBCPALM	Ring the alarm
	..1.	WEBSAHLD	HOLDING
	...1	WEBCPMDE	CP-generated bit = 1
	...1	WEBSAMOR	MORE VM-generated bit = 0
	.... 1...	WEBCPHLW	Highlight WRITE requested
	.... .1..	WEBCPRSP	Indicates CCS response
	.... .1..	WEBSARSP	Indicates VSCS response
	.... ..1.	WEBCPRRR	CCS request, requires response
	.... ..1.	WEBSARRR	VSCS request, requires response
	1	WEBCPRNR	CCS request, requires no response
	1	WEBSARNR	VSCS request, requires no response
16–17	WEBLINE	Line number for CMS WEBCMWRT	
16	WEBLAID	Logical attention identifier	
17	WEBNLLOS	Number of lines left on screen	

18–19	WEBCURSR	Cursor position may be a relative address or a buffer address	
1A	WEBTABCH	Tab (not DTITAB) character	
1B	WEBRSVSA	Reserved	
	1...	WEBCPRDS	DTIWEB is a redisplay DTIWEB
	.1..	WEBNOVCK	Do not validity check data
	..1.	WEBATTNW	Attention WRITE (VSCS only) yes = 1
	...1	WEBINLIP	Logon in progress
	.... x...		Zero
	.... .1..	WEBCHDIR	Change Direction required
	.... ..1.	WEBMASKW	WRITE inhibit mask
	x		Zero
1C–1F	WEBDATA	First four bytes of data field	

Note: This field is valid only if WEBCOUNT > 0.

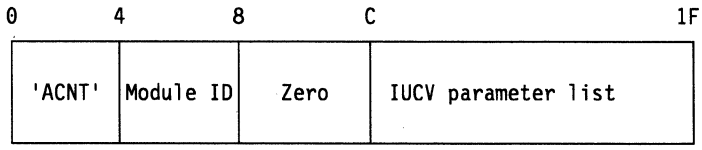
CCS Data Trace Entries



Displacement:

00–03	Header		
	'ACNT'	Accept connection	CCS => VSCS
	'CMSG'	Completion of a 1-way SEND	CCS => VSCS
	'CNCT'	Connect	VSCS => CCS
	'GMSG'	Ignored traffic on global path	CCS => VSCS
	'LIER'	Logical interface error	CCS => VSCS
	'SCNT'	Request to SEVER connection	CCS => VSCS
	'SEVR'	Connection severed	VSCS => CCS or CCS => VSCS
04–1F	First 28 bytes of data from area passed		

ACNT Entry: Accept connection, CCS => VSCS

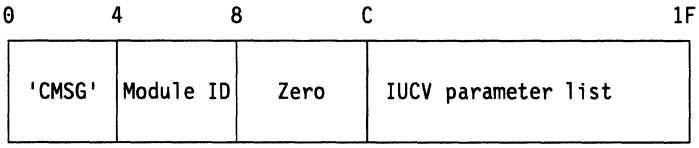


Displacement:

00–03	Characters 'ACNT'
04–07	Last four characters of module making trace entry. First four characters are always DTIP.
08–0B	Zero
0C–0D	Path ID

0E	IPFLAGS1		
	1...	IPALL	QUIESCE, RESUME, and SEVER ALL
	.1..	IPQUSCE	Connect in QUIESCE mode
	..1.	IPPRTY	Priority message, REPLY required
	...1	IPNORPY	One-way protocol
	.... 1...	IPCPENTY	Entry from CP
	.... .1..	IPFGMID	Message ID specified
	.... ..1.	IPFGPID	Path ID specified
	1	IPFGMCL	Message class specified
0F	IPTYPE	External interrupt code	
10–11	IPSMGLIM	Message limit	
12–13	Zero		
14–15	IPAUDIT	Audit trail as follows:	
14	IPAUDIT1	Audit trail byte 1	
	1...	IPADRPLE	REPLY too long for buffer
	.1..	IPADSNPX	Protection exception on SEND buffer
	..1.	IPADSNAX	Address exception on SEND buffer
	...1	IPADANPX	Protection exception on ANSWER buffer
	.... 1...	IPADANAX	Address exception on ANSWER buffer
	.... .1..	IPADRJCT	Message was rejected
	xx		Zero
15	IPAUDIT2	Audit trail byte 2	
	1...	IPADRCPX	Protection exception on RECEIVE buffer
	.1..	IPADRCAX	Address exception on RECEIVE buffer
	..1.	IPADRPPX	Protection exception on REPLY buffer
	...1	IPADRPAX	Address exception on REPLY buffer
	.... 1...	IPADSVRD	Path was severed
	xxx		Zero
16–17	Zero		
18–1B	User data		
1C–1F	First four bytes of logical unit name		

CMSG Entry: Completion of a one-way send, CCS => VSCS

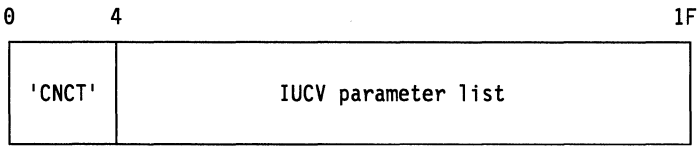


Displacement:

00–03	Characters 'CMSG'		
04–07	Last four characters of module making trace entry. First four characters are always DTIP.		
08–0B	Zero		
0C–0D	Path ID		
0E	IPFLAGS1		
	1...	IPALL	QUIESCE, RESUME, and SEVER ALL
	.1..	IPQUSCE	Connect in QUIESCE mode

	..1.	IPPRTY	Priority message, REPLY required
	...1	IPNORPY	One-way protocol
	.... 1...	IPCPENTY	Entry from CP
	.... .1..	IPFGMID	Message ID specified
	.... ..1.	IPFGPID	Path ID specified
	1	IPFGMCL	Message class specified
0F	IPTYPE	External interrupt code	
10–13	IPMSGID	Message ID	
14–17	IPTRGCLS	Target class	
14–15	IPAUDIT	Audit trail as follows:	
14	IPAUDIT1	Audit trail byte 1	
	1...	IPADRPLE	REPLY too long for buffer
	.1..	IPADSNPX	Protection exception on SEND buffer
	..1.	IPADSNAX	Address exception on SEND buffer
	...1	IPADANPX	Protection exception on ANSWER buffer
	.... 1...	IPADANAX	Address exception on ANSWER buffer
	.... .1..	IPADRJCT	Message was rejected
	xx		Zero
15	IPAUDIT2	Audit trail byte 2	
	1...	IPADRCPX	Protection exception on RECEIVE buffer
	.1..	IPADRCAX	Address exception on RECEIVE buffer
	..1.	IPADRPPX	Protection exception on REPLY buffer
	...1	IPADRPAX	Address exception on REPLY buffer
	.... 1...	IPADSVRD	Path was severed
	xxx		Zero
16–17		Zero	
18–1B	IPBFADR1	Address of buffer 1	
1C–1D		Zero	
1E–1F	IPBFLN1	Length of buffer 1	

CNCT Entry: Request for connection, VSCS => CCS

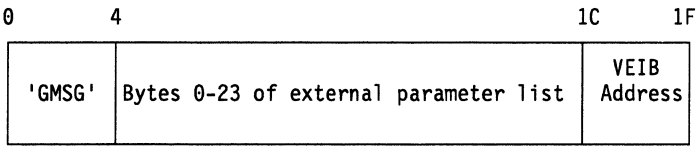


Displacement:

00–03	Characters 'CNCT'		
04–05	Path ID		
06	IPFLAGS1		
	1...	IPALL	QUIESCE, RESUME, and SEVER ALL
	.1..	IPQUSCE	Connect in QUIESCE mode
	..1.	IPPRTY	Priority message, REPLY required
	...1	IPNORPY	One-way protocol
	.... 1...	IPCPENTY	Entry from CP
	.... .1..	IPFGMID	Message ID specified
	.... ..1.	IPFGPID	Path ID specified

	1	IPFGMCL	Message class specified
07	IPRCODE	Return code	
08–09	IPSMGLIM	Message limit	
0A	IPFCNCD	Function code	
0B	Zero		
0C–13	IPVMID	Target virtual machine ID	
14–1B	'*CCS'	(These characters will be left-justified in the eight-byte field.)	
1C–1F	First four bytes of logical unit name		

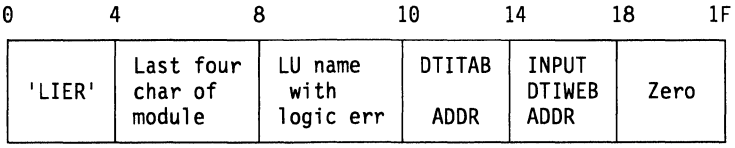
GMSG Entry: Traffic on global path is ignored, CCS => VSCS



Displacement:

00–03	Characters 'GMSG'		
04–05	Path ID		
06	IPFLAGS1		
	1...	IPALL	QUIESCE, RESUME, and SEVER ALL
	.1..	IPQUSCE	Connect in QUIESCE mode
	..1.	IPPRTY	Priority message, REPLY required
	...1	IPNORPY	One-way protocol
	.... 1...	IPCPENTY	Entry from CP
	.... .1..	IPFGMID	Message ID specified
	.... ..1.	IPFGPID	Path ID specified
	1	IPFGMCL	Message class specified
07	IPTYPE	External interrupt code	
08–0B	IPMSGID	Message ID	
0C–0F	IPTRGCLS	Target class	
0C–0D	IPAUDIT	Audit trail as follows:	
0C	IPAUDIT1	Audit trail byte 1	
	1...	IPADRPLE	REPLY too long for buffer
	.1..	IPADSNPX	Protection exception on SEND buffer
	..1.	IPADSNAX	Address exception on SEND buffer
	...1	IPADANPX	Protection exception on ANSWER buffer
	.... 1...	IPADANAX	Address exception on ANSWER buffer
	.... .1..	IPADRJCT	Message was rejected
	xx		Zero
0D	IPAUDIT2	Audit trail byte 2	
	1...	IPADRCPX	Protection exception on RECEIVE buffer
	.1..	IPADRCAX	Address exception on RECEIVE buffer
	..1.	IPADRPPX	Protection exception on REPLY buffer
	...1	IPADRPAX	Address exception on REPLY buffer
	.... 1...	IPADSVRD	Path was severed
	xxx		Zero
0E–0F		Zero	
10–13	IPBFADR1	Address of buffer 1	
14–15		Zero	
16–17	IPBFLN1	Length of buffer 1	
18–1B	IPSRCCLS	Source class	
1C–1F	VEIB Address		

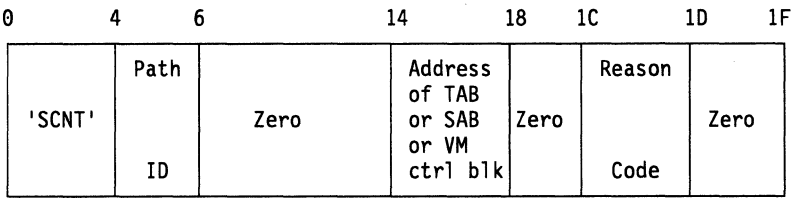
LIER Entry: Logical interface error, CCS => VSCS



Displacement:

00–03 Character 'LIER'
04–07 Last four characters of module making trace entry. First four characters are always DTIP.
08–0F Logical unit name of device with logical interface error.
10–13 Address of DTITAB
14–17 Address of DTIWEB that was input to module making trace entry.
18–1F Zero

SCNT Entry: Request to SEVER connection, CCS => VSCS



Displacement:

00–03 Characters 'SCNT'
04–05 ID of path that is to be severed
06–13 Zero
14–17 Address of terminal anchor block (TAB), service application block (SAB), or VTAM control block
18–1B Zero
1C Reason code
- 1 VSCS not enabled by CP
- 2 Maximum logical unit limit exceeded
- 3 No VSCS global CONNECT accepted
- 4 VTAM service machine (VSM) not enabled
- 5 Duplicate logical unit name found on VTAM service machine (VSM) RDEVBLOK chain
- 6 This path ID is currently in use
- 7 VTAM service machine was previously connected
- 8 Message limit exceeded
- 9 VTAM service machine global path severed

1D–1F Zero

SEVR Entry: Connection severed, VSCS => CCS or CCS => VSCS

0	4	6	14	18	1F
'SEVR'	Path ID	Zero	Address of TAB or SAB	Zero	

Displacement:

00–03 Characters 'SEVR'
04–05 ID of path that was severed
06–13 Zero
14–17 Address of TAB or SAB
18–1F Zero

VSCS Cleanup Error Trace Record Formats

The VSCS cleanup trace is activated automatically if any of the other VSCS traces are active. The cleanup trace applies only to the set of logical units specified for the other VSCS traces. Normal session terminations and error situations are traced. Three kinds of cleanup trace records are produced:

- Logical unit-related (VPUR, PBRK, or PFOR)
- Data-related (associated with VSCS global path) (CXIT)
- ABEND or internal errors (ABND or INTR).

Logical Unit-Related Trace Entry (VPUR or PBRK)

00	Module ID			
04	DTITAB address			
08	RPL REQ	LBCSTATE	PLBFSSMF	PLBLFLG1
0C	PLBFLG1	PLBNOM		PLBOFLG1
10	LBCSTATE	VLBLRCD	WEBFUN	TRCFLOW
14	VLBSWAP			
18	VLBCID			
1C	DTIWEB or RPL address			

Displacement:

00–03 Four-character module ID (characters "PBRK" or "VPUR")
04–07 TAB address for current request
08 RPL request code
09 LBCSTATE from PLB

1...	LBCINT	Internal mode if on (on=1). If LBCINT is on, see "LBCSTATE Values for PLBINxxx" on page 535. If LBCINT is off, the following values apply:
.11.	LBCFSSMF	Full-screen support mode flags

	.1..	LBCFSAV	Screen available
	..1.	LBCFSII	Full-screen input inhibited
			Description of settings for LBCFSSMF: B'00' CP (console) MODE B'01' Virtual machine mode state between ATTN and READ, queue CP messages B'10' Virtual machine mode queue CP messages B'11' Virtual machine mode available for switch.
	...1	LBCCOPY	Copy mode: on=1
	.... 1...	LBCNAL	NAL flag: on=1
	.... .11.	LBCHOLD	HOLDING flag: on=B'11'
	.... .1..	LBCM10	MORE 10 flag: on=1
	.... ..1.	LBCM50	MORE 50 flag: on=1
	1	LBCCMSD	Screen written with a CMS DIAGNOSE: on=1
	....		No MORE/HOLDING state
0A	PLBFSSMF	Full-screen support mode flags	
	1...	PLBFPA1	PA1 key received switch mode
	.1..	PLBFSRFC	Mode switch required (FSSM to console)
	..1.	PLBFSAFC	Switch available (FSSM to console)
	...1	PLBFCNMP	Console message pending
	.... 1...	PLBFVSIP	VTAM services SEND in progress
	.... .1..	PLBFCLGF	Conditional LOGOFF received
	.... ..1.	PLBFCMRR	CP message pending, response required
	1	PLBCMSEW	Screen just cleared
0B	PLBLFLG1		
	1...	PLBVS	VTAM services LOGOFF received bypass logo
	.1..	PLBLGFC	WEBINLGF sent to CCS conditional LOGOFF
	..1.	PLBLGFU	WEBINLGF sent to CCS unconditional LOGOFF
	...1	PLBLERR	WEBINERR sent to CCS for hard error
	.... 1...	PLBLLERR	WEBINLER sent to CCS for interface error
	.... .1..	PLBLVERR	WEBINVER sent to CCS for internal logic error
	.... ..1.	PLBLSEVR	CP CCS SEVER received, VSCS must SEVER
	1	PLBNSTR	No storage indicator in LOGOFF processing
0C	PLBFLG1		
	1...	PLBLGO	Logo or no logo bit (logo = 1)
	.1..	PLBINAR	APL or TEXT starting in input area
	..1.	PLBLFLNP	LOGON from logo not permitted

	...1	PLBCONCT	Current connect count incremented for logical unit
	.... 1...	PLBCNCTI	IUCV CONNECT has been issued
	.... .1..	PLBINREC	Input received during purge
	.... ..1.	PLBPRMPT	Usability prompt message flag. For display terminals: CP READ received but prompt not sent. For hard-copy terminals: VSCS sent prompt to logical unit.
	1	PLBSDIAL	SNA-dialed logical unit
0D–0E	PLBNOM	Notify output manager	
0D	PLBNFLG1	NOM flag 1	
	1...	PLBCEND	Command end (READ)
	.1..	PLBRDTX	Redisplay timer has expired
	..1.	PLBSAC	Status area change required
	...1	PLBCLROA	CLEAR output area
	.... 1...	PLBCLRIA	CLEAR input area
	.... .1..	PLBRING	Ring the alarm
	.... ..x.		Zero
	1	PLBUNLK	Unlock keyboard
0E	PLBNFLG2	NOM flag 2	
	1...	PLBRDPND	Redisplay pending
	.xxx xxxx		Zero
0F	PLBOFLG1	Output manager flags	
	1...	PLBSDIAG	CMS DIAGNOSE on screen
	.1..	PLBPSMSG	Priority message on screen
	..1.	PLBOSRIP	SEND retry in progress
	...1	PLBOKLOK	Keyboard locked
	.... 1...	PLBOKPMI	Hard-copy terminal mask input area sent
	.... .1..	PLBOSWRT	CMS WRITE active (no input received)
	.... ..1.	PLBOCRRQ	Carriage return required before SEND (on hard-copy terminal)
	1	PLBOEXRQ	"!" required before SEND (on hard-copy terminal)
10	LBCSTATE	From VLB	
	1	LBCVSST	VTAM services state: 0 = Normal, 1 = Termination
11	VLBLRCD	LOGOFF reason codes	
	1...	VLBVTLO	VTAM LOGOFF
	.1..	VLBVRYI	VARY INACTIVE
	..1.	VLBHLTQ	HALT QUICK
	...1	VLBHLTC	HALT CANCEL
	.... 1...	VLBIOE	Unrecoverable I/O error
	.... .1..	VLBINERR	Internal error
	xx		Zero
12	WEBFUN code or zero (See "WEB Function Codes" on page 535.)		

13	TRCFLOW	Cleanup-module-dependent data	
14–17	VLBSWAP	The following four bytes are the individual fields that make up VLBSWAP:	
14	VLBFLG	VLB flags	
	1...	VLBBSPBB	Bracket state - pending begin bracket
	.1..	VLBBSPEB	Bracket state - pending end bracket
	..1.	VLBBSINB	Bracket state - in bracket
	...1	VLBSR	Application SEND or RECEIVE state: 1 = SEND state, 0 = RECEIVE state
	 xxxx		Zero
15	VLBFLG1	VLB flag 1	
	1...	VLBIOTR	Inbound data received since last SEND indicator
	.1..	VLBCLSD	CLSDST: 0 = required, 1 = not required
	..1.	VLBPCS	Purging chain state
	...1	VLBSIGNL	SIGNAL sent
	.... 1...	VLBCHGD	Change Direction required
	.... .1..	VLBRETR	Retry RECEIVE request
	.... ..1.	VLBRETS	Retry SEND request
	1	VLBSNDRD	Active SEND is a Read (ER mode only)
16	VLBFLG2	VLB flag 2	
	1...	VLBRPLB	RPL in use
	.1..	VLBCLPND	VTAM CLSDST pending
	..xx		Zero
	.... 1...	VLBBUSY	Printer busy
	.... .1..	VLBNAV	Printer not available
	.... ..1.	VLBOPCK	Operation check occurred
	1	VLBWAIT	Waiting for data or LUSTAT
17	VLBSENDF	Special SEND indicator	
	1...	VLBCLR	CLEAR required
	.1..	VLBSDT	SDT required
	..1.	VLBCANC	CANCEL required
	...1	VLBSIGRQ	SIGNAL required
	.... 1...	VLBCHECK	Bypass CHECK macro
	xxx		Zero
18–1B	VLBCID	Network address pair (CID)	
1C–1F	Address	DTIWEB if WEBFUN not zero, RPL if RPLREQ not zero	

Cleanup Module-Dependent Data

The trace flow flags are located at byte X'1C' of the cleanup error trace entries. The settings for these flags are as follows for each of the cleanup modules:

DTIPRBRK (ID='PBRK'):

01	TRXPURG	Logical unit in PLBINPRG (88) state
02	TRXNWEB	No input WEB

03	TRXLOGO	Logo error
04	TRXCOND	Conditional LOGOFF

DTIVPURG (ID='VPUR'): Trace flow flags are set as follows:

1...	Schedule failed		
.1..	Storage shortage		
..xx xxxx	Space reserved for module flow indicator		
	00	TRXNORM	Normal cleanup
	01	TRXNTERM	NOTERM entered
	02	TRXLTERM	Logical unit in termination
	03	TRXWORK	VSCS ignoring work to process request
	04	TRXNWORK	No work to process DTIVLOFF called
	05	TRXCONN	Logical unit in termination state with CCS connection
	06	TRXACTIV	Logical unit state is not termination with no connection
	07	TRXIGNOR	CCS has been notified or DTIPRBRK has set a timer to force purge. VSCS is waiting.

Logical Unit-Related Trace Entry (PFOR)

00	P	F	O	R
04	DTITAB address			
08	LBCSTATE	PLBLFLG1	PLBNFLG1	PLBOFLG1
0C	TABDTYPE	TABDSUB	LBCSTATE	VLBLRCD
10	VLBSWAP			
14 18	TABLUNM			
1C	X'FF'	MOD FLOW 1	MOD FLOW 2	STAT IND

Displacement:

00–03 Four-character module ID

04–07 TAB address for current request

08 LBCSTATE from PLB

1...	LBCINT	Internal mode if on (on=1). If LBCINT is on, see "LBCSTATE Values for PLBINxxx" on page 535. If LBCINT is off, the following values apply:
.11.	LBCFSSMF	Full-screen support mode flags
.1..	LBCFSAV	Screen available
..1.	LBCFSII	Full-screen input inhibited
		Description of settings for LBCFSSMF: B'00' CP (console) mode B'01' Virtual machine mode state between ATTN and READ, queue CP messages B'10' Virtual machine mode queue CP messages B'11' Virtual machine mode available for switch.
...1	LBCCOPY	Copy mode: on=1
.... 1...	LBCNAL	NAL flag: on=1
.... .11.	LBCHOLD	HOLDING flag: on=B'11'
.... .1..	LBCM10	MORE 10 flag: on=1
.... ..1.	LBCM50	MORE 50 flag: on=1
....1	LBCCMSD	Screen written with a CMS DIAGNOSE: on=1
....		No MORE/HOLDING state

09 PLBLFLG1

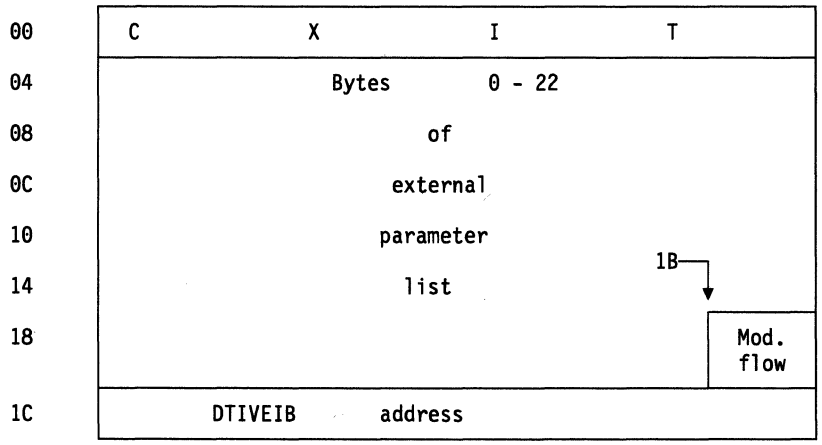
1...	PLBVS	VTAM services LOGOFF received bypass logo
.1..	PLBLGFC	WEBINLGF sent to CCS conditional LOGOFF

	..1.	PLBLGFU	WEBINLGF sent to CCS unconditional LOGOFF
	...1	PLBLERR	WEBINERR sent to CCS for hard error
	.... 1...	PLBLLERR	WEBINLER sent to CCS for interface error
	.... .1..	PLBLVERR	WEBINVER sent to CCS for internal logic error
	.... ..1.	PLBLSEVR	CP CCS SEVER received, VSCS must SEVER
	1	PLBNSTR	No storage indicator in LOGOFF processing
0A–0B	PLBNOM	Notify output manager	
0A	PLBNFLG1	NOM flag 1	
	1...	PLBCEND	Command end (READ)
	.1..	PLBRDTX	Redisplay timer has expired
	..1.	PLBSAC	Status area change required
	...1	PLBCLROA	CLEAR output area
	.... 1...	PLBCLRIA	CLEAR input area
	.... .1..	PLBRING	Ring the alarm
	.... ..x.		Zero
	1	PLBUNLK	Unlock keyboard
0B	PLBOFLG1	Output manager flags	
	1...	PLBSDIAG	CMS DIAGNOSE on screen
	.1..	PLBPSMSG	Priority message on screen
	..1.	PLBOSRIP	SEND retry in progress
	...1	PLBOKLOK	Keyboard locked
	.... 1...	PLBOKPMI	Hard-copy terminal mask input area sent
	.... .1..	PLBOSWRT	CMS WRITE active (no input received)
	.... ..1.	PLBOCRRQ	Carriage return required before SEND (on hard-copy terminal)
	1	PLBOEXRQ	"!" required before SEND (on hard-copy terminal)
0C	TABDTYPE		
	01 DTYPEDSP for a display		
	02 DTYPEKP for a keyboard/printer		
	03 DTYPEPRT for a printer		
0D	TABDSUB		
	01 VDEV3767 for a 3767		
	02 VDEV2741 for a 2741		
	03 VDEVTWX for a TWX/TTY/3101		
0E	LBCSTATE	VLB	
	1	LBCVSST	VTAM services state: 0 = Normal, 1 = Termination
0F	VLBLRCD	LOGOFF reason codes	
	1...	VLBVTLO	VTAM LOGOFF
	.1..	VLBVRYI	VARY INACTIVE
	..1.	VLBHLTQ	HALT QUICK
	...1	VLBHLTC	HALT CANCEL

	.... 1...	VLBIOE	Unrecoverable I/O error
	.... .1..	VLBINERR	Internal error
	xx		Zero
10–13	VLBSWAP	The following four bytes are the individual fields that make up VLBSWAP:	
10	VLBFLG	VLB flags	
	1...	VLBBSPBB	Bracket state-pending begin bracket
	.1..	VLBBSPEB	Bracket state-pending end bracket
	..1.	VLBBSINB	Bracket state-in bracket
	...1	VLBSR	Application SEND or RECEIVE state: 1 = SEND state, 0 = RECEIVE state
	 xxxx		Zero
11	VLBFLG1	VLB flag 1	
	1...	VLBIOTR	Inbound data received since last SEND indicator
	.1..	VLBCLSD	CLSDST: 0 = required, 1 = not required
	..1.	VLBPCS	Purging chain state
	...1	VLBSIGNL	SIGNAL sent
	.... 1...	VLBCHGD	Change Direction required
	.... .1..	VLBRETR	Retry RECEIVE request
	.... ..1.	VLBRETS	Retry SEND request
	1	VLBSNDRD	Active SEND is a Read (ER mode only)
12	VLBFLG2	VLB flag 2	
	1...	VLBRPLB	RPL in use
	.1..	VLBCLPND	VTAM CLSDST pending
	..xx		Zero
	.... 1...	VLBBUSY	Printer busy
	.... .1..	VLBNAV	Printer not available
	.... ..1.	VLBOPCK	Operation check occurred
	1	VLBWAIT	Waiting for data or LUSTAT
13	VLBSENDF	Special function flags	
	1...	VLBCLR	CLEAR required
	.1..	VLBSDT	SDT required
	..1.	VLBCANC	CANCEL required
	...1	VLBSIGRQ	SIGNAL required
	.... 1...	VLBCHECK	Bypass CHECK macro
	xxx		Zero
14–1B	TABLUNM	Logical unit name being forced	
1C		X'FF' to indicate data trace entry	
1D		Module flow indicator 1 done by SPECIALX	
	1...	SPECIALX called	
	.1..	CONECTYS called	
	..1.	CONECTNO called	
	...1	COPYSIT called	
	.... 1...	DTIPRCLP called	
	.... .1..	DTIPRBRK called	
	.... ..1.	DTIPRSVR called	

	1	DTIPRLGF called	
1E		Module flow indicator 2 done by SPECIALX	
	1...	SPECIALX called	
	.1..	CONECTYS called	
	..1.	CONECTNO called	
	...1	COPYSIT called	
	.... 1...	DTIPRCLP called	
	.... .1..	DTIPRBRK called	
	.... ..1.	DTIPRSVR called	
	1	DTIPRLGF called	
1F		Queue status indicators	
	1...	LUNORMAL	Logical unit control blocks on normal queue
	.1..	LUREC	Logical unit control blocks on recovery queue
	..1.	LUPQUEUE	PS PIB on PS dispatcher queue
	...1	LUVQUEUE	VS PIB on VS dispatcher queue
	.... 1...	LUBFKPIB	WEBCPBRK on PS PIB work queue
	.... .1..	LUBRKLBC	WEBCPBRK on LBC pending queue
	.... ..1.	LUVSBRK	WEBVSBRK found on one of the work queues
	1	LUNTF60	WEBNTF60 found on PS PIB

Data-Related Error Trace Entry (CXIT)



Displacement:

00–03	Characters 'CXIT'		
04–05	Path ID		
06	IPFLAGS1		
	1...	IPALL	QUIESCE, RESUME, and SEVER ALL
	.1..	IPQUSCE	Connect in QUIESCE mode
	..1.	IPPRTY	Priority message, REPLY required
	...1	IPNORPY	One-way protocol
	.... 1...	IPCPENTY	Entry from CP

	.... .1..	IPFGMID	Message ID specified
	.... ..1.	IPFGPID	Path ID specified
	1	IPFGMCL	Message class specified
07	IPTYPE	External interrupt code	
08–0B	IPMSGID	Message ID	
0C–0F	IPTRGCLS	Target class	
0C–0D	IPAUDIT	Audit trail as follows:	
0C	IPAUDIT1	Audit trail byte 1	
	1...	IPADRPLE	REPLY too long for buffer
	.1..	IPADSNPX	Protection exception on SEND buffer
	..1.	IPADSNAX	Address exception on SEND buffer
	...1	IPADANPX	Protection exception on ANSWER buffer
	.... 1...	IPADANAX	Address exception on ANSWER buffer
	.... .1..	IPADRJCT	Message was rejected
	xx		Zero
0D	IPAUDIT2	Audit trail byte 2	
	1...	IPADRCPX	Protection exception on RECEIVE buffer
	.1..	IPADRCAX	Address exception on RECEIVE buffer
	..1.	IPADRPPX	Protection exception on REPLY buffer
	...1	IPADRPAX	Address exception on REPLY buffer
	.... 1...	IPADSVRD	Path was severed
	xxx		Zero
0E–0F		Zero	
10–13	IPBFADR1	Address of buffer 1	
14–15		Zero	
16–17	IPBFLN1	Length of buffer 1	
18–1A	IPSRCCLS	Source class	
1B	Module flow	(failure point)	
		01	VSCS abnormal termination in progress
		02	Pending CONNECT received - not supported
		03	Error dequeueing VEIB or no VEIBs received
		04	Error enqueueing VEIB on global message queue
		06	Enqueue or dequeue error found
		07	Schedule with priority failed
		08	Schedule failed
		09	No match found for path ID in path ID table
		0A	Control block passed not a valid DTISAB or DTITAB
		0B	Storage shortage obtaining additional VEIBs
		0C	Storage shortage obtaining additional VEIBs
1C–1F	VEIB address		

ABEND Trace Entry (ABND)

00	A	B	N	D
04	Task ID		Abend code	
08 0C	Failing PSW			
10	SDWA address			
14	Register 13			
18	DTITAB address			
1C	X'FF'	RTNCODE	SABABIPF	Dump number

Displacement:

00–03	Characters "ABND"		
04–05	Task ID		
06–07	Abend code		
08–0F	Failing PSW		
10–13	Address of SDWA		
14–17	Register 13 at time of failure		
18–1B	DTITAB address or zeros		
1C	Data record indicator X'FF'		
1D	RTNCODE	Return code from the module	
1E	SABABIPF	Abend in progress flags	
	1...	SABPABIP	Presentation services abend in progress
	.1..	SABUABIP	Utility services abend in progress
	..11 1111	Reserved	
1F	Dump number 0–255		

Internal Error Trace Entry (INTR)

00	I	N	T	R
04	Occurrence value			
08	Last 4 characters of module name			
0C	Return register first caller			
10	Return register second caller			
14	Register 13			
18	DTITAB address			
1C	X'FF'	RTNCODE	SABABIPF	DUMP NUMBER

Displacement:

00–03	Characters "INTR"		
04–07	Occurrence value		
08–0B	Last four characters of module name		
0C–0F	First caller's return address		
10–13	Second caller's return address		
14–17	Register 13 at time of failure		
18–1B	DTITAB address or zeros		
1C	Data record indicator X'FF'		
1D	RTNCODE	Return code from the module	
1E	SABABIPF	Abend in progress flags	
	1...	SABPABIP	Presentation services abend in progress
	.1..	SABUABIP	Utility services abend in progress
	..11 1111	Reserved	
1F	Dump number 0–255		

Module trace flow indicators

DTIPRBRK: The settings for these flags are as follows for each of the cleanup modules:

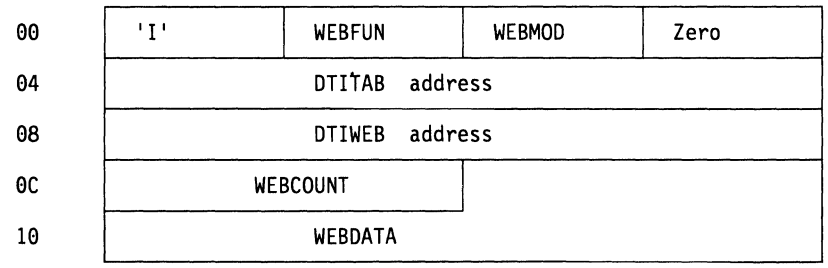
00	TRXNORM	Normal cleanup
01	TRXPURG	Logical unit in PLBINPRG (88) state
02	TRXNWEB	No input WEB
03	TRXLOGO	Logo error
04	TRXCOND	Conditional LOGOFF

DTIVPURG: Trace flow flags are set as follows:

1...		Schedule failed
.1..		Storage shortage
..xx xxxx		Space reserved for module flow indicator
00	TRXNORM	Normal cleanup
01	TRXNTERM	NOTERM entered
02	TRXLTERM	Logical unit in termination
03	TRXWORK	VSCS ignoring work to process request
04	TRXNWORK	No work to process DTIVLOFF called
05	TRXCONN	Logical unit in termination state with CCS connection
06	TRXACTIV	Logical unit state is not termination with no connection
07	TRXIGNOR	CCS has been notified or DTIPRBRK has set a timer to force purge. VSCS is waiting.

VSCS Data Trace Record Formats

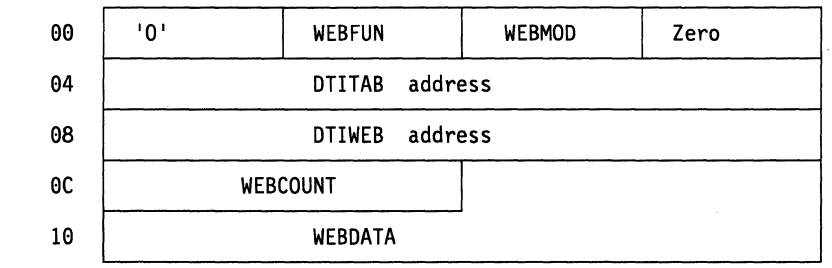
Input Trace Entry



Displacement:

00 Character "I" for Input
01 WEB function code
02 WEB mode
  00 Console (No MORE/HOLDING)
  01 CMS
  02 Full-screen
  04 Internal
03 Zero
04–07 DTITAB address
08–0B DTIWEB address
0C–0D WEBCOUNT
0E–xx WEBDATA (18 bytes for internal or 221 bytes for external)

Output Trace Entry



Displacement:

00 Character "O" for Output
01 WEB function code
02 WEB mode
  00 Console (No MORE/HOLDING)
  01 CMS
  02 Full-screen
  04 Internal
03 Zero
04–07 DTITAB address
08–0B DTIWEB address
0C–0D WEBCOUNT
0E–xx WEBDATA (18 bytes for internal or 221 bytes for external)

VSCS Dispatcher Trace Record Formats

When the dispatcher trace option is active, VSCS traces all traffic through the VSCS state managers. This trace can be limited to one or several logical units by starting the VSCS dispatcher trace for selected logical units. Use the VSCS TRACEON (DISP command to start the VSCS dispatcher trace. See "VSCS Tracing Commands" in *VTAM Operation* for additional information on the VSCS dispatcher trace. The dispatcher trace produces two types of record formats:

- *Logical-unit-related* for traffic associated with logical units
- *Global* for traffic not associated with a logical unit.

The following examples show the format of records created for each event.

Dispatcher Trace Entry for a Logical Unit: Entry 1

00	Header	LBCSTTUS	Zero	
04	DTITAB address			
08	DTIWEB address			
0C	WEBFUN	WEBMOD	WEBVSRPS	Zero
10	LBCSTATE	PLBFSSMF	PLBLFLG1	PLBFLG1
14	PLBPRLBC	PLBOFLG1	PLBNOM	
18	PLBEDIT	PLBIAFLG	PLBDFLG1	Zero
1C	Last four characters of module name			

Displacement:

00 Header ID

- D No module ID found or address of module is zero
- P Presentation services dispatcher
- V VTAM services dispatcher

01 LBCSTTUS and current task running

1...	LBCSIP	1=busy, 0=not busy
.xxx x...		Zero
.... .1..		Unknown task running
.... ..1.		Presentation services task running
.... ..11		VTAM services task running

02–03 Zero

04–07 DTITAB Address of TAB passed

08–0B DTIWEB Address of current WEB

0C WEBFUN WEB function code from input WEB

0D WEBMODE WEB mode from input WEB

- 00 Console (No MORE/HOLDING)
- 01 CMS
- 02 Full-screen
- 04 Internal

0E WEBVSRPS VTAM services response to SEND request from presentation services

	01	WEBVSOPC	Operation check
	02	WEBVSLFU	Unconditional LOGOFF
	03	WEBVSERR	Unrecoverable I/O error
	04	WEBVSBSY	Printer busy
	05	WEBVSIRQ	Intervention required
	06	WEBVSNAV	Printer not available
	07	WEBTWXER	TWX READ error, sense code 0827
	08	WEBVSAOK	Successful completion
	0C	WEBVSVER	Unrecoverable internal error
	0D	WEBVSPSL	Presentation space integrity lost
	0E	WEBVSCNL	SEND canceled by cancel key
	0F	WEBVSCC1	Reflect attention, CC1 on SIO
	10	WEBVSECU	Equipment check and unit specify
	11	WEBVSDCU	Data check and unit specify
	12	WEBVSEC	Equipment check
	13	WEBVSCRJ	Command reject
	14	WEBVSDC	Data check
	15	WEBVSCC	Command check
0F		Zero	
10	LBCSTATE from PLB		
	1...	LBCINT	Internal mode if on (on=1).
	.11.	LBCFSSMF	Full-screen support mode flags
	.1..	LBCFSAV	Screen available
	..1.	LBCFSII	Full-screen input inhibited: B'00' CP (Console) mode B'01' Virtual machine mode state between ATTN and READ, queue CP messages B'10' Virtual machine mode queue CP messages B'11' Virtual machine mode available for switch
	...1	LBCCOPY	Copy mode: on=1
	.... 1...	LBCNAL	NAL flag: on=1
	.... .11.	LBCHOLD	HOLDING flag: on=B'11'
	.... .1..	LBCM10	MORE 10 flag: on=1
	.... ..1.	LBCM50	MORE 50 flag: on=1
	1	LBCCMSD	Screen written with a CMS DIAGNOSE: on=1
	....		No MORE/HOLDING state (all bits off)
11	PLBFSSMF	Full-screen support mode flags	
	1...	PLBFPA1	PA1 key received switch mode
	.1..	PLBFSRFC	Mode switch required (FSSM to Console)
	..1.	PLBFSAFC	Switch available (FSSM to Console)
	...1	PLBFCNMP	Console message pending
	.... 1...	PLBFVSIP	VTAM services SEND in progress
	.... .1..	PLBFCLGF	Conditional LOGOFF received
	.... ..1.	PLBFCMRR	CP message pending, response required

	1	PLBCMSEW	Screen has just been erased
12	PLBLFLG1	VTAM services LOGOFF flags for presentation services	
	1...	PLBLVS	VTAM services LOGOFF received, bypass logo
	.1..	PLBLGFC	WEBINLGF sent to CCS conditional LOGOFF
	..1.	PLBLGFU	WEBINLGF sent to CCS unconditional LOGOFF
	...1	PLBLERR	WEBINERR sent to CCS for hard error
	.... 1...	PLBLLERR	WEBINLER sent to CCS for interface error
	.... .1..	PLBLVERR	WEBINVER sent to CCS for internal logic error
	.... ..1.	PLBLSEVR	CP CCS SEVER received, must SEVER
	1	PLBLNSTR	No storage indicator in LOGOFF
13	PLBFLG1	PLB flags	
	1...	PLBLGO	Logo mode logical unit
	.1..	PLBINAR	APL/TEXT starting in input area
	..1.	PLBLFLNP	LOGON from logo not permitted
	...1	PLBCONCT	Current count incremented
	.... 1...	PLBCNCTI	IUCV CONNECT issued
	.... .1..	PLBINREC	Input received during purge processing
	.... ..1.	PLBPRMPT	Usability prompt message flag
	1	PLBSDIAL	SNA-dialed logical unit
14	PLBPRLBC	Previous logical unit LBCSTATE	
15	PLBOFLG1	Output manager flags	
	1...	PLBSDIAG	CMS DIAGNOSE on screen
	.1..	PLBPSMSG	Priority message on screen
	..1.	PLBOSRIP	SEND retry in progress
	...1	PLBOKLOK	Keyboard locked
	.... 1...	PLBOKPMI	Keyboard printer mask input area sent
	.... .1..	PLBOSWRT	CMS WRITE active (no input received)
	.... ..1.	PLBOCRRQ	Carriage return required before SEND on hard-copy terminal
	1	PLBOEXRQ	'!' required before SEND on hard-copy terminal
16–17	PLBNOM	Notify output manager	
16	PLBNFLG1	NOM flag 1	
	1...	PLBCEND	Command end (READ)
	.1..	PLBRDTX	Redisplay timer has expired
	..1.	PLBSAC	Status area change required
	...1	PLBCLROA	Clear output area
	.... 1...	PLBCLRIA	Clear input area
	.... .1..	PLBRING	Ring the alarm
	.... ..x.		Zero
	1	PLBUNLK	Unlock keyboard
17	PLBNFLG2	NOM flag 2	
	1...	PLBRDPND	Redisplay pending
	.xxx xxxx		Zero

18	PLBEDIT	Editing flags	
	1...	PLBNCONV	User-translated output data
	.1..	PLBPPWRT	Password prompt message
	..xx		Zero
	.... 1...	PLBHIEKO	Highlight input redisplay
	.... .1..	PLBEDPTI	Do not print or do not display
	.... ..1.	PLBNOEKO	Do not redisplay input on next output line
	1	PLBEDCUP	Convert to upper case
19	PLBIAFLG	Input manager attention handling flags	
	1...	PLBMATTN	Multiple attention received
	.1..	PLBSATTN	Single attention received
	..1.	PLBIPA1	Attention received in MORE
	...1	PLBBRKNN	CP TERMINAL BRKKEY - NONE in effect
	.... 1...	PLBUNSRD	Unsolicited READ received
	xxx		Zero
1A	PLBDFLG1	Device manager flags	
	1...	PLBDASS	Display has alternate size
	.1..	PLBDNEW	Bit is on for a 3278, bit is off for a 3277
	..1.	PLBNOVCK	Do not validity check data
	...1 11..	PLBDCH	Display has new color or highlight functions, no B'000'
	...1	PLBDXTDS	Extended data stream available
	.... 1...	PLBDXCLR	Extended color available
	.... .1..	PLBDXHLI	Extended highlight available
	.... ..1.	PLBDATTN	Printer attention
	1	PLBDMASK	Password mask
1B		Zero	
1C–1F	Last four characters of module ID		

Dispatcher Trace Entry for a Logical Unit: Entry 2

00	'Z'	Zero	LU path ID	
04	LBCSTATE	VLBFLG	VLBFLG1	VLBFLG2
08	VLBSENDF	VLBERFLG	VLBLRCD	VLBERCVY
0C	V T A M		C I D	
10	Zero			
14				
18				
1C				

Displacement:

00	Header ID 'Z'
01	Zero

02–03 Logical unit path ID

04	LBCSTATE	VLB	
	1	LBCVSST	VTAM services state: 0=Normal, 1=Termination
05	VLBFLG	VLB flags	
	1...	VLBBSPBB	Bracket-state pending begin bracket
	.1..	VLBBSPEB	Bracket-state pending end bracket
	..1.	VLBBSINB	Bracket-state in bracket
	...1	VLBSR	Application SEND or RECEIVE state: 1=SEND state, 0=RECEIVE state
	 xxxx		Zero
06	VLBFLG1	VLB flag 1	
	1...	VLBIOTR	Inbound data received since last SEND
	.1..	VLBCLSD	CLSDST not required
	..1.	VLBPCS	Purging chain state
	...1	VLBSIGNL	SIGNAL sent
	.... 1...	VLBCHGD	Change Direction required
	.... .1..	VLBRETR	Retry RECEIVE request
	.... ..1.	VLBRETS	Retry SEND request
	1	VLBSNDRD	Active SEND is a Read (ER mode only)
07	VLBFLG2	VLB flag 2	
	1...	VLBRPLB	RPL in use
	.1..	VLBCLPND	VTAM CLSDST pending
	..xx		Zero
	.... 1...	VLBBUSY	Printer busy
	.... .1..	VLBNAV	Printer not available
	.... ..1.	VLBOPCK	Operation check occurred
	1	VLBWAIT	Waiting for data or LUSTAT
08	VLBSENDF	Special SEND indicator	
	1...	VLBCLR	CLEAR required
	.1..	VLBSDT	START DATA TRAFFIC (SDT) required
	..1.	VLBCANC	CANCEL required
	...1	VLBSIGRQ	SIGNAL required
	.... 1...	VLBCHECK	Bypass CHECK macro
	xxx		Zero
09	VLBERFLG — Exception response flags		
	1...	VLBPSHLD	DTIXMT pending (PS held)
	.1..	VLBERPND	Error recovery pending
	..1.	VLBERRIP	Error recovery in progress
	...1	VLBERPRG	Permanent error: purge logical unit
	.... 1...	VLBPRGDN	Purge complete
	.... .1..	VLBERPRM	Permanent error
	.... ..1.	VLBEREXT	Exit received control before return
	1	VLBRCVNO	Do not reissue RECEIVE SPECIFIC for logical unit
0A	VLBLRCD	LOGOFF reason codes	
	1...	VLBVTLO	VTAM LOGOFF

	.1..	VLBVRYI	VARY INACTIVE
	..1.	VLBHLTQ	HALT QUICK
	...1	VLBHLTC	HALT CANCEL
	.... 1...	VLBIOE	Unrecoverable I/O error
	.... .1..	VLBINERR	Internal error
	xx		Zero
0B	VLBERCVY		
	1...	VLBERCLR	CLEAR requested
	.1..	VLBERBRK	Bracket error occurred
	..1.	VLBERSDT	START DATA TRAFFIC required
	...1	VLBERSIG	SIGNAL required
	.... 1...	VLBERCNL	CANCEL received
	.... .1..	VLBERCDR	Change Direction required
	.... ..1.	VLBERPSL	Presentation space integrity lost
	1	VLBERHRD	Hard error occurred
0C–0F	VTAM CID (OAF/DAF)		
10–1F		Zero	

Dispatcher Trace Entry for a Global Connection

00	Header ID	Task/Busy	SABIQECB	
04	Address passed			
08	DTIWEB address			
0C	SABFLG1	SABABIPF	SABTFLG1	SABRFLG1
10	WEBFUN	WEBMODE	WEBCHAR	WEBEDIT
14	WEBFLAGS	WEBCPFLG	WEBLINE	
18	WEBCURSR		WEBTABCH	WEBRSVSA
1C	Last four characters of module name			

Displacement:

0	Trace record header ID	
	D	No module ID found or module address is zero
	P	Presentation services dispatcher
	V	VTAM services dispatcher
1	LBCSTTUS and current task running	
	...1	Address of SAB included
	.xxx x...	Zero
	.... ..1.	Presentation services task running
	11	VTAM services task running
	.... .1..	Unknown task running
02–03	SABIQECB	First two bytes of SABIQECB event control block (ECB)
04–07		Address passed
08–0B		Address of WEB
0C	SABFLG1	

	1...	SABDBCSC	VSCS and CCS are DBCS capable
	.1..	SABVSCSV	VSCS running in VTAM machine
	..1.	SABDUIP	Dump in progress
	...1	SABCSOPN	VSIUCV open
	.... 1...	SABBUFDQ	DTISGETB dequeue type: FIFO=1, LIFO=0
	.... .1..	SABGETFL	GETMAIN failed
	....1.	SABNEWLG	New logo format allowed
	1	SABVEIBS	VEIB shortage has occurred
0D	SABABIPF	Abend-in-progress flags	
	1...	SABPABIP	Presentation services abend in progress
	.1..	SABUABIP	Utility services abend in progress
	..xx xxxx		Zero
0E	SABTFLG1	Global termination flags	
	1...	SABTETPE	TPEND in progress
	.1..	SABTESIQ	Termination scheduled by DTIIQUIT
	..1.	SABTETIP	Termination in progress
	...1	SABTEATP	Abnormal termination in progress
	 xxxx		Zero
0F	SABRFLG1	Global recovery flags	
	1...	SABVSCTL	VTAM services task in control
	.1..	SABPSCTL	Presentation services task in control
	..1.	SABTICTL	Timer exit in control
	...1	SABCXCTL	Communications services exit in control
	 xxxx		Zero
10	WEBFUN	WEB function code. This is explained in "WEB Function Codes" on page 535.	
11	WEBMODE	Mode: 00 Console 01 CMS 02 Full-screen 04 Internal	
12	WEBCHAR	Character set (set by CCS for Console and CMS modes) 00 EBCDIC character set 01 APL character set 02 TEXT character set 04 DBCS character set 05 DBCS and APL character set 06 DBCS and TEXT character set	
13	WEBEDIT	Editing characteristics	
	1...	WEBIMMED	Mode switch immediately
	.1..	WEBPETCM	'PRESS ENTER TO CONTINUE' message
	..1.	WEBCLRSC	CP clear screen
	...x		Zero
	.... 1...	WEBHIEKO	Highlight input display
	.... .1..	WEBEDPTI	Do not print or do not display

	.... ..1.	WEBNOEKO	Do not redisplay input on next output line
	1	WEBEDCUP	Convert to uppercase
14	WEBFLAGS		
	x...	WEBRSV1	Zero
	.1..	WEBCONMD	Request for display in CONMODE 3270
	..1.	WEBLED	Limited edit diagnose
	...1	WEBANFSW	Allow any full-screen WRITE; first full-screen WRITE after reset.
	.... 1...	WEBDIAL	SNA-dialed logical unit
	.... .1..	WEBNOMOR	CMS ERASE or WRITE no "MORE" requested
	.... ..1.	WEBPRMPT	WEBDATA contains pre-logon message
	1	WEBPASPA	Pass PA1 (FSSM) as data to application
15	WEBCPFLG	CCS status flags for VSCS	
	1...	WEBCPNCR	Do not imbed carriage return in data stream (on hard-copy terminal)
	.1..	WEBCPPRI	Priority flag
	..1.	WEBCPALM	Ring the alarm
	..1.	WEBSAHLD	HOLDING
	...1	WEBCPMDE	CP-generated bit = 1
	...1	WEBSAMOR	MORE VM-generated bit = 0
	.... 1...	WEBCPHLW	Highlight WRITE requested
	.... .1..	WEBCPRSP	Indicates CCS response
	.... .1..	WEBSARSP	Indicates VSCS response
	.... ..1.	WEBCPRRR	CCS request, requires response
	.... ..1.	WEBSARRR	VSCS request, requires response
	1	WEBCPRNR	CCS request, requires no response
	1	WEBSARNR	VSCS request, requires no response
16–17	WEBLINE	Line number for CMS WEBCMWRT	
16	WEBLAID	Logical attention identifier	
17	WEBNLLOS	Number of lines left on screen	
18–19	WEBCURSR	Cursor position may be a relative address or a buffer address	
1A	WEBTABCH	Tab (not DTITAB) character	
1B	WEBRSVSA	Reserved for VSCS use	
	1...	WEBCPRDS	DTIWEB is a redisplay DTIWEB
	.1..	WEBNOVCK	Do not validity check data
	..1.	WEBATTNW	Attention WRITE (VSCS only) yes = 1
	...1	WEBINLIP	Logon in progress
	.... x...		Zero
	.... .1..	WEBCHDIR	Change Direction required
	.... ..1.	WEBMASKW	WRITE inhibit mask
	x		Zero
1C–1F	Last four characters of module ID (or entry address)		

VSCS Operator Trace Record Formats

MSGS Trace Entry (MSGS)

The MSGS trace entry provides an indication in the VSCS trace table (or externally) when a message occurs. A single record will contain the message number and any variable text. This information can provide a good starting place in the VSCS trace for a problem that occurs after or just before a VSCS operator message is issued.

This trace will be active when the VSCS trace option OPER is activated or the ALL trace option is used to start the trace.

The content of each record would be as follows:

00	M	S	G	S
04 08	Time Stamp			
0C	Message ID			
10	Return address			
14	Variable 1			
18	Variable 2			
1C	Variable 3			

Displacement:

- 00–03 Characters "MSGS"
- 04–0B Timestamp when message was issued
- 0C–0F Message ID DTIxxxx
- 10–13 Caller return address
- 14–17 Message variable 1
- 18–1B Message variable 2
- 1C–1F Message variable 3

Following is an example of formatted records:

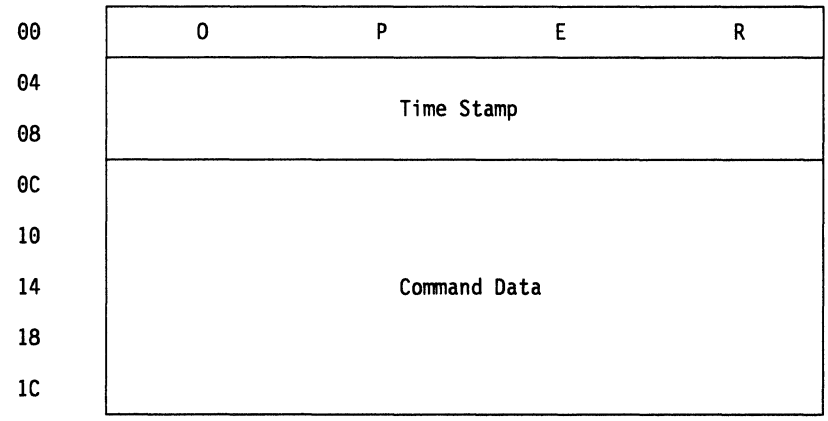
MSGS TIMESTAMP A5F726FF 841B4101 MSGI S15I CALLER 402B344C
VARIABLES 002B3980 FAB3A9C8 D4D7D3C5

OPER Trace Entry (OPER)

The OPER trace entry provides an indication in the VSCS trace table (or externally) when a VSCS operator command is issued. As much of the command text as will fit into one record will be recorded. This information can provide a good starting place in the VSCS trace for a problem that occurs after or just before a VSCS operator command is issued.

This trace is active when the VSCS trace option OPER is activated or the ALL trace option is used.

The content of each record would be as follows:



Displacement:

00–03	Characters "OPER"
04–0B	Time stamp when message was issued
0C–1F	Command data

Formatted records look like:

```
OPER  TIMESTAMP A5F726FF 84093501  CMD DATA S
```

VSCS Storage Statistics Trace Record Format

STAT Trace Entry (STAT)

The STAT trace entry records storage statistics when the storage manager CHECK function is processed. This trace provides a method for monitoring VSCS storage utilization over a period of time. The information can be used to determine the correct storage size of the VSCS virtual machine and determine the storage costs of various VSCS initialization parameters and start options.

This trace is active when the VSCS trace option STAT is activated.

The content of each record would be as follows:

00	S	T	A	T
04	STATLVL	STATVTMM	STATSSO	STATVES
08	STATONL	STATFSR	STATXTDS	
0C			STATDPAC	
10	STATDEX	STATAQLM	STATXTKP	
14		STATKPAC		STATKEX
18	STATRECS	STATER	STATCHKT	
1C			STATRELT	
20			STATVTMS	
24			STATCPS	
28			STATMCPS	
2C			STATVSAM	
30	STATVEIB			
34	STATBLKM	STATRPLS	STATRECL	
38			STATTSKR	
3C	STATTABR		STATGET	
40			STATGETS	
44			STATSSZ	
48			STATGETF	
4C			STATSWAP	
50			STATFREM	
54			STATFRES	
58			STATBDC	
5C			STATBDS	
60			STATHW(1)	

64		STATVS(1)
68		STATAL(1)
6C		STATAV(1)
70		STATTL(1)
74		STATHW(2)
78		STATVS(2)
7C		STATAL(2)
80		STATAV(2)
84		STATTL(2)
88		STATHW(3)
8C		STATVS(3)
90		STATAL(3)
94		STATAV(3)
98		STATTL(3)
9C		STATHWM(1)
A0		STATVTM(1)
A4		STATSEG(1)
A8		STATTSG(1)
AC		STATHWM(2)
B0		STATVTM(2)
B4		STATSEG(2)
B8		STATTSG(2)
BC		STATHWM(3)
C0		STATVTM(3)
C4		STATSEG(3)
C8		STATTSG(3)

Displacement:

00–03	Characters "STAT"	
04	STATLVL	VSCS level
05	STATVTMM	VSCS in VTAM machine
06	STATSSO	Storage shortage occurred
07	STATVES	VEIB shortage
08	STATONL	ONE LOGO value
09	STATFSR	FS READ BUFFER value
0A–0B	STATXTDS	DPXMTL value

0E–0F	STATDPAC	DPACE value
10	STATDEX	DEXIT value
11	STATAQLM	AQLIMIT value in K (1024 bytes)
12–14	STATXTKP	KPXMTL value
15–16	STATKPAC	KPACE value
17	STATKEX	KEXTI value
18	STATRECS	RECEIVE SPECIFIC value
19	STATER	Exception response active
1A–1B	STATCHKT	Storage check time
1E–1F	STATRELT	Storage release time
22–23	STATVTMS	Number of VTAM sessions
26–27	STATCPS	Number of CP connections
2A–2B	STATMCPS	Maximum CP connections
2E–2F	STATVSAM	VSAMLM value
30–33	STATVEIB	Number of VEIBs
34	STATBLKM	BLKMULT value
35	STATRPLS	Number of RPLs
36–39	STATRECL	RECBFL value
3A–3B	STATTSKR	TASK retry value
3C–3D	STATTABR	LU retry value
3E–3F	STATGET	Number of GETMAIN's issued
42–43	STATGETS	Number of special GETMAINs
46–47	STATSSZ	Size of special GETMAINs
4A–4B	STATGETF	Number of GETMAIN failures
4E–4F	STATSWAP	Number of SWAP's done
52–53	STATFREM	Number of FREEMAIN's issued
56–57	STATFRES	Number of special FREEMAINs
5A–5B	STATBDC	Count of bad segments
5E–5F	STATBDS	Size of bad segments
62–63	STATHW(1)	Highwater mark
66–67	STATVS(1)	VTAM sessions
6A–6B	STATAL(1)	Allocated segments
6E–6F	STATAV(1)	Available segments
72–73	STATTL(1)	Total storage in bytes
76–77	STATHW(2)	Highwater mark
7A–7B	STATVS(2)	VTAM sessions
7E–7F	STATAL(2)	Allocated segments
82–83	STATAV(2)	Available segments
86–87	STATTL(2)	Total storage in bytes
8A–8B	STATHW(3)	Highwater mark
8E–8F	STATVS(3)	VTAM sessions
92–93	STATAL(3)	Allocated segments
96–97	STATAV(3)	Available segments
9A–9B	STATTL(3)	Total storage in bytes
9E–9F	STATHWM(1)	Highwater mark
A2–A3	STATVTM(1)	VTAM sessions
A6–A7	STATSEG(1)	Number of segments
AA–AB	STATTSG(1)	Total storage in bytes

AE–AF STATHWM(2) Highwater mark
B2–B3 STATVTM(2) VTAM sessions
B6–B7 STATSEG(2) Number of segments
BA–BB STATTSG(2) Total storage in bytes
BE–BF STATHWM(3) Highwater mark
C2–C3 STATVTM(3) VTAM sessions
C6–C7 STATSEG(3) Number of segments
CA–CB STATTSG(3) Total storage in bytes

Formatted records look like:

```
STAT   VSCSLVL  07        VTAM     N         SHORTAGE N         VSHORT   N         ONELOGO  N         FSREAD   Y
       DPXMTL   0000079C  DPACE    0000      DEXIT    N         AQLIMIT  10        KPXMTL   00011C00  KPACE    000A
       KEXIT    .         SPEC     N         SCHED    N         STCHKTM  0000000A  STRELTM  00000000
       VTAMSES  00000001  CPCON    00000001  MAXCPCON 00000001  VSAMLM   000A      VEIBS    0000000F
       BLKMULT  01        RPLNUM   10        RCVBFRL  00000059  TSKRTRY  000A      LURTRY   0005
       GETMAIN  00000019  SGETMN   00000000  SGETSIZE 00000000  GETFAILS 00000000  SWAPS    00000002
       FREEMAIN 00000003  SFREEMN  00000000  BADCOUNT 00000000  BADSIZE  00000000
       HW/MARK  0000000A  VTAMSES  00000001  ALLOCATE 0000000A  AVAILABL 00000007  TOTAL    0000C000
       HW/MARK  00000002  VTAMSES  00000001  ALLOCATE 00000002  AVAILABL 00000001  TOTAL    00002000
       HW/MARK  0000000A  VTAMSES  00000001  ALLOCATE 00000008  AVAILABL 00000004  TOTAL    0000B000
       HW/MARK  00000002  VTAMSES  00000001                     AVAILABL 00000000  TOTAL    00000000
       HW/MARK  00000000  VTAMSES  00000000                     AVAILABL 00000000  TOTAL    00000000
       HW/MARK  00000000  VTAMSES  00000000                     AVAILABL 00000000  TOTAL    00000000
```

VSCS Storage Trace Record Formats

FREB Trace Entry

00	F	R	E	B
04	DTIPDB address			
08	Block address			
0C	DTISMP identifier		DTIEHDR identifier	
10	POOLID	Zero		
14	Return address			
18	Return address			
1C	Zero			

Displacement:

00–03 Characters "FREB"
04–07 DTIPDB address
08–0B Block being freed
0C–0D SMP header
0E–0F Block ID (DTIEHDR identifier)
10 Storage pool number
- 1 Presentation services
- 2 VTAM services
- 3 VTAM Exit Services

11–13 Zero
14–17 First caller's return address
18–1B Caller's caller's (second caller's) return address

1C–1F Zero

GETB Trace Entry

00	G	E	T	B
04	DTIPDB address			
08	Block address			
0C	Request size (3 bytes)			CLEAR size
10	CLEAR size (bytes 2 and 3)		POOLID	Return code
14	Return address			
18	Return address			
1C	State			PDB flags

Displacement:

00–03 Characters "GETB"
04–07 DTIPDB address
08–0B Block address obtained
0C–0E Size requested
0F–11 Size to be cleared
12 Storage pool number
- 1 Presentation services
- 2 VTAM services
- 3 VTAM Exit Services

13 Module return code
14–17 First caller's return address
18–1B Caller's caller's (second caller's) return address
1C–1E State
- ENA Caller running enabled
- DIS Caller running disabled

1F PDB flags
- 1... .... Storage shortage
- .1.. .... Used next PDB
- ..1. .... Forced to use static storage
- ...1 .... Checked all PDBs, no storage available
- .... 1... Bad PDB count
- .... .1.. PDB lock not available
- .... ..xx Zero

VSCS VTAM Exit Trace Record Formats

If you are running the default VSCS trace and you have activated the trace with an operator command, or by starting VSCS with the VTAM exit trace active, VSCS traces all inbound traffic through the VSCS VTAM exits. You can limit the VTAM exit trace to a selected set of logical units by starting the VSCS VTAM exit trace for those logical units.

You can produce the VTAM exit trace in two formats:

- *Normal trace record* if the request is a valid VSCS logical unit

- *Data trace record* if the request or response was not valid, or if the data traced is different from the normal VTAM exit trace record.

Below is the format for the records created for each event.

VSCS VTAM Exit Trace: Entry 1

00	'V'	Module ID	
04	DTITAB address		
08	IFGRPL address		
0C	RPLREQ	RTNCODE	V T A M N A U
10	RPLFDBK2		RPLSSNSI or RPLUSNSI
14	LBCSTATE	VLBERFLG	LU path ID
18	V L B S W A P		
1C	VLBLRCD	V X I T D A T A	

Created by:

- DTIVCLSX
- DTIVDFAX
- DTIVLOGX
- DTIVNSEX
- DTIVOPNX
- DTIVRECX
- DTIVRESX
- DTIVRLQX
- DTIVSCIX
- DTIVSNDX

Displacement:

00	V=VTAM services		
01–03	Three-character module ID DTIVxxxX		
04–07	TAB address for current request		
08–0B	RPL address		
0C	RPL request code		
0D	Module return code		
0E–0F	Source network address		
10–11	RPLFDBK2		
12–13	RPLUSNSI (user sense) or RPLSSNSI (system sense)		
14	LBCSTATE from VLB		
	1	LBCVSST	VTAM services state: 0=Normal 1=Termination
15	VLBERFLG — Exception response flags		
	1...	VLBPSHLD	DTIXMT pending (PS held)
	.1..	VLBERPND	Error recovery pending
	..1.	VLBERRIP	Error recovery in progress
	...1	VLBERPRG	Permanent error: purge logical unit
	.... 1...	VLBPRGDN	Purge complete
	.... .1..	VLBERPRM	Permanent error
	.... ..1.	VLBERSND	Exception Response SEND issued
	1	VLBRCVNO	Do not reissue RECEIVE SPECIFIC for logical unit
16–17		LU path ID	
18–1B	VLBSWAP		
18	VLBFLG	VLB flags	
	1...	VLBBSPBB	Bracket state-pending begin bracket
	.1..	VLBBSPEB	Bracket state-pending end bracket
	..1.	VLBBSINB	Bracket state-in bracket
	...1	VLBSR	Application SEND or RECEIVE state: 1=SEND state, 0=RECEIVE state
	 xxxx		Zero
19	VLBFLG1	VLB flag 1	
	1...	VLBIOTR	Inbound data received since last SEND indicator
	.1..	VLBCLSD	CLSDST: 0=required, 1=not required
	..1.	VLBPCS	Purging chain state
	...1	VLBSIGNL	SIGNAL sent
	.... 1...	VLBCHGD	Change Direction required
	.... .1..	VLBRETR	Retry RECEIVE request
	.... ..1.	VLBRETS	Retry SEND request
	1	VLBSNDRD	Active SEND is a Read (ER mode only)
1A	VLBFLG2	VLB flag 2	
	1...	VLBRPLB	RPL in use
	.1..	VLBCLPND	VTAM CLSDST pending
	..xx		Zero
	.... 1...	VLBBUSY	Printer busy

	.... .1..	VLBNAV	Printer not available
	.... ..1.	VLBOPCK	Operation check occurred
	1	VLBWAIT	Waiting for data or LUSTAT
1B	VLBSENDF	Special function flags	
	1...	VLBCLR	CLEAR required
	.1..	VLBSDT	SDT required
	..1.	VLBCANC	CANCEL required
	...1	VLBSIGRQ	SIGNAL required
	.... 1...	VLBCHECK	Bypass CHECK macro
	xxx		Zero
1C	VLBLRCD	LOGOFF reason codes	
	1...	VLBVTLO	VTAM LOGOFF
	.1..	VLBVRYI	VARY INACTIVE
	..1.	VLBHLTQ	HALT QUICK
	...1	VLBHLTC	HALT CANCEL
	.... 1...	VLBIOE	Unrecoverable I/O error
	.... .1..	VLBINERR	Internal error
	xx		Zero
1D–1F	VTAM Module-dependent data		

VTAM Exit Module-Dependent Data

The module flow indicator values in byte 03 are in hexadecimal.

DTIVCLSX:

01–03 WEB address or zero

DTIVDFAX:

01 WEBFUN value

02 Reserved

03 Module flow indicator

1...	DTIVPURG called
..xx xxxx	Space reserved for module flow indicator

02 SEND issued unsupported command
03 DTIWBLD issued
04 DTISCHED issued

DTIVLOGX:

01 WEBFUN value
02 SABTFLG1 field
03 Module flow indicator
- `1... ....` Asynchronous CLSDST issued
- `.1.. ....` Synchronous CLSDST issued
- `..1. ....` Purge called for recovering LU CLSDST not done
- `..xx xxxx` Space reserved for module flow indicator
  - 01 Start NIB build process
  - 03 Started building session parameters
  - 05 DTIVTRMI called
  - 06 Attempting OPNDST
  - 07 Device or BIND format is not valid
  - 08 Unsupported auxiliary device or unsupported screen size

DTIVNSEX:

01-03 Bytes 0-2 of input data field

DTIVOPNX:

01 WEBFUN value
02 Zero
03 Module flow indicator
- `1... ....` CHECK failed
- `..xx xxxx` Module flow indicator
  - 01 DTIWBLD issued
  - 02 DTISKED issued
  - 04 Cleanup request ignored
  - 08 DTIWBLD for RECEIVE SPECIFIC failed
  - 10 Cleanup called for LU

DTIVRECX:

01 WEBFUN value
02 Return code from CHECK macro
03 Module flow indicator
- `11.. ....` Reserved

..xx xxxx Space reserved for module flow indicator

- 01 Major (non-recoverable) error detected
- 02 Asynchronous device end received
- 03 Failure sending exception response (to exception request)
- 04 SEND failure trying to issue a normal response
- 05 DTIVRECM failure
- 06 DTINQ failure
- 07 DTIWBLD failure
- 08 DTIDQ failure
- 09 Chaining accumulator error
- 0A Error attempting to send chain format error response
- 0B Error attempting to send RU length error response
- 0C In purging chain state
- 0D First in chain received
- 0E Middle of chain received
- 0F End of chain received
- 10 VTAM chain indicator error
- 11 LUSTAT=No input mechanism
- 12 TWX read error
- 13 Null RU with Change Direction processed
- 14 Entry is not valid — TAB is not valid or CID mismatch
- 15 Entry ignored — logical unit undergoing CLSDST
- 16 DTIVLOFF called for logical unit
- 17 Null RU without Change Direction processed
- 18 LUSTAT=Device available
- 19 LUSTAT=Printer not available
- 1A LUSTAT=Component failure
- 1B LUSTAT=Presentation space lost
- 1C I/O error occurred
- 1D Normal SEND response received
- 1E Request ignored; RPL is busy
- 1F DTIVRECM called

DTIVRESX:

01	WEBVSRPS		
	X'15'	WEBVSCC	Command check
	X'14'	WEBVSDC	Data check
	X'13'	WEBVSCRJ	Command reject
	X'12'	WEBVSEC	Equipment check
	X'11'	WEBVSDCU	Data check and unit specify
	X'10'	WEBVSECU	Equipment check and unit specify
	X'0F'	WEBVSCC1	Reflect attention, CC1 on SIO
	X'0E'	WEBVSCNL	SEND canceled - CANCEL key
	X'0D'	WEBVSPSL	Presentation space integrity lost
	X'0C'	WEBVSVER	Unrecoverable internal error
	X'08'	WEBVSAOK	Successful completion
	X'07'	WEBTWXER	TWX read error sense X'0827'
	X'06'	WEBVSNAV	Printer not available
	X'05'	WEBVSIRQ	Intervention required
	X'04'	WEBVSBSY	Printer busy
	X'03'	WEBVSERR	Unrecoverable I/O error
	X'02'	WEBVSLFU	Unconditional LOGOFF
	X'01'	WEBVSOPC	Operation check
02	VLBERCVY		
	1...	VLBERCLR	CLEAR requested
	.1..	VLBERBRK	Bracket error occurred
	..1.	VLBERSDT	START DATA TRAFFIC required
	...1	VLBERSIG	SIGNAL required
	.... 1...	VLBERCNL	CANCEL received
	.... .1..	VLBERCDR	Change Direction required
	.... ..1.	VLBERPSL	Presentation space integrity lost
	1	VLBERHRD	Hard error occurred
03	Module flow indicator		
	11..	Reserved	

..xx xxxx Module flow indicator

- 01 Purge logical unit
- 02 SEND requested
- 03 Error recovery pending
- 04 Error recovery in progress, request ignored
- 05 Error element not found
- 06 Unrecoverable error occurred
- 07 Dequeue failure
- 08 Logical unit being purged
- 09 CLEAR canceled request
- 0A Failure not found during recovery
- 0B Recovery pending or in progress and LU is busy
- 0C Recovery pending or in progress and LU is not busy
- 0D No recovery setup and device is busy
- 14 Entry is not valid — TAB is not valid or CID mismatch
- 15 Entry ignored — logical unit undergoing CLSDST
- 16 DTIVLOFF called
- 17 Full-screen error
- 18 Cancel key 0811

DTIVRLQX:

01–03 Zero

DTIVSCIX:

01 Request code
02 UNBIND type
03 Zero

DTIVSNDX:

01 WEBVSRPS

X'15'	WEBVSCC	Command check
X'14'	WEBVSDC	Data check
X'13'	WEBVSCRJ	Command reject
X'12'	WEBVSEC	Equipment check
X'11'	WEBVSDCU	Data check and unit specify
X'10'	WEBVSECU	Equipment check and unit specify
X'0F'	WEBVSCC1	Reflect attention, CC1 on SIO
X'0E'	WEBVSCNL	SEND canceled - CANCEL key
X'0D'	WEBVSPSL	Presentation space integrity lost
X'0C'	WEBVSVER	Unrecoverable internal error
X'08'	WEBVSAOK	Successful completion
X'07'	WEBTWXER	TWX read error sense X'0827'
X'06'	WEBVSNAV	Printer not available
X'05'	WEBVSIRQ	Intervention required
X'04'	WEBVSBSY	Printer busy

	X'03'	WEBVSERR	Unrecoverable I/O error
	X'02'	WEBVSLFU	Unconditional LOGOFF
	X'01'	WEBVSOPC	Operation check
02	VLBERCVY		
	1...	VLBERCLR	CLEAR requested
	.1..	VLBERBRK	Bracket error occurred
	..1.	VLBERSDT	START DATA TRAFFIC required
	...1	VLBERSIG	SIGNAL required
	.... 1...	VLBERCNL	CANCEL received
	.... .1..	VLBERCDR	Change Direction required
	.... ..1.	VLBERPSL	Presentation space integrity lost
	1	VLBERHRD	Hard error occurred
03	Module flow indicator		
	1...	DTIVPURG called	
	.1..	DTIVLOFF called	
	..1.	Entry ignored — logical unit undergoing CLSDST	
	...1	Entry is not valid — TAB is not valid or CID mismatch	
	.... 1...	Sense 0C0D occurred during Exception Response	
	.... .1..	LUDOWN routine called	

VSCS VTAM Exit Trace: Entry 2

00	'X'	Module ID		
04	LBCSTATE	PLBFSSMF	PLBLFLG1	PLBFLG1
08	PLBPRLBC	PLBOFLG1	PLBNOM	
0C	PLBEDIT	PLBIAFLG	PLBDFLG1	RPLCNTL
10			DATA	LENGTH
14	XMT/CMT	ADDRESS		
18	DATA	BYTES 1 - 4		
1C	SEQUENCE #1	SEQUENCE #2		

Created by:

- DTIVCLSX
- DTIVDFAX
- DTIVLOGX
- DTIVNSEX
- DTIVOPNX
- DTIVRECX
- DTIVRESX
- DTIVRLQX
- DTIVSCIX
- DTIVSNDX

Displacement:

00	X=VTAM services

01–03	Three-character module ID DTIVxxxX		
04	LBCSTATE from PLB		
	1...	LBCINT	Internal mode --> 1
	.11.	LBCFSSMF	Full screen support mode flags
	.1..	LBCFSAV	Screen available
	..1.	LBCFSII	Full-screen input inhibited: B'00' CP (Console) MODE B'01' Virtual machine mode state between ATTN and READ, queue CP messages B'10' Virtual machine mode queue CP messages B'11' Virtual machine mode available for switch.
	...1	LBCCOPY	Copy mode: on=1
	.... 1...	LBCNAL	NAL flag: on=1
	.... .11.	LBCHOLD	HOLDING flag: on=B'11'
	.... .1..	LBCM10	MORE 10 flag: on=1
	.... ..1.	LBCM50	MORE 50 flag: on=1
	1	LBCCMSD	Screen written with a CMS DIAGNOSE: ON=1
	....		No MORE/HOLDING state (all bits off)
05	PLBFSSMF	Full-screen support mode flags	
	1...	PLBFPA1	PA1 key received switch mode
	.1..	PLBFSRFC	Mode switch required (FSSM to Console)
	..1.	PLBFSAFC	Switch available (FSSM to Console)
	...1	PLBFCNMP	Console message pending
	.... 1...	PLBFVSIP	VTAM services SEND in progress
	.... .1..	PLBFCLGF	Conditional LOGOFF received
	.... ..1.	PLBFCMRR	CP message pending, response required
	1	PLBCMSEW	Screen just cleared
06	PLBLFLG1	VTAM services LOGOFF flags for presentation services	
	1...	PLBLVS	VTAM services LOGOFF received
	.1..	PLBLGFC	WEBINLGF sent to for CCS conditional LOGOFF
	..1.	PLBLGFU	WEBINLGF sent to for CCS unconditional LOGOFF
	...1	PLBLERR	WEBINERR sent to CCS for hard error
	.... 1...	PLBLLERR	WEBINLER sent to CCS for interface error
	.... .1..	PLBLVERR	WEBINVER sent to CCS for internal logic error
	.... ..1.	PLBLSEVR	CP CCS SEVER received, VSCS must SEVER
	1	PLBLNSTR	No storage indicator in LOGOFF
07	PLBFLG1	flags	
	1...	PLBLGO	Logo mode logical unit
	.1..	PLBINAR	APL or TEXT starting in input area

	..1.	PLBLFLNP	LOGON from logo not permitted for this LU
	...1	PLBCONCT	Current count incremented
	.... 1...	PLBCNCTI	IUCV CONNECT issued
	.... .1..	PLBINREC	Input received during purge processing
	.... ..1.	PLBPRMPT	Usability prompt message flag
	1	PLBSDIAL	SNA dialed logical unit
08	PLBPRLBC	Previous logical unit LBCSTATE	
09	PLBOFLG1	Output manager flags	
	1...	PLBSDIAG	CMS DIAGNOSE on screen
	.1..	PLBPSMSG	Priority message on screen
	..1.	PLBOSRIP	SEND retry in progress
	...1	PLBOKLOK	Keyboard locked
	.... 1...	PLBOKPMI	Keyboard printer mask input area sent
	.... .1..	PLBOSWRT	CMS WRITE active (no input received)
	.... ..1.	PLBOCRRQ	Carriage return required before SEND (on hard-copy terminal)
	1	PLBOEXRQ	"!" required before SEND (on hard-copy terminal)
0A–0B	PLBNOM	Notify output manager	
0A	PLBNFLG1	NOM flag 1	
	1...	PLBCEND	Command end (READ)
	.1..	PLBRDTX	Redisplay timer has expired
	..1.	PLBSAC	Status area change required
	...1	PLBCLROA	CLEAR output area
	.... 1...	PLBCLRIA	CLEAR input area
	.... .1..	PLBRING	Ring the alarm
	.... ..x.		Zero
	1	PLBUNLK	Unlock keyboard
0B	PLBNFLG2	NOM flag 2	
	1...	PLBRDPND	Redisplay pending
	.xxx xxxx		Zero
0C	PLBEDIT	Editing flags	
	1...	PLBNCONV	User translated output data
	.1..	PLBPPWRT	Password prompt message
	..xx		Zero
	.... 1...	PLBHIEKO	Highlight input redisplay
	.... .1..	PLBEDPTI	Input printing or non-display
	.... ..1.	PLBNOEKO	Do not display input on next output line
	1	PLBEDCUP	Convert to upper case
0D	PLBIAFLG	Input manager attention handling flags	
	1...	PLBMATTN	Multiple attention received
	.1..	PLBSATTN	Single attention received
	..1.	PLBIPA1	Attention received in more
	...1	PLBBRKNN	CP TERMINAL BRKKEY - NONE in effect
	.... 1...	PLBUNSRD	Unsolicited read received

	.... .1..	PLBIPA1M	Multiple attentions received for a TWX LU and a Read is not Active or Response pending
	.... ..1.	PLBATTN	Signal Request-to-Send received
	1	PLBAWLOS	PRESS ENTER TO CONTINUE message is on the screen
0E	PLBDFLG1	Device manager flags	
	1...	PLBDASS	Display has alternate size
	.1..	PLBDNEW	Display has new APL or TEXT
	..1.	PLBNOVCK	Do not validity check
	...1 11..	PLBDCH	Display has new color or highlight functions No = no color or highlight functions
	...1	PLBDXTDS	Extended data stream available
	.... 1...	PLBDXCLR	Extended color available
	.... .1..	PLBDXHL1	Extended highlight available
	.... ..1.	PLBDATTN	Printer attention
	1	PLBDMASK	Password mask
0F–1F	Zero		
0F–1F	Reserved except for VDFA, VREC and VSND		
0F–11	RPLCNTL	Control information from current RPL	
12–1B	Reserved for VDFA		
12–13	DATALEN	Length of data sent or received	
14–17	WEBADDR	Work element address	
18–1B	WEBDATA	First four bytes of data sent or received	
1C–1D	Sequence number 1, only used by VRES, VSND and VREC. For VRES and VREC, sequence number 1 contains RPLSEQNO. For VSND, sequence number 1 contains RPLSEQNO if the SEND was non-LMPEO (RPLLMPEO was off). If the SEND was LMPEO, then sequence number 1 is the first sequence number in the chain of PIUS (RPLOBSQV).		
1E–1F	Sequence number 2, only used by VSND. It contains the sequence number of the last PIU in the chain of PIUs if the SEND was LMPEO (RPLLMPEO is on). For non-LMPEO SEND its contents are the same as sequence number 1.		

VTAM Exit Data Trace Entries

DTIVCLSX:

00	V	C	L	S
04	RPLREQ	Zero		
08	DTITAB address			
0C	RPL	CID		
10	RPL address			
14	DTIWEB address			
18	Zero			
1C	X'FF'	Zero		

Displacement:

00–03 Characters "VCLS"
04 RPL request code
05–07 Zero
08–0B TAB address is not valid
0C–0F RPL CID
10–13 Input RPL address
14–17 DTIWEB address
18–1B Zero
1C X'FF' to indicate data entry
1D–1F Zero

DTIVDFAX:

00	V	D	F	A
04	ACB address			
08	RPL CID			
0C	DTITAB address			
10	Zero			
14	RPL address			
18	WEB address			
1C	X'FF'	Exit	Data	

Displacement:

00–03 Characters "VDFA"
04–07 Input ACB address
08–0B Input CID
0C–0F Input TAB address
10–13 Zero
14–17 Input RPL address
18–1B DTIWEB address
1C X'FF' to indicate data trace entry
1D WEBFUN value
1E Zero
1F Module flow indicator
1... .... DTIVPURG called
..xx xxxx Module flow indicator
01 SEND issued supported command
02 SEND issued unsupported command
03 DTIWBLD issued
04 DTISCHED issued

DTIVLOGX:

00	V	L	O	G
04	ACB address			
08	LU name address			
0C	Zero			
10	Length of logon message			
14 18	LU name			
1C	X'FF'	WEBFUN	SABTFLG1	Mod. flow

Displacement:

00–03 Characters "VLOG"
04–07 Input ACB address
08–0B Address of logical unit name
0C–0F Zero
10–13 Length of logon message
14–1B Logical unit name
1C X'FF' to indicate data trace entry
1D WEBFUN value. This is explained in "WEB Function Codes" on page 535.
1E SABTFLG1 field
1F See "DTIVLOGX" for settings for 1F.

DTIVLOSX:

00	V	L	O	S
04	ACB address			
08	RPL CID			
0C	NIB user field			
10	Reason code			
14	DTITAB address			
18	DTIWEB address			
1C	X'FF'	Zero		Mod. flow

Displacement:

00–03 Characters "VLOS"
04–07 Input ACB address
08–0B Input CID
0C–0F Input NIB user field
10–13 Reason code (The reason codes are right-justified in the four-byte field.)
- 0C Contact lost
- 14 Unconditional LOGOFF
- 20 Conditional LOGOFF
- 24 Buffer limit

14–17 TAB address or zeros
18–1B WEB address or zeros
1C X'FF' to indicate data trace entry
1D–1E Zero
1F Module flow indicator
- 1... .... DTIVPURG called
- ..xx xxxx Module flow indicator
  - 1 DTIWBLD issued
  - 2 DTISKED issued

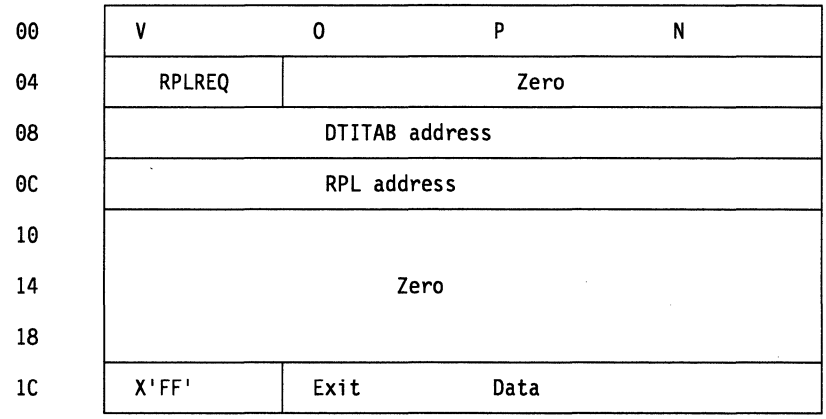
DTIVNSEX:

00	V	N	S	E
04	ACB address			
08	RPL CID			
0C	DTITAB address			
10	RPL address			
14	RPL address			
18	RPLREQ	Zero		
1C	X'FF'	Bytes 0 - 2 of user data field		

Displacement:

00–03 Characters "VNSE"
04–07 Input ACB address
08–0B Input CID
0C–0F Input TAB address
10–13 RPL address
14–17 RPL address
18 RPL request code
19–1B Zero
1C X'FF' to indicate data trace entry
1D–1F First three bytes of input data field

DTIVOPNX:



Displacement:

00–03 Characters "VOPN"
04 RPL request code
05–07 Zero
08–0B TAB address is not valid
0C–0F RPL address
10–1B Zero
1C X'FF' to indicate data trace entry
1D WEBFUN
1E Zero
1F Module flow indicator
1... .... CHECK failed
..xx xxxx Module flow indicator
01 DTIWBLD issued
02 DTISKED issued
04 Cleanup request ignored
08 DTIWBLD for RECEIVE SPECIFIC failed
10 Cleanup called for LU

DTIVTPNX:

00	V	T	P	N
04	ACB address			
08	TAB address			
0C	Number of current VTAM sessions			
10	Number of current CP connections			
14	Maximum number of CP connections up to this point			
18	SABFLGS1 field			
1C	X'FF'	Zero		

Displacement:

00–03	Characters "VTPN"		
04–07	ACB address		
08–0B	TAB address		
0C–0F	Number of current VTAM sessions		
10–13	Number of current CP connections		
14–17	Maximum number of CP connections up to this point		
18–1B	SABFLGS1 field		
18	SABFLG1	First set of DTISAB flags	
	1...	SABDBCSC	VSCS and CCS are DBCS capable
	.x..		Zero
	..1.	SABDUIP	Dump in progress
	...1	SABCSOPN	IUCVINI SET done
	.... 1...	SABBUFDQ	Dequeue type: FIFO=1 LIFO=0
	.... .1..	SABGETFL	GETMAIN failed
	.... ..1.	SABNEWLG	New logo format allowed
	1	SABVEIBS	VEIB shortage has occurred
19	SABABIPF	ABEND in progress flags	
	1...	SABPABIP	Presentation services abend in progress
	.1..	SABUABIP	Utility services abend in progress
	..xx xxxx	Zero	
1A	SABTFLG1	Global termination flags	
	1...	SABTETPE	TPEND in progress
	.1..	SABTESIQ	Termination scheduled by IQUIT
	..1.	SABTETIP	Termination in progress
	...1	SAPTEATP	Abnormal termination in progress
	 xxxx	Zero	
1B	SABRFLG1	Global recovery flags	
	1...	SABVSCTL	VTAM services in control
	.1..	SABPSCTL	Presentation services in control
	..1.	SABTICTL	Timer services exit in control
	...1	SABCXCTL	Communication services exit in control
	.... 1...	SABVXCTL	VTAM Services exit in control

.... .xxx Zero
1C X'FF' to indicate data trace entry
1D–1F Zero

DTIVRECX:

00	V	R	E	C
04	RPLREQ	Zero		
08	DTITAB address			
0C	DTIVLB address			
10	DTIWEB address			
14	RPL address			
18	Zero			
1C	X'FF'	Exit data		

Displacement:

00–03 Characters "VREC"
04 RPL request code
05–07 Zero
08–0B TAB address is not valid
0C–0F VLB address
10–13 Input WEB address
14–17 RPL address
18–1B Zero
1C X'FF' to indicate data trace entry
1D WEBFUN
1E Check return code
1F Module flow indicator
xx.. .... Zero
..xx xxxx Module flow indicator
01 Major (non-recoverable) error detected
02 Asynchronous device end received
03 Failure sending exception response (to exception request)
04 SEND failure trying to issue a normal response
05 DTIVRECM failure
06 DTINQ failure
07 DTIWBLD failure
08 DTIDQ failure
09 Chaining accumulator error
0A Error attempting to send chain format error response
0B Error attempting to send RU length error response
0C In purging chain state
0D First in chain received
0E Middle of chain received
0F End of chain received
10 VTAM chain indicator error
11 LUSTAT=No input mechanism

12 TWX read error
13 Null RU with Change Direction processed
14 Entry is not valid — TAB is not valid or CID mismatch
15 Entry ignored — logical unit undergoing CLSDST
16 DTIVLOFF called for logical unit
17 Null RU without Change Direction processed
18 LUSTAT=Device available
19 LUSTAT=Printer not available
1A Component failure
1B Presentation space lost
1C I/O error occurred
1D Normal SEND response received

DTIVRESX:

00	V	R	E	S
04	RPLREQ	Zero		
08	DTITAB address			
0C	DTIVLB address			
10	DTIWEB address			
14	RPL address			
18	Zero			
1C	X'FF'	Exit Data		

Displacement:

00–03 Characters "VRES"
04 RPL request code
05–07 Zero
08–0B TAB address is not valid
0C–0F VLB address
10–13 Input WEB address
14–17 RPL address
18–1B Zero
1C X'FF' to indicate data trace entry
1D WEBFUN
1E Check return code
1F Module flow indicator
xx.. .... Zero
..xx xxxx Module flow indicator
01 Purge logical unit
02 SEND requested
03 Error recovery pending
04 Error recovery in progress, request ignored
05 Error element not found
06 Unrecoverable error occurred
07 Dequeue failure
08 Logical unit being purged
09 CLEAR canceled request

0A Failure not found during recovery
0B Recovery pending or in progress and LU is busy
0C Recovery pending or in progress and LU is not busy
0D No recovery setup and device is busy
14 Entry is not valid — TAB is not valid or CID mismatch
15 Entry ignored — logical unit undergoing CLSDST
16 DTIVLOFF called
17 Full-screen error
18 Cancel key 0811

DTIVRLQX:

00	V	R	L	Q
04	ACB name			
08				
0C	LU name			
10				
14	DTITAB address			
18	Zero			
1C	X'FF'	Return code	Zero	

Displacement:

00–03 Characters "VRLQ"
04–0B ACB name
0C–13 Logical unit name
14–17 TAB address
18–1B Zero
1C X'FF' to indicate data trace entry
1D Return code passed to VTAM
1E–1F Zero

DTIVSNDX:

00	V	S	N	D
04	RPLREQ	Zero		
08	DTITAB address			
0C	DTIVLB address			
10	DTIWEB address			
14	RPL address			
18	Zero			
1C	X'FF'	Exit data		

Displacement:

00–03 Characters "VSND"
04 RPL request code
05–07 Zero
08–0B TAB address is not valid
0C–0F VTAM services local block (VLB) address
10–13 Input WEB address
14–17 RPL address
18–1B Zero
1C X'FF' to indicate data trace entry
1D WEBFUN
1E Check return code
1F Module flow indicator

xx.. .... Zero
..xx xxxx Module flow indicator

- 01 Major (non-recoverable) error detected
- 02 Asynchronous device end received
- 03 Failure sending exception response (to exception request)
- 04 SEND failure trying to issue a normal response
- 05 DTIVRECM failure
- 06 DTINQ failure
- 07 DTIWBLD failure
- 08 DTIDQ failure
- 09 Chaining accumulator error
- 0A Error attempting to send chain format error response
- 0B Error attempting to send RU length error response
- 0C In purging chain state
- 0D First in chain received
- 0E Middle in chain received
- 0F Last in chain received
- 10 VTAM chain indicator error
- 11 LUSTAT = no input mechanism
- 12 TWX read error
- 13 Null RU with CD
- 14 Entry is not valid — TAB is not valid or CID mismatch
- 15 Entry ignored — logical unit undergoing CLSDST
- 16 DTIVLOFF called

DTIVSCIX:

00	V	S	C	I
04	ACB address			
08	CID (not used)			
0C	DTITAB address			
10	RPL address			
14	RPL address			
18	RPLREQ	Zero		
1C	X'FF'	Exit data		

Displacement:

00–03 Characters "VSCI"
04–07 ACB address
08–0B CID (not used)
0C–0F TAB address
10–13 RPL address
14–17 RPL address
18 RPL request
19–1B Zero
1C X'FF' to indicate data trace entry
1D Request code
1E UNBIND type
1F Zero

DTIVSEND: Entry 1

00	V	S	E	N
04	DTITAB address			
08	DTIWEB address			
0C	XMTFLG1	XMTFLG2	MODE/TYPE	XMTSTATE
10	LBCSTATE	PLBFSSMF	PLBLFLG1	PLBFLG1
14	PLBPRLBC	PLBNFLG1	PLBOFLG1	PLBFSSM1
18	Caller's return address			
1C	X'FF'	Zero		

Displacement:

00–03 Characters "VSEN"
04–07 DTITAB address
08–0B DTIWEB address (DTICMT, DTIWEB, DTIXMT)
0C XMTFLG1

1...	XMTCD	Change Direction required (for start/stop)
.1..	XMTRSHOW	Redisplay in DTIXMT buffer

	..1.	XMTPMSG	Priority message on screen
	...1	XMTDIAG	CMS DIAGNOSE on screen
	.... 1...	XMTLRU	Logo RU
	.... .1..	XMTATTN	Attention write (VSCS only) yes = 1
	.... ..1.	XMTINPT	Input expected from logical unit
	1	XMTPWMSK	Password mask
0D	XMTFLG2		
	1...	XMTVSRBL	DTIVDATR called
	.1..	XMTERCMT	DTIXMT is really a DTICMT
	..1.	XMTTERERR	Error retry necessary
	...1	XMTERPSL	Presentation space lost error
	.... 1...	XMTLMPEO	LMPEO SEND initiated
	.... .xx.		Zero
	1	XMTKEYRS	Contains data that will reset the keyboard (LU0 and LU2)
0E	MODE or TYPE of request		
	B Busy		
	C Control command		
	D Definite response		
	E Exception response		
	I Send request ignored.		
	P Logical unit purged		
	R Send request delayed.		
	X Recovery request		
	W Logical unit waiting		
0F	XMTSTATE — LBCSTATE of PLB saved in XMT		
	1...	LBCINT	Internal mode if on (on=1)
	.11.	LBCFSSMF	Full-screen support mode flags
	.1..	LBCFSAV	Screen available
	..1.	LBCFSII	Full-screen input inhibited:
			B'00' CP (Console) mode
			B'01' Virtual machine mode state between ATTN and READ, queue CP messages
			B'10' Virtual machine mode queue CP messages
			B'11' Virtual machine mode available for switch.
	...1	LBCCOPY	Copy mode: on=1
	.... 1...	LBCNAL	NAL flag: on=1
	.... .11.	LBCHOLD	HOLDING flag: on=B'11'
	.... .1..	LBCM10	MORE 10 flag: on=1
	.... ..1.	LBCM50	MORE 50 flag: on=1
	1	LBCCMSD	Screen written with a CMS DIAGNOSE: on=1
	....		No MORE/HOLDING state
10	LBCSTATE from PLB		
	1...	LBCINT	Internal mode if on (on=1).
	.11.	LBCFSSMF	Full-screen support mode flags

	.1..	LBCFSAV	Screen available
	..1.	LBCFSII	Full-screen input inhibited: B'00' CP (Console) mode B'01' Virtual machine mode state between ATTN and READ, queue CP messages B'10' Virtual machine mode queue CP messages B'11' Virtual machine mode available for switch.
	...1	LBCCOPY	Copy mode: on=1
	.... 1...	LBCNAL	NAL flag: on=1
	.... .11.	LBCHOLD	HOLDING flag: on=B'11'
	.... .1..	LBCM10	MORE 10 flag: on=1
	.... ..1.	LBCM50	MORE 50 flag: on=1
	1	LBCCMSD	Screen written with a CMS DIAGNOSE: on=1
	....		No MORE/HOLDING state
11	PLBFSSMF — Full-screen support mode flags		
	1...	PLBFPA1	PA1 key received switch mode
	.1..	PLBFSRFC	Mode switch required (FSSM to Console)
	..1.	PLBFSAFC	Switch available (FSSM to Console)
	...1	PLBFCNMP	Console message pending
	.... 1...	PLBFVSIP	VTAM services SEND in progress
	.... .1..	PLBFCLGF	Conditional LOGOFF received
	.... ..1.	PLBFCMRR	CP message pending, response required
	1	PLBCMSEW	Screen has just been erased
12	PLBLFLG1 – VTAM services LOGOFF flags for presentation services		
	1...	PLBLVS	VTAM services LOGOFF received
	.1..	PLBLGFC	WEBINLGF sent to CCS for conditional LOGOFF
	..1.	PLBLGFU	WEBINLGF sent to CCS for unconditional LOGOFF
	...1	PLBLERR	WEBINERR sent to CCS for hard error
	.... 1...	PLBLLERR	WEBINLER sent to CCS for interface error
	.... .1..	PLBLVERR	WEBINVER sent to CCS for internal logic error
	.... ..1.	PLBLSEVR	CP CCS SEVER received, VSCS must SEVER
	1	PLBLNSTR	No storage indicator in LOGOFF
13	PLBFLG1	PLB flags	
	1...	PLBLGO	Logo mode logical unit
	.1..	PLBINAR	APL or TEXT starting in input area
	..1.	PLBLFLNP	LOGON from logo not permitted for this LU
	...1	PLBCONCT	Current count incremented
	.... 1...	PLBCNCTI	IUCV CONNECT issued
	.... .1..	PLBINREC	Input received during purge processing

	.... ..1.	PLBPRMPT	Usability prompt message flag
	1	PLBSDIAL	SNA dialed logical unit
14	PLBPRLBC	Previous LBCSTATE from PLB	
15	PLBNFLG1 — NOM flag 1		
	1...	PLBCEND	Command end (READ)
	.1..	PLBRDTX	Redisplay timer has expired
	..1.	PLBSAC	Status area change required
	...1	PLBCLROA	CLEAR output area
	.... 1...	PLBCLRIA	CLEAR input area
	.... .1..	PLBRING	Ring the alarm
	.... ..x.		Zero
	1	PLBUNLK	Unlock keyboard
16	PLBOFLG1 — Output manager flags		
	1...	PLBSDIAG	CMS DIAGNOSE on screen
	.1..	PLBPSMSG	Priority message on screen
	..1.	PLBOSRIP	SEND retry in progress
	...1	PLBOKLOK	Keyboard locked
	.... 1...	PLBOKPMI	Keyboard printer mask input area sent
	.... .1..	PLBOSWRT	CMS WRITE active (no input received)
	.... ..1.	PLBOCRRQ	Carriage return required before SEND on hard-copy terminal
	1	PLBOEXRQ	"!" required before SEND on hard-copy terminal
17	PLBFSSM1		
	1...	PLBFSCLR	CMS CLEAR while in FSSM state
	.1..	PLBFSRBI	Full-screen read buffer issued
	..1.	PLBPASPA	Pass PA1 as data (FSSM)
	...1	PLBWSFRO	WSF query issued
	.... 1...	PLBCONMD	CONMODE 3270 request
	.... .1..	PLBFSRBS	Full-screen read buffer SEND in progress
	.... ..1.	PLBBEEP	Full-screen break-in beep in progress
	1	PLBFSPND	Full-screen request could not be processed and is on the pending queue
18–1B	Caller's return address		
1C	X'FF' to indicate data trace entry		
1D–1F		Zero	

DTIVSEND: Entry 2

00	V	S	E	1
04	VTAM CID			
08	VLBSWAP			
0C	VLBLRCD	VLBERFLG	VLBFLG3	VLBERCVY
10	COMMAND CODE	Zero	DATA	COUNT
14	WEB/CMT		ADDRESS	
18	BYTES 1 - 4	OF	DATA	
1C	X'FF'	Zero		

00–03 Characters 'VSE1'
04–07 VTAM CID (OAF/DAF)
08–0B VLBSWAP — VTAM services flags

08	VLBFLG — VLB flag		
	1...	VLBBSPBB	Bracket-state pending begin bracket
	.1..	VLBBSPEB	Bracket-state pending end bracket
	..1.	VLBBSINB	Bracket-state in bracket
	...1	VLBSR	Application SEND or RECEIVE state: 1=SEND state, 0=RECEIVE state
	 xxxx		Zero
09	VLBFLG1 — VLB flag 1		
	1...	VLBIOTR	Inbound data received since last SEND
	.1..	VLBCLSD	CLSDST not required
	..1.	VLBPCS	Purging chain state
	...1	VLBSIGNL	SIGNAL sent
	.... 1...	VLBCHGD	Change Direction required
	.... .1..	VLBRETR	Retry RECEIVE request
	.... ..1.	VLBRETS	Retry SEND request
	1	VLBSNDRD	Active SEND is a Read (ER mode only)
0A	VLBFLG2 — VLB flag 2		
	1...	VLBRPLB	RPL in use
	.1..	VLBCLPND	VTAM CLSDST pending
	..xx		Zero
	.... 1...	VLBBUSY	Printer busy
	.... .1..	VLBNAV	Printer not available
	.... ..1.	VLBOPCK	Operation check occurred
	1	VLBWAIT	Waiting for data or LUSTAT
0B	VLBSENDF — Special function flags		
	1...	VLBCLR	CLEAR required
	.1..	VLBSDT	START DATA TRAFFIC (SDT) required
	..1.	VLBCANC	CANCEL required
	...1	VLBSIGRQ	SIGNAL required
	.... 1...	VLBCHECK	Bypass CHECK macro
	xxx		Zero

0C	VLBLRCD — LOGOFF reason codes		
	1...	VLBVTLO	VTAM LOGOFF
	.1..	VLBVRYI	VARY INACTIVE
	..1.	VLBHLTQ	HALT QUICK
	...1	VLBHLTC	HALT CANCEL
	.... 1...	VLBIOE	Unrecoverable I/O error
	.... .1..	VLBINERR	Internal error
	xxx		Zero
0D	VLBERFLG — Exception response flags		
	1...	VLBPSHLD	DTIXMT pending (PS held)
	.1..	VLBERPND	Error recovery pending
	..1.	VLBERRIP	Error recovery in progress
	...1	VLBERPRG	Permanent error: purge logical unit
	.... 1...	VLBPRGDN	Purge complete
	.... .1..	VLBERPRM	Permanent error
	.... ..1.	VLBERSND	Exception Response SEND issued
	1	VLBRCVNO	Do not reissue RECEIVE SPECIFIC for logical unit
0E	VLBFLG3		
	1...	VLBNULLS	Null SEND executed
	.1..	VLBSCIPR	SCIP exit retry in progress
	..1.	VLBCLRCV	Last input received was a CLEAR key
	...1	VLBSCSIM	SCIP exit retry in progress SIMLOGON required
	.... 1...	VLBINOFC	Operator FORCE command issued for this LU
	.... .1..	VLBERRCV	Reissue RECEIVE SPECIFIC in response mode
	.... ..1.	VLBERCLQ	Clean queue when SEND completed
	1	VLBERSCH	Schedule response when SEND completes
0F	VLBERCVY – Recovery flags		
	1...	VLBERCLR	CLEAR needed
	.1..	VLBERBRK	Bracket error
	..1.	VLBERSDT	SDT needed
	...1	VLBERSIG	SIGNAL needed
	.... 1...	VLBERCNL	CANCEL received
	.... .1..	VLBERCDR	Change Direction needed
	.... ..1.	VLBERPSL	Presentation space lost
	1	VLBERHRD	Hard error occurred
10	Command code when TRDMODE is a C = control command sent		
	C = CLEAR	N = None	
	S = SDT	X = CANCEL	
	Z = SIGNAL		
11	Zero		
12–13	Data count sent		
14–17	WEB/CMT address		
18–1B	First 4 bytes of data sent		

1C X'FF' to indicate data trace entry
1D–1F Zero

DTIVSIML

00	V	S	I	M
04	RPL address			
08	RPLFDBWD			
0C	RPLFDBK2			
10 14	LU name			
18	DTITAB address			
1C	X'FF'	Exit data		

Displacement:

00–03 Characters "VSIM"
04–07 Input RPL address
08–0B RPLFDBWD
0C–0F RPLFDBK2
10–17 Logical unit name
18–1B DTITAB address
1C X'FF' to indicate data trace entry
1D–1F Return code from CHECK macro

LBCSTATE Values for PLBINxxx

Console Mode

Name	Hex Value	Description
PLBINNMH	00	No MORE or HOLDING condition
PLBINM50	02–03	MORE condition, 50 second timer set
PLBINM10	04–05	MORE condition, 10 second timer set
PLBINHLD	06–07	HOLDING
PLBINNAL	08–0F	Not accepted logic condition
PLBINCPY	10–1F	PF key Copy in progress

CMS Edit Mode

Name	Hex Value	Description
PLBINCMS	01	No MORE or HOLDING condition
PLBINM50	02–03	MORE condition, 50 second timer set
PLBINM10	04–05	MORE condition, 10 second timer set
PLBINHLD	06–07	HOLDING
PLBINNAL	08–0F	Not accepted logic condition

Full-Screen Support Mode (FSSM)

Name	Hex Value	Description
PLBINFSS	20, 40, 60	Full-screen support mode

Internal Modes

Name	Hex Value	Description
PLBINLIP	80	LOGON in progress
PLBINLUP	81	Logo up
PLBINCIP	82	Connect in progress
PLBINIIP	83	Inlog in progress
PLBINAWL	84	Attention wait before logo
PLBINLGF	85	LOGOFF
PLBINCLF	86	Conditional LOGOFF
PLBINSVR	87	SEVER required
PLBINPRG	88	Purge in progress

WEB Function Codes

The tables in this section show the bits defined in field WEBFUN, the function code for VSCS. The function code depends on the value in WEBMODE:

- If WEBMODE is 04, use "Function Codes for Internal States" on page 536.

Internal function codes represent work functions within VSCS. These are not considered to be inbound or outbound.

- For all other WEBMODEs:
  - For inbound, use "Function Codes for Inbound Processing in All Modes" on page 537. *Inbound* function codes represent work elements that are received by CCS.
  - For outbound, use the WEBMODE:
    - If the WEBMODE is 00, use "Function Codes for Outbound Processing in Console Mode" on page 537.
    - If the WEBMODE is 01, use "Function Codes for Outbound Processing in CMS Mode" on page 538.
    - If the WEBMODE is 02, use "Function Codes for Outbound Processing in Full-Screen Mode" on page 538.

  Outbound function codes represent work functions sent from CCS to VSCS, or generated internally in VSCS. The mode of the request (CMS, console, or FSSM) determines which function is performed.

Note: In this book, whenever a flow is referred to as *inbound* or *outbound*, use console communication services (CCS) as the reference point. That is, flows *toward* CCS are *inbound*; flows *away from* the CCS are *outbound*.

Function Codes for Internal States

Name	Hex Value	Description
WEBNTM10	00	Timer expired - MORE 10 second, source = System Services
WEBNVLFC	01	LOGOFF conditional, source = VTAM services
WEBNVLFU	02	LOGOFF unconditional, source = VTAM services
WEBINERR	03	Unrecoverable I/O error, LOGOFF condition
WEBINPUT	04	Attention identifier and optional data input
WEBNVFLW	05	LOGOFF warning, source = VTAM services
WEBNSRSP	06	Input/output response (could be either normal or error), source = VTAM services
WEBNVLGO	07	LOGON response, source = VTAM services
WEBNACON	08	Accept connect, source = CCS
WEBNSEVR	09	Sever connection, source = CCS
WEBNMSGC	0A	IUCV one-way message complete, source = CCS
WEBNTM50	0B	Timer expired - MORE 50 second, source = System Services
WEBNTNAL	0D	Timer expired - not accepted 3 second
WEBNTF60	0E	Timer expired = FSSM 60 second, source = System Services
WEBVSBRK	0F	LOGOFF request, source = presentation services
WEBNVASY	10	Asynchronous error - WEBVSRPS indicates type
WEBPSSRU	10	SEND request, source = presentation services
WEBNVCLU	11	Internal VTAM request cleanup, source = VTAM services
WEBNSIML	12	Request to VTAM services to issue SIMLOGON

WEBNTCPY	13	PF key Copy wait for printer SIMLOGON timer expired
WEBNTPRT	14	Timer expired for request to release printer after RELREQ exit driven
WEBINOFC	14	Operator issued VSCS FORCE command — processing continues in DTIPFORC

Function Codes for Inbound Processing in All Modes

Name	Hex Value	Description
WEBINLOG	01	LOGON request
WEBINATT	02	Attention interrupt without data
WEBINERR	03	Unrecoverable I/O error, LOGOFF condition
WEBINPUT	04	Attention identifier with optional data
WEBINCB1	05	Move cursor back one
WEBINLGF	06	VTAM-generated LOGOFF condition, CCS logs user off
WEBINFNU	07	FSSM request rejected, screen not in user mode
WEBINAOK	08	Normal response
WEBINACT	09	Accounting data for CCS
WEBINOPC	0A	Operation check for FSSM request
WEBINLER	0B	Logical interface error (DTIWEB)
WEBINVER	0C	VSCS internal processing error
WEBRDERR	0D	Error in input data found by the NCP
WEBINVTM	0E	Identify VTAM user ID to CCS

Function Codes for Outbound Processing in Console Mode

Name	Hex Value	Description
WEBCPWRT	00	WRITE to next available line in output area
WEBCPRED	01	READ request
WEBCPBRK	02	Break connection (CP LOGOFF or DISCONNECT)
WEBCPWIC	03	WRITE to input area and then position cursor as indicated in WEBCURSR
WEBCPCPY	04	COPY request
WEBCPIED	05	Environment definition is not valid
WEBCPFKR	06	PFK REPLY - treat as terminal input
WEBCPIDA	07	Data accepted, proceed with input processing
WEBCPINA	08	Input not accepted
WEBCPREX	09	IREPLY expected by way of synchronous
WEBCPTAB	0A	PF key TAB operation
WEBCPLER	0B	Logical interface error (WEB) (ISEND or REPLY)
WEBCPLOG	0C	Logo
WEBCPTMR	0D	Command end (no output for command entered)
WEBCPPTH	0E	IUCV path ID
WEBCPCAH	0F	Color and highlight map

WEBCPLSA	10	Device information (terminal command)

Function Codes for Outbound Processing in CMS Mode

Name	Hex Value	Description
WEBCMWRT	00	WRITE to line number specified in WEBLINE
WEBCMEWT	01	ERASE output area and WRITE to line number specified in WEBLINE
WEBCMCLR	02	ERASE entire screen and refresh the format of the output, input, and status areas

Function Codes for Outbound Processing in Full-Screen Mode

Name	Hex Value	Description
WEBFSWRT	00	FSSM WRITE
WEBFSEWT	01	FSSM ERASE/WRITE
WEBFSEWA	02	FSSM ERASE/WRITE ALTERNATE
WEBFSRDM	03	FSSM READ MODIFIED
WEBFSRDB	04	FSSM READ BUFFER
WEBFSWSF	05	FSSM WRITE STRUCTURED FIELD
WEBFSEAU	06	FSSM ERASE ALL UNPROTECTED
WEBFSRBP	07	FSSM READ BUFFER by position
WEBFSRMP	08	FSSM READ MODIFIED by position
WEBINCC1	0F	FSSM reflect attention, CC1 on SIO
WEBINECU	10	FSSM equipment check and unit specify
WEBINDCU	11	FSSM data check and unit specify
WEBINEC	12	FSSM equipment check
WEBINCRJ	13	FSSM command reject
WEBINDC	14	FSSM data check
WEBINCC	15	FSSM control check

Chapter 12. Using Other Problem Solving Tools

Many different service aids are available to help you collect information about SNA network problems. This chapter describes when to use the following aids:

Alert Messages from NCP

The NCP in a 3720, 3725, or 3745 communication controller sends alert messages to the VTAM program whenever a serious or permanent error occurs in the communication controller.

The NCP sends hardware error records to the maintenance operator subsystem (MOSS). If the MOSS determines that they are permanent errors, it sends them back to the NCP, which forwards them to all owning host processors.

If a communication network management (CNM) application program, such as the NetView program, is active and authorized to receive alert messages, VTAM forwards the alert messages to that program. Otherwise, VTAM sends a message to the operator's console.

Recording NMVT Alerts in LOGREC

A network management vector transport (NMVT) is a SNA request unit (RU) that contains solicited or unsolicited data, such as line statistics and generic alerts. MVS VM LOGREC and VSE SYSREC are host data sets that contain records of various types of system failures, both hardware and software.

VTAM records all *unsolicited* NMVT alerts from 3745s and *all* NMVT alerts from local area networks and ES/9370* processors in either LOGREC or SYSREC. If you have the NetView program, VTAM also forwards the NMVT alerts to the NetView hardware monitor for recording. The NetView program interprets the error information for its operator panels. To determine what book contains more information on NetView's presentation of generic alerts, see Table 80 on page 1165. For more information on generic alerts generated by First Failure Support Technology (FFST), see "First Failure Support Technology (FFST) for VTAM (MVS) (VM)" on page 217.

VTAM identifies and records an NMVT alert as follows:

- Each NMVT has an SNA network services (NS) header of X'41038D'.

- Each NMVT that is an alert has a management services (MS) major vector of X'0000'.
- To determine the type of device that sent the NMVT alert, VTAM checks the product ID subvector (X'11') of the NMVT for the hardware machine type. Each type of device has its own unique machine type; for example, each NMVT alert that comes from a 3745 has a machine type of C'3745'.
- VTAM records the NMVT alert in LOGREC or SYSREC as a miscellaneous data record (MDR: type=X'91') with a device type of NMVT (X'30').

  You can format and print NMVT alerts from LOGREC or SYSREC using the Environmental Record Editing and Printing program (EREP).
- At offset X'1A' of the MDR record is the 8-byte resource name of the sender of the MNVT alert.
- At offset X'24' of the MDR record is the NMVT data. The first five bytes of the data are not included. See Table 80 on page 1165 to determine what book contains information on how to use EREP.

Alerts (VM) (VSE)

VTAM uses the alert function to notify an operator of an actual or impending loss of availability of a resource. VTAM builds a network management vector transport (NMVT) for the alert. The NMVT can be traced with the PIU option of the VIT. The PIU trace of the NMVT contains several extensions. Figure 78 is an example PIU trace entry of an alert NMVT.

```
D7C9E440  43990000  00863DD8  40000000     *PIU .......Q ...*
20000000  00000001  00000001  1C000001     *................*
D7C9E4F2  00000001  00860B00  0041038D     *PIU2............*
00001000  10007B00  000A0401  80000000     *......#.........*
D7C9E4F2  01000C0A  01081057  061A0C0F     *PIU2............*
153C1000  2311040E  02F5F6F6  F4F2F8F0     *.........5664280*
D7C9E4F2  F0F1F3F1  F10804F0  F3F0F1F0     *PIU201311..03010*
F10A06C1  C3C661E5  E3C1D416  11011300     *1..ACF/VTAM.....*
D7C9E4F2  11F9F3F7  F5000000  0000F0F0     *PIU2.9375.....00*
F0F0F0F5  F60E5104  02000508  03400000     *00056........ . *
D7C9E4F2  00003105  52030404  0D910001     *PIU2............*
0F008000  C100C100  0004A108  4003A105     *....A.A.........
```

Figure 78. PIU Trace Entry Example (NMVT RU) VM VSE

The NMVT RU that is traced contains the alert (X'0000') management services major vector. See Table 80 on page 1165 to determine what books describe the alert major vector and its subvectors.

In the preceding example, the alert major vector has three subvectors that identify alert-description-code information. The first subvector, hex 91, is a basic alert subvector that has an alert description code of C1. Two hex A1 values identify the detail-qualifier subvectors for alert C1. (Depending on the type of alert, up to three detail-qualifier subvectors follow the basic alert subvector; see "LAN Alert Description Codes (VM) (VSE)" on page 541 for details.) The detail-qualifier subvectors have values of 0840 and 05, respectively.

LAN Alert Description Codes (VM) (VSE)

The list that follows describes each alert by description code and provides the list of detail qualifier items that accompany the alert. See "Description of Detail Qualifier Items (VM) (VSE)" on page 547 for descriptions of the detail qualifier items.

Code Type of Failure

66 Open Failure: Token-Ring Lobe

Description: The adapter detected a problem on its lobe during the wrap-test portion of the insertion process. The insertion process did not complete.

Detail data:

1. Channel unit address
2. Command
3. Error code.

67 Open Failure: Token-Ring Fault Domain

Description: The adapter detected a beaconing condition on the ring during the insertion process. The insertion process did not complete.

Detail data:

1. Channel unit address
2. Command
3. Error code.

68 Open Failure: Token-Ring Duplicate Station Address

Description: The adapter detected the presence of a station with its address on the ring during the insertion process. The insertion process did not complete.

Detail data:

1. Channel unit address
2. Command
3. Error code.

69 Open Failure: Token-Ring Remove Command Received

Description: The adapter received a Remove Ring Station MAC frame during the insertion process. The insertion process did not complete.

Detail data:

1. Channel unit address
2. Command
3. Error code.

6A Open Failure: Token-Ring Lobe

Description: An error was detected during the insertion process that is not defined in the previous alerts. These conditions are not expected to occur, so they are included within one alert definition. The insertion process did not complete.

Detail data:

1. Channel unit address
2. Command
3. Error code.

6B Adapter Error: Local Token-Ring Adapter

Description: A hardware error in the reporting station's adapter has occurred. The adapter is now inoperable.

Detail data:

1. Channel unit address
2. Command
3. Error code.

6C Wire Fault: Token-Ring Lobe

Description: The reporting station's adapter detected a wire-fault condition on the ring.

Detail data:

1. Channel unit address
2. Command
3. Error code.

6D Auto Removal: Token-Ring Lobe

Description: The reporting station's adapter has left the ring as part of the beacon automatic-recovery process. That is, the reporting station's adapter was a member of the beacon fault domain. The station removed itself from the token ring and ran a self test, which was unsuccessful.

Detail data:

1. Channel unit address
2. Command
3. Error code.

6E Remove Command Received: Network Operator

Description: The reporting station's adapter received a Remove Adapter command from a LAN manager and, as a result, left the ring.

Detail data:

1. Channel unit address
2. Command
3. Error code.

70 Token-Ring Inoperative: Token-Ring Fault Domain

Description: The ring has been beaconing for more than 52 seconds.

Detail data:

1. Channel unit address
2. Command
3. Error code.

72 Token-Ring Temporary Error: Token-Ring Fault Domain

Description: The ring was in a beaconing state for less than 52 seconds and then recovered. The sender of this alert does not know whether a station removed itself from the token-ring to bypass the fault, or whether the fault was temporary.

Detail data:

1. Channel unit address
2. Command
3. Error code.

83 Link Error: LAN LLC Communications/Remote Node

Description: The remote link station does not respond. The response timer (t1) has expired, causing the remote station to be polled. The retry count (n2) for the number of polls has been exceeded.

Detail data:

1. Channel unit address
2. Command
3. Error code.

84 Link Error: LAN LLC Communications

Description: The remote link station sent a Disconnect Mode (DM) to the local link station.

Detail data:

1. Channel unit address
2. Command
3. Error code.

87 Software Program Error: LAN LLC Communications

Description: The remote link station sent an SABME (set asynchronous balanced mode-extended) to the local link that had been previously initialized through an SABME-UA (unnumbered acknowledgement) exchange.

Detail data:

1. Channel unit address
2. Command
3. Error code.

A0 Software Program Error: LAN LLC Communications

Description: The local link station sent an unsupported frame or a frame that is not valid to the remote link station, which resulted in the remote link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

A1 Software Program Error: LAN LLC Communications

Description: The local link station sent an I-field when not permitted. That resulted in the remote link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

A2 Software Program Error: LAN LLC Communications

Description: The local link station sent a send sequence number that is not valid, which resulted in the remote link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

A3 Software Program Error: LAN LLC Communications

Description: The local link station sent an I-field that was too long, which resulted in the remote link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

AA Software Program Error: LAN LLC Communications

Description: The remote link station sent an unsupported frame or a frame that is not valid to the local link station, which resulted in the local link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

AB Software Program Error: LAN LLC Communications

Description: The remote link station sent an I-field when not permitted, which resulted in the local link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

AC Software Program Error: LAN LLC Communications

Description: The remote link station sent a receive sequence number that is not valid, which resulted in the local link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

AD Software Program Error: LAN LLC Communications

Description: The remote link station sent an I-field that was too long, which resulted in the local link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command

3. Error code.

BE Configuration or Customization Error: Local System Operator

Description: The local system operator made an error in the VTAM definition. One of the following situations occurred:

- A physical unit dialed in, and there were no more available lines defined in the LAN major node.
- The activation of a line in a LAN major node caused the total number of active lines to exceed the maximum stations coded for the LAN major node.
- A dial out was attempted that will cause VTAM to exceed the maximum stations defined for the adapter. If more than one user is connected to the adapter, then the maximum number of stations must include the total number of stations for all connections.

Detail data:

1. Channel unit address
2. Reason code:
   04 Dial in line not available
   08 Maximum number of stations exceeded (line activation)
   09 Maximum number of stations exceeded (dial out).

BF Communications Subsystem Failure: Communications Subsystem Controller Interface

Description: One of the channel programs for the CETI group failed.

Detail data:

1. Channel unit address
2. Channel status
3. Sense.

C0 Communications Subsystem Failure: Communications Subsystem Controller Interface

Description: A command to the adapter has reported an unsuccessful completion or an asynchronous indication has been received from the adapter.

Detail data:

1. Channel unit address
2. Command
3. Error code.

C1 Communications Subsystem Failure: Communications Subsystem Controller Interface

Description: Local area network support code detected an error. The error is not the result of a user definition.

Detail data:

1. Channel unit address
2. Reason code:
   01 MAXDATA segmenting error
   02 Sequence error
   03 PIU is not valid

05 Adapter input is discarded
06 CETI architecture violation
07 Unexpected SABME or XID received.

C2 Configuration or Customization Error: Local System Operator

Description: The local system operator has made an error in the VTAM definition. A local or remote node address that is not valid was specified. Either the SAP address or the MAC address is not valid or not recognized, or the station is not on the ring.

Detail data:

1. Channel unit address
2. Command
3. Error code.

C3 Configuration or Customization Error: Local System Operator

Description: The local system operator has made an error in the VTAM definition. The maximum data size is too large or is not valid.

Detail data:

1. Channel unit address
2. Command
3. Error code.

C4 Configuration or Customization Error: Local System Operator

Description: The local system operator has made an error in the VTAM definition. The service access point (SAP) address on the PORT statement is already in use.

Detail data:

1. Channel unit address
2. Command
3. Error code.

C5 Configuration or Customization Error: Local System Operator

Description: The local system operator has made an error in the VTAM definition. The number of stations specified is not valid on the PORT statement or the number of lines defined in the LAN major node is too large for the adapter.

Detail data:

1. Channel unit address
2. Command
3. Error code.

C6 Warning: Single Station on the Ring

Description: The adapter has opened and it is the only station on the ring.

Detail data:

1. Channel unit address
2. Command
3. Error code.

C7 Configuration or Customization Error: Local System Operator

Description: The local system operator has made an error in the VTAM definition. The station that is being activated as a LAN resource is not defined to VTAM as LAN-capable.

Detail data:

1. Channel unit address.

Description of Detail Qualifier Items (VM) (VSE)

Channel unit address This is the channel unit address specified on the PORT statement that defines a CETI group.

Command This is the command that was processed when the alert was reported. Commands are used by VTAM LAN support to communicate with the adapter. See "Command Values Reported in Trace Entries (VM) (VSE)" for more about commands used by LAN support.

Error code This is the completion status code that is associated with the command.

Channel status This is the channel status that is reported for a failing CETI channel program.

Sense This is the sense that is reported for a failing CETI channel program. This value may be 0 if no sense data is available.

Reason code This is the reason code for the failure reported by the alerts with description codes of BE and C1. See the listing of those description codes for explanations of the values reported.

Command Values Reported in Trace Entries (VM) (VSE)

Table 33 describes the command values reported in trace entries. VTAM LAN support and token-ring LANs attached through the ES/9370 controller use these commands to communicate.

Note: Token-ring LANs attached through the IBM 3172 Interconnect Controller do not use these commands.

Table 33 (Page 1 of 2). Command Value Descriptions

Command Value	Description
810000	Set network parameters
040C0D 04CC0D	Activate VTAM's service access point
040C0E 04CC0E	Deactivate VTAM's service access point
040C0A 04CC0A	Cause the subsystem to send test frames for remote station address resolution and to establish a logical connection with a remote station
040C0B 04CC0B	Cause the subsystem to break a logical connection with a remote station

Table 33 (Page 2 of 2). Command Value Descriptions

Command Value	Description
040C20 04CC20 044C20 048C20	Perform SNA exchange identification (XID)
040D00 04CD00 044D00 048D00	Perform SABME - UA exchange
040D01 04CD01 044D01	Perform DISC - UA/DM exchange
040D10 04CD10 044D10	Send and receive normal data
040D50 04CD50 044D50	Control the flow of data to and from a remote station
040660 04C660	Send and receive test frames to perform VTAM link level 2 (LL2) test
044D1A 024D1A	Pass asynchronous status and statistics

The commands in the table are in the form:

xxyxxx

where

xx_xxx is the command identifier and
y is the command type.

The command types for *y* and the direction of flow are:

y	Command Type	Direction of Flow
0	Request	From VTAM LAN support to the subsystem
4	Indication	From the subsystem to VTAM LAN support
8	Response to an indication	From VTAM LAN support to the subsystem
C	Confirmation of a request	From the subsystem to VTAM LAN support

Note: The set network parameters command flows as both a request and a confirmation. It is always shown as 810000 in the VTAM trace command fields.

Messages Issued for 3745 Bus Switching

A 3745 can be configured with one or two CCUs. For twin-CCU 3745s, NCP V5 supports the two CCUs as independent communication controllers.

A twin-CCU 3745 can be operated such that, if one of the CCUs fails, the maintenance and operator subsystem (MOSS) switches the I/O controller (IOC) buses from the failing CCU to the active CCU. The operator can then activate all or part of the resources of the failing CCU to the other CCU.

The bus-switch can also be performed at the operator's request.

If the bus-switch occurs in the middle of a VTAM channel operation, such as a Read or a Write, VTAM issues messages IST881I and IST882I.

Message IST881I: Message IST881I tells the operator that VTAM either was unable to contact a link station, or lost contact to a link station.

This message is issued for one of the following reasons:

- A bus is being switched from one CCU to the other.
- A dump or load is being done on another channel.
- A dump or load is being done by a local disk.

When the link station becomes available (as indicated in other messages), VTAM resumes CONTACT processing.

To terminate CONTACT processing before the link station becomes available, issue a VARY INACT,FORCE command to deactivate the link station.

Message IST882I: Message IST882I tells the operator that VTAM is waiting for a device end from the link station identified in message IST881I.

You should check to see if the link station is online. If it is, then there is a possibility that NCP is being dumped or loaded over another channel adapter, and no further action is necessary.

Note: If the link station is not operating, not physically connected, or not online, VTAM never receives the device end. In those cases, you should issue a VARY INACT,FORCE command to deactivate the link station.

Hardware Error Recording

During error recovery processing (ERP), VTAM writes outboard recorder (OBR) records and miscellaneous data records (MDRs) to MVS VM LOGREC or VSE SYSREC. OBR records are written for hardware errors on channel-attached devices. (VM VSE OBR records are written for communication adapter-attached devices as well.) MDR's or alert's are written for hardware errors on NCP-attached devices. See "Recording NMVT Alerts in LOGREC" on page 539 for more information on MDRs.

EREP formats and prints the MVS VM LOGREC or VSE SYSREC data set.

OBR records contain information about the following:

- Sense and status data on all channel-attached devices

- Failures on teleprocessing devices
- Temporary or intermittent failures on I/O devices
- End-of-day requests
- Permanent channel and device errors (unrecoverable errors and unit checks).

Permanent error records show the date, time, logical unit name, type of record, contents of counters, failing CCW, channel device name, CSW, sense information, device type, and flags. The time field shows the time at which the permanent error occurred.

Counter overflow and end-of-day records show the date, time, logical unit name, type of record (counter overflow or end-of-day), contents of counters, channel or unit address, and device type. The time field shows the time at which the counter overflow or end-of-day error occurred.

Counter overflow records are written when the temporary error counter or a device statistics table counter is about to overflow. VTAM maintains a counter for each channel-attached device. This counter keeps track of temporary errors. Counters of unit check errors by error type are also maintained in the device statistics table.

End-of-day records are written whenever a VARY INACT command is entered for a link or channel.

MDRs contain the following information:

- Statistics on the overflow of error counters for communication controllers
- Record maintenance statistics (RECMS) RUs
- Permanent errors on NCP-attached devices.

See the EREP entry in Table 80 on page 1165 to determine what book describes how to print and interpret MDR and OBR records.

Logical Unit Connection Test (IBMTEST)

You can enter the IBMTEST command from a terminal to find out whether that terminal can communicate with its owning SSCP. When you use the IBMTEST command, an unformatted RU is sent through the network path supporting the LU-SSCP session. This RU contains the IBMTEST command followed by the number of times the SSCP is to return (echo) the data to the logical unit and optional data (up to 247 bytes) being sent to the SSCP.

You can increase the possibility of repeating an intermittent error that is hard to re-create by using IBMTEST, because you can request up to 255 echoes. You can also use it to determine whether a suspended LU-LU session is caused by either a hardware problem or by a problem with VTAM or an application program.

Start this test with the following command:

IBMTEST [*n*][,*data*]

n specifies the number of times the test data should be returned to the terminal. Specify *n* as a decimal number in the range 1–255. If no value is specified, a value of 10 is used by default.

data specifies the test data to be returned. Specify a character string of up to 247 characters, or the maximum message length of the terminal, whichever is smaller. If no test data is supplied, VTAM returns the following alphanumeric sequence:

ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789

The IBMTEST command is valid only for terminals that use the USS LOGON format. The IBMTEST command must be defined in the USS table for that terminal.

Note: This echo check does not verify that a terminal can establish a session with an application program in the host, because the session request may specify a different network path than the one supporting the terminal's LU-SSCP session. If the requested path is unavailable, the session request is rejected, even though another path is available.

If there are any errors, the VTAM operator receives a message that contains the logical unit name associated with the terminal, the number of echoes that took place before the I/O error, and the error sense code.

NCP Error Recording

NCP error recording procedures create record maintenance statistics (RECMS) RUs that contain:

- The initial error status that began the recovery process
- The final error status that caused the permanent RECMS RU to be generated.

RECMS RUs are created for each adapter check, program check, unresolved interruption, counter overflow, I/O operation, and permanent line error. The RECMS RUs, which contain the miscellaneous data record (MDR), are sent to the VTAM host that owns the failing component. VTAM then records the MDR and writes the error information to the LOGREC or SYSREC file.

Patch Areas

Patch areas are available for VTAM, TSO/VTAM, and VSCS.

VTAM Patch Area

VTAM supplies a fixed patch area as a separate module. You can insert short service programs in this area to do maintenance-related functions. ISTPATCH is loaded into LPALIB during initialization of the operating system and is pointed to by the ATCPTCHA field in the ATCVT. The initial size of the patch area is 64 bytes, but you can change the size by link-editing a module named ISTPATCH of the desired size.

Code the necessary change in the patch area. (ISTPATCH follows the same coding rules as other modules in LPALIB.) Then replace part of the failing code with a branch to the patch area, allowing you to bypass the failing code.

TSO/VTAM Patch Area (MVS)

TSO/VTAM maintains a patch area in each module. The size of the patch area varies from module to module. If you need more information on using these patch areas, contact the IBM Support Center.

VSCS Patch Area (VM)

VSCS supplies a fixed patch area as a separate module. You can insert short service programs in this area to do maintenance-related functions. DTIPATCH is addressable from all VSCS code modules and is pointed to in the VSCS internal trace table header.

VTAM Load Module List

VTAM has a module list pointed to by ATCMDLST in the ATCVT. Each 16-byte entry in the list contains the following information in the form of

XXXXXYYYYYYYAAAA, where

- XXXXX is the five significant characters of the module name
- YYYYYYY is the PTF level (or Julian date if PTF level is not present)
- AAAA is the address of the module in storage.

Following is an example of some module list entries.

```
C1C9C3C1 D9E4E8F9 F3F7F4F4 00D8ACE8  *AICARUY93744.Q.Y*
C1C9C3E5 C340F9F1 4BF2F0F4 00D8C490  *AICVC 91.204.QD.*
C1C9C3C9 D6E4E8F9 F4F2F8F9 00D8A4C8  *AICIOUY94289.QuH*
C1C9C3C9 D9E4E8F8 F4F2F9F3 00D87000  *AICIRUY84293.Q..*
C1C9C3E7 D440F9F1 4BF0F8F9 00D8C288  *AICXM 91.089.QBh*
C1D7C3D2 E440F9F1 4BF2F9F5 00D885C0  *APCKU 91.295.Qe.*
C1D7C3D9 E440F9F1 4BF0F9F2 00D8C190  *APCRU 91.092.QA.*
C1D7C3E2 D940F9F1 4BF0F8F9 00D8BE60  *APCSR 91.089.Q.-*
C1D7C3E2 E4E4E8F8 F5F3F8F7 00D8B9D8  *APCSUUY85387.Q.Q*
C1D7C3E4 C5E4E8F9 F2F9F1F5 00D88E78  *APCUEUY92915.Q..*
```

You can use this module list table to

- Determine issuer entries in VIT records
- Search for save area base registers for modules that reside in LPA
- Verify PTF levels of modules.

Using Save Area Module Linkage Conventions—Subarea

VTAM traces the flow of the execution of three VTAM components, SSCP, PUS, and LUS, by saving the work areas of modules in these components. The addresses of the module work areas are stored in either of these control blocks:

- Network configuration services parameter list (NCSPL)
- Request/response unit processing element (RUPE).

In the RUPE the work area address can be found at RUPEDAP. In the NCSPL the work area address can be found at NCSPL.WKA. For the hex offsets of these fields, see *VTAM Data Areas for MVS/ESA*, *VTAM Data Areas for VM/ESA*, or *VTAM Data Areas for VSE/ESA*.

The NCSPL or RUPE work area contains the work and save areas for each module invoked for the command that the NCSPL or RUPE represents. The module work and save areas provide status information that pertains to both the processing of that command and any interruptions in the processing.

This status information includes a record of which modules were entered, which modules returned to their callers, and which modules returned with a return code. Each module save area contains the 4th, 5th, 7th, and 8th characters of the module name and the register 15 value that includes a pointer to the last module called by this module. If this address is not in the dump, the module can be obtained by comparing the address to the addresses in VTAM's module list pointed to by ATCMDLST out of the ATCVT (see 552).

The high-order byte of the register 15 save area also indicates the status of the last module called. (In 31-bit mode the address fills register 15, causing the status to overlay the high-order byte of the address in the register 15 save area.)

Byte Value	Status Indicated
FE	The called module has returned to this module without a return code.
FF	The called module has not returned to this module.
nn	The called module has returned to this module with a return code of nn.

Figure 79 is an example of what the NCSPL or RUPE work area might contain for modules invoked for a VTAM process using save area module linkage conventions. Using this convention, the save area contains a 4-byte module identifier, such as ACRT, at the location pointed to by register 13 for each entry in the chain.

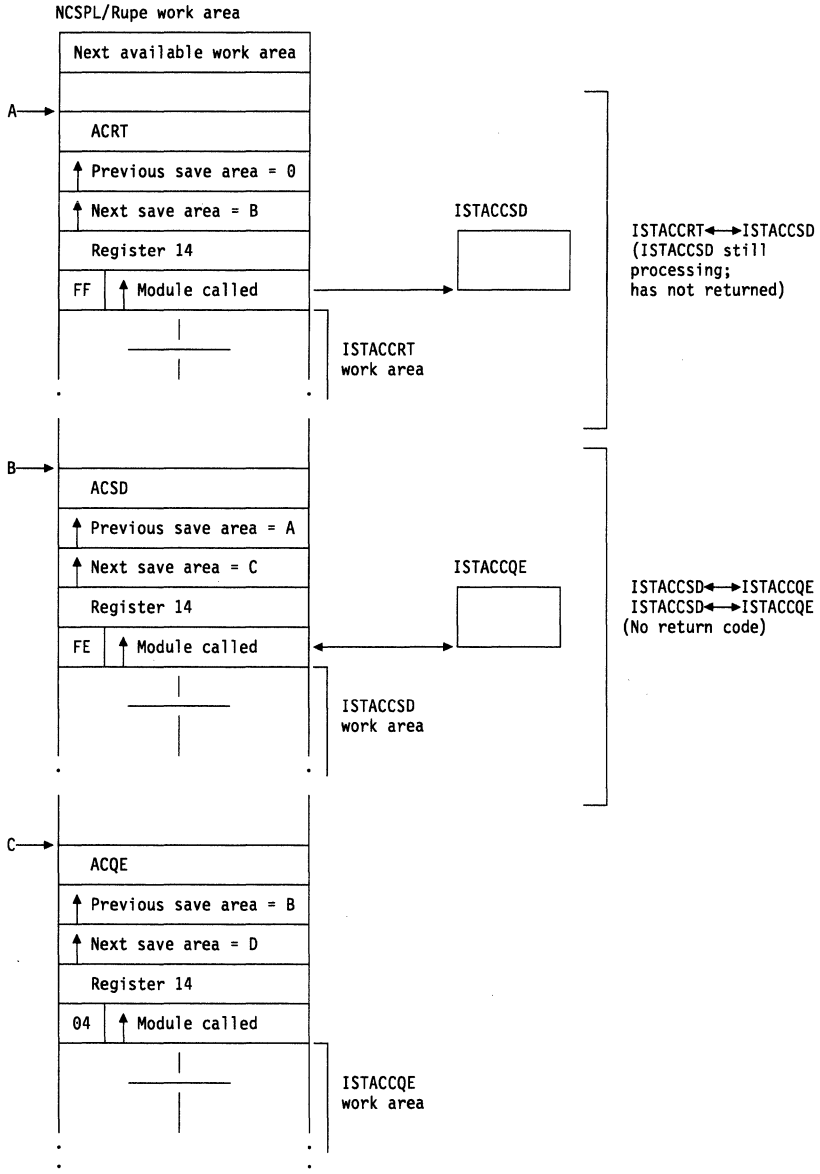


Figure 79. Save Area Module Linkage Conventions—Subarea

Using Save Area Module Linkage Conventions—APPN

Figure 80 on page 556 shows the save area module linkage convention for APPN used by some VTAM modules. To determine the convention used for your module find the location pointed to by register 13 and check the 8-byte field preceding this address. If you find an 8-byte module name, such as ISTACCRT your module was coded using the save area module linkage convention for APPN. The first three characters will always be IST for a VTAM module.

The addresses of the module work areas for modules using the save area module linkage convention for APPN are stored in process scheduling services (PSS) control blocks. The first word of the save area, pointed to by register 13 contains the pointer to the VTAM work area (VWA) header in the PSS.

For the save area module linkage convention for APPN, the following save area format pointed to by the address in register 13 is used.

Offset	Contents
X'-08'	Module eye-catcher C'XXXXXXXX'
X'+00'	Address of ISTVWA
X'+04'	Backward save area pointer (to previous save area)
X'+08'	Pointer to next available area
X'+0C'...	Registers 14 - 12

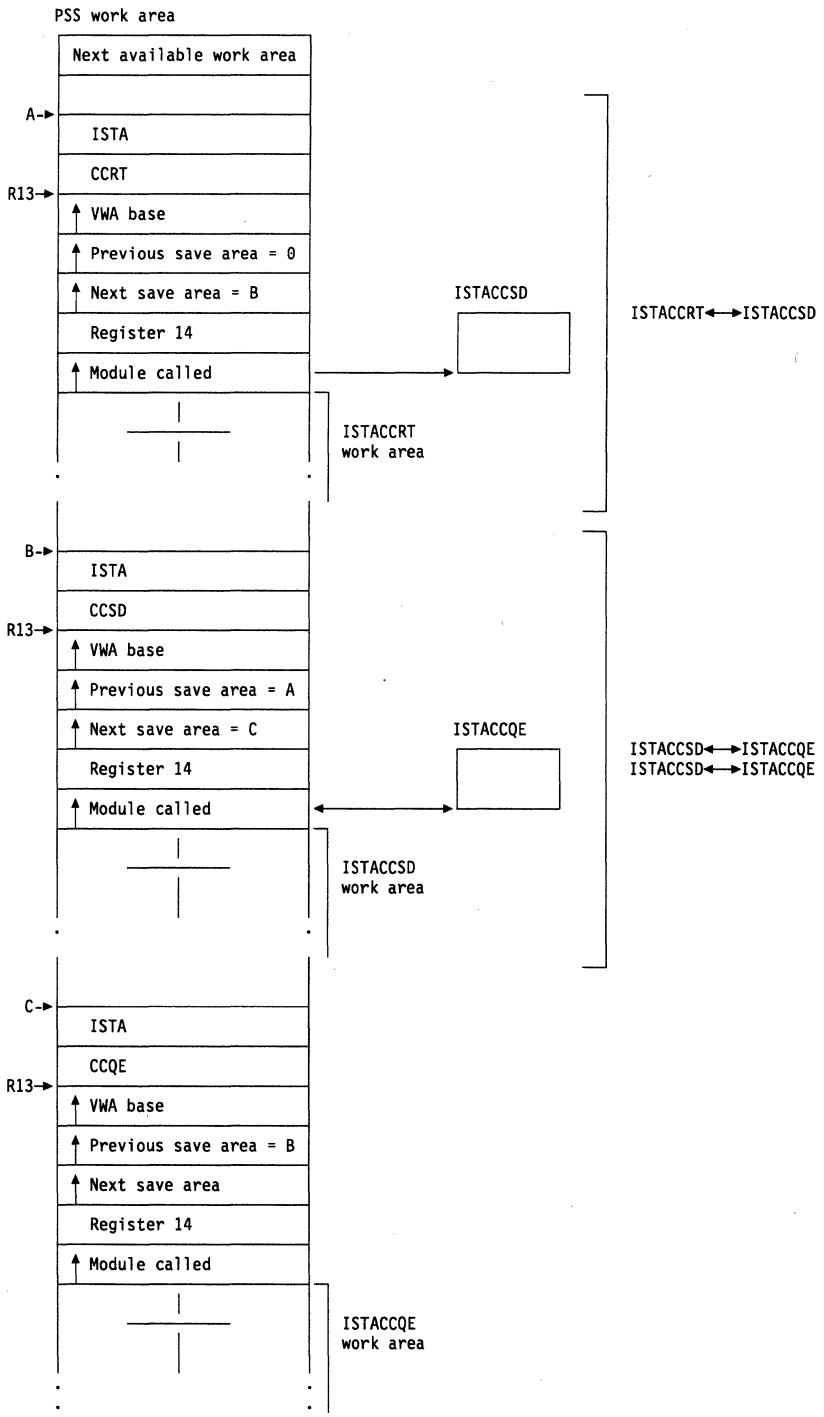


Figure 80. Save Area Module Linkage Conventions—APPN

VTAM Error Recording for Integrated Communication Adapter Lines (VM) (VSE)

VTAM error recording procedures create record maintenance statistics (RECMS) RUs to record information about the following conditions:

- Recovered (temporary) switched SDLC link and station errors
- Recovered (temporary) BSC link and station errors

- Unrecoverable (permanent) switched SDLC link and station errors
- Unrecoverable (permanent) BSC link and station errors
- BSC 3270 sense/status data.

RECMS RUs are sent to a communication network management (CNM) application program, if one is designated to receive these RUs.

VM For SDLC nonswitched lines, VTAM builds an NMVT instead of an RECMS.

VSE VTAM also writes the RECMS records on SYSREC.

See Table 80 on page 1165 to determine what book describes formats for RECMS RUs.

Use EREP to edit and print the error data. See the EREP entry in Table 80 on page 1165 to determine what book describes how to edit and print error data.

Part 3. Diagnostic References

Appendix A. VTAM Internal Trace (VIT) Record Descriptions

This appendix contains the VTAM internal trace (VIT) record descriptions. The entries are listed alphabetically by entry name.

Table 34 summarizes the VIT options and the records they create.

Table 34. VIT Options and the Records They Create

API	APPC	CFS MVS	CIO	ESC	LCS	LOCK	MSG	NRM	PIU	PSS	SMS	SSCP	TCP MVS	VCNS MVS VM
AIx	ACAx	CFCx	ADPx	ESC	LCSx	LKEX	**MSGx**	**BSPx**	**DSCx**	ADSP	AREL	**ALSx**	IPx	CNA
IOx	ACIx	CFDx	ATTx			LKSH	**OPEx**	**BSSx**	**NRSx**	ATSK	EXPN	**AP**	MIx	CNPx
RE	ACPx	CFEx	CONN			ULKA	**QRYL**	**BSXx**	**PIUx**	AXIT	FBLx	**A2**	MPTx	CNRx
UEx	ACRx	CFFC	CPTO			UNLK	**TRNx**	**PROx**	**RDSx**	BTSK	GBLx	**CCx**	NMPx	NSD
UP	ACSN	CFLx	DISC					**RCEx**		DSP	QREQ	**CIx**	SOCx	VCCx
	ACUx	CFNF	ERPx					**SRTx**		DTSK	RELS	**COx**		VCDQ
	MUx	CFRB	HIOx							ETSK	REQx	**CPPx**		VCPx
	RACR	CFUS	INTx							EXIT	VTAL	**CPRx**		
			LICx							IRBx	VTFR	**CPWx**		
			LNKx							POST		**CRx**		
			LSNx							QUEx		**DBx**		
			PIC							RESM		**DLTx**		
			PKTx							SCHD		**GNAx**		
			SIOx							SRBx		**SPTx**		
			XIDx							VPST		**TGMx**		
										VRSM		**TREx**		
										VWAI				
										WAIT				
										XPST				

Notes:

1. The **** (FFST), ABND, and LOST trace records are not activated by specific VIT options. They are activated as a result of exception conditions.
2. The entries in boldface type are forced, that is, always traced even if they are not specified by the user.

For more information on VIT options, see "Selecting Trace Options" on page 398.

**** Entry for VTAM FFST

Entry:	**** (FFST)
VIT Option:	None (Generated by VTAM)
Event:	VTAM probe triggered
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	ISTRACZE
Applies to:	MVS VM

The **** (FFST) trace record is written when VTAM disables the VTAM internal trace (VIT) to collect information for a probe that has been triggered. This entry is not associated with any VIT options but is recorded when an unusual condition triggers a probe instruction.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
****					'VIT DISABLED FOR VTAM/FFST'																										

Byte (hex)	Contents
00–03	Record ID: C'****'
04	Blank
05–1E	C'VIT DISABLED FOR VTAM/FFST'
1F	Blank.

ABND Entry for Abend SNAP Routine

Entry:	**ABND**
VIT Option:	None (Generated by SNAP routine)
Event:	Abend
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	VTAM abend recovery routine (many possible)
Applies to:	MVS VM

This trace record is written when an abend occurs in a VTAM module. This entry is not associated with any VIT options but is recorded as an exception condition when an abend occurs.

Note: VM If the abend occurred as the result of a program exception in a channel end appendage, the request parameter header (RPH) address will be for the last PAB dispatch and not for the request parameter header (RPH) of the channel end appendage.

0 0 0 0 0 1 2 3	0 0 0 0 0 4 5 6 7	0 0 0 0 0 8 9 A B	0 0 0 0 0 C D E F	1 1 1 1 1 0 1 2 3	1 1 1 1 1 4 5 6 7	1 1 1 1 1 1 1 1 1 8 9 A B C D E F
ABND	ABEND CODE	PST ADDRESS or 0	RPH ADDRESS or 0	SDWA ADDRESS or 0	SDWA FLAGS or 0	ABEND PSW or 0

Byte (hex)	Contents
00–03	Record ID: C'ABND'
04–07	Abend completion code
08–0B	PST address or zero
0C–0F	Request parameter header (RPH) address or zero
10–13	SDWA address or zero
14–17	SDWA flags or zero
18–1F	Abend PSW or zero.

ACA1 or ACI1 Entry for LU 6.2 Authorized IO or LU 6.2 TPIO (Part 1)

Entry:	**ACA1 or ACI1**
VIT Option:	APPC
Event:	LU 6.2 authorized IO or LU 6.2 TPIO (Part 1)
VIT Processing Module:	ISTRACAC
Control is Returned to:	ISTAICAR
Applies to:	MVS VM VSE

The ACA1 trace record shows LU 6.2 requests under the authorized path or LU 6.2 requests not under the authorized path but issued by programs running under an authorized key. The ACI1 trace record shows LU 6.2 requests running under a normal path.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ACA1 or ACI1				ID	EXTDS	OPT1	OPT6	RPL ADDRESS				CNTRL	QUAL	BUFL		RPL AREA				APPCCMD FLAGS or 0				USR FLD or CID or CONVID or 0				SENSE DATA or 0			

Byte (hex) Contents

00–03 Record ID:
- C'ACA1' for authorized path
- C'ACI1' for normal path

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Exit definition (RPLEXTDS)

06 Option code byte 1 (RPLOPT1)

07 Option code byte 6 (RPLOPT6)

08–0B RPL address

0C APPCCMD CONTROL operand value:
X'10' ALLOC
X'20' RESETRCV
X'30' DEALLOC
X'31' DEALLOCQ
X'40' OPRCNTL
X'50' PREPRCV
X'60' RCVFMH5
X'70' RECEIVE
X'71' RCVEXPD
X'80' REJECT
X'90' SEND
X'91' SENDEXPD
X'A0' SETSESS
X'B0' TESTSTAT

0D APPCCMD QUALIFY operand value:
X'01' ABNDPROG

X'02' ABNDSERV
X'03' ABNDTIME
X'04' ABNDUSER
X'05' ANY
X'06' CNOS
X'07' CONFIRM
X'08' CONFRMD
X'09' DATA
X'0A' DATACON
X'0B' DATAFLU
X'0C' DEFINE
X'0D' DISPLAY
X'0E' ERROR
X'0F' FLUSH
X'10' RQSEND
X'11' SPEC
X'12' ACTSESS
X'13' DACTSESS
X'14' ALLOCD
X'15' IMMED
X'16' CONWIN
X'17' SESSION
X'18' CONV
X'19' SUSPEND
X'1A' RESUME
X'1B' RESTORE
X'1C' SYNCBEG
X'1D' SYNCEND
X'1E' CONVGRP
X'1F' WHENFREE
X'20' IANY
X'21' ISPEC
X'22' ALL
X'23' IALL

0E–0F User buffer length (RPLBUFL) for receive RPL or zero

10–13 Address of SEND data or RECEIVE buffer

14–17 APPCCMD flags (RPL6FLGS) or zero

18–1B User field for LU 6.2 commands where CONTROL = either ALLOC or RCVFMH5. CID for LU 6.2 commands where CONTROL = OPRCNTL and QUALIFY = either ACTSESS or DACTSESS. Zero for LU 6.2 commands for which one of the following is true:

- CONTROL = OPRCNTL, and QUALIFY = either CNOS, DEFINE, or DISPLAY
- CONTROL = RECEIVE and QUALIFY = ANY
- CONTROL = REJECT and QUALIFY = SESSION or CONVGRP
- CONTROL = SETSESS and QUALIFY = RESUME

Conversation ID for all other LU 6.2 commands

1C–1F Sense data (RPL6SNSO) or zero.

ACA2 or ACI2 Entry for LU 6.2 Authorized IO or LU 6.2 TPIO (Part 2)

Entry: **ACA2 or ACI2**
VIT Option: APPC
Event: LU 6.2 authorized IO or LU 6.2 TPIO (Part 2)
VIT Processing Module: ISTRACAC
Applies to: MVS VM VSE

The ACA2 trace record is a continuation of ACA1. The ACI2 trace record is a continuation of ACI1.

0 0 0 0 0 1 2 3	0 0 0 0 4 5 6 7	0 0 0 0 0 0 0 0 8 9 A B C D E F	1 1 1 1 1 1 1 1 0 1 2 3 4 5 6 7	1 1 1 1 1 1 1 1 8 9 A B C D E F
ACA2 OR ACI2	RETURN ADDRESS	See description.	LOCAL LU NAME, 0, OR BLANKS	PARTNER LU NAME, 0, OR BLANKS

Byte (hex) Contents

00–03 Record ID:
- C'ACA2' for continuation of ACA1
- C'ACI2' for continuation of ACI1

04–07 Address of the issuer of the APPCCMD macro

08–0F
- When CONTROL=ALLOC or REJECT, and QUALIFY=CONVGRP, then 08–0B is the conversion group ID (CGID), and 0C–0F is zero.
- When CONTROL = ALLOC (except when QUALIFY = CONVGRP) or when CONTROL = OPRCNTL, and QUALIFY = either CNOS, DEFINE, or DISPLAY, then 08–0F is the mode name for LU 6.2 commands.
- When CONTROL = SETSESS or when CONTROL = REJECT, and QUALIFY = SESSION, then 08–0F is the session instance identifier for LU 6.2 commands.

10–17 Local logical unit name for LU 6.2 commands where CONTROL = ALLOC, or where CONTROL = OPRCNTL, and QUALIFY = either CNOS, DEFINE, or DISPLAY.

Otherwise, this field contains blanks or zero.

18–1F Partner logical unit name for LU 6.2 commands where CONTROL = ALLOC, or where CONTROL = OPRCNTL, and QUALIFY = either CNOS, DEFINE or DISPLAY.

This field contains blanks or zero. when CONTROL=ALLOC or REJECT, and QUALIFY=CONVGRP.

ACA3 or ACI3 Entry for LU 6.2 Authorized IO or LU 6.2 TPIO (Part 3)

Entry: **ACA3 or ACI3**
VIT Option: APPC
Event: LU 6.2 authorized IO or LU 6.2 TPIO (Part 3)
VIT Processing Module: ISTRACAC
Applies to: MVS VM VSE

The ACA3 trace record is a continuation of ACA2. The ACI3 trace record is a continuation of ACI2.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ACA3 OR ACI3				0																				PARTNER NETID, 0, OR BLANKS							

Byte (hex) Contents

00–03 Record ID:
- C'ACA3' for continuation of ACA2
- C'ACI3' for continuation of ACI2

04–17 Zero

18–1F Partner network identifier for the LU 6.2 commands where CONTROL = ALLOC (except for CONTROL=ALLOC,QUALIFY=CONVGRP), or where control = OPRCNTL, and QUALIFY = either CNOS, DEFINE, or DISPLAY. Otherwise, this field contains blanks or zero.

ACP1 or ACR1 Entry for LU 6.2 User Post or RPL Exit (Part 1)

Entry:	**ACP1 or ACR1**
VIT Option:	APPC
Event:	LU 6.2 user post or RPL exit (Part 1)
VIT Processing Module:	ISTRACAC
Control is Returned to:	ISTAICPT and ISTAPCSX for ACP1 ISTAPCSX and ISTAPCUE for ACR1
Applies to:	MVS VM VSE

The ACP1 trace record is written when an ECB is posted. The ACR1 trace record is written when an RPL exit is dispatched. These records signal that the APPCCMD macro has completed execution and show the data returned to the user application program.

00 01 02 03	04	05	06	07	08 09 0A 0B	0C	0D	0E 0F	10 11	12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
ACP1 OR ACR1	ID	RTUN	RTNCD	FDB2	RPL ADDRESS	CNTRL	QUAL	0	RCPRI	RCSEC	APPCCMD FLAGS	CONV ID OR 0	SENSE DATA OR 0

Byte (hex) Contents

00–03 Record ID:
- C'ACP1' for ECB posting
- C'ACR1' for RPL exit dispatching

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 APPCCMD returned indicators (RPL6RTUN)

Bit	Meaning
1...	FMH5 received indicator
.1..	LOGON received indicator
..1.	SIGNAL received indicator

06 VTAM return code (RPLRTNCD)

07 VTAM feedback code (RPLFDB2)

08–0B RPL address

0C APPCCMD CONTROL operand value:

X'10'	ALLOC
X'20'	RESETRCV
X'30'	DEALLOC
X'31'	DEALLOCQ
X'40'	OPRCNTL
X'50'	PREPRCV
X'60'	RCVFMH5
X'70'	RECEIVE
X'71'	RCVEXPD
X'80'	REJECT
X'90'	SEND

	X'91' SENDEXPD
	X'A0' SETSESS
	X'B0' TESTSTAT
0D	APPCCMD QUALIFY operand value:
	X'01' ABNDPROG
	X'02' ABNDSERV
	X'03' ABNDTIME
	X'04' ABNDUSER
	X'05' ANY
	X'06' CNOS
	X'07' CONFIRM
	X'08' CONFRMD
	X'09' DATA
	X'0A' DATACON
	X'0B' DATAFLU
	X'0C' DEFINE
	X'0D' DISPLAY
	X'0E' ERROR
	X'0F' FLUSH
	X'10' RQSEND
	X'11' SPEC
	X'12' ACTSESS
	X'13' DACTSESS
	X'14' ALLOCD
	X'15' IMMED
	X'16' CONWIN
	X'17' SESSION
	X'18' CONV
	X'19' SUSPEND
	X'1A' RESUME
	X'1B' RESTORE
	X'1C' SYNCBEG
	X'1D' SYNCEND
	X'1E' CONVGRP
	X'1F' WHENFREE
	X'20' IANY
	X'21' ISPEC
	X'22' ALL
	X'23' IALL
0E–0F	Zero
10–11	LU 6.2 primary return code (RPL6RCPR)
12–13	LU 6.2 secondary return code (RPL6RCSC)
14–17	APPCCMD flags (RPL6FLGS)
18–1B	Conversation ID or zero
1C–1F	Sense data returned (RPL6SNSI) or zero.

ACP2 or ACR2 Entry for LU 6.2 User Post or RPL Exit (Part 2)

Entry: **ACP2 or ACR2**
VIT Option: APPC
Event: LU 6.2 user post or RPL exit (Part 2)
VIT Processing Module: ISTRACAC
Applies to: MVS VM VSE

The ACP2 trace record is a continuation of the ACP1 trace record. The ACR2 trace record is a continuation of the ACR1 trace record.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ACP2 OR ACR2				ECB ADDRESS OR EXIT ADDRESS				RPL6 ADDRESS				CNTRL	0	RCVIN	0	RPL AREA				RPL RECORD LENGTH				CGID				0			

Byte (hex) Contents

00–03 Record ID:
- C'ACP2' for continuation of ACP1
- C'ACR2' for continuation of ACR1

04–07 ECB address for ACP2
RPL exit address for ACR2

08–0B RPL6 address

0C APPCCMD CONTROL operand value:

Bit	Meaning
X'10'	ALLOC
X'20'	RESETRCV
X'30'	DEALLOC
X'31'	DEALLOCQ
X'40'	OPRCNTL
X'50'	PREPRCV
X'60'	RCVFMH5
X'70'	RECEIVE
X'71'	RCVEXPD
X'80'	REJECT
X'90'	SEND
X'91'	SENDEXPD
X'A0'	SETSESS
X'B0'	TESTSTAT

0D Zero

0E Indicators for the type of information received or zero

Bit	Meaning
1...	DATA
.1..	DATA_COMPLETE
..1.	DATA_INCOMPLETE
...1	SEND
.... 1...	CONFIRM
.... .1..	DEALLOCATE

0F	Zero
10–13	Address of SEND data or RECEIVE buffer (RPLAREA)
14–17	RPL record length
18–1B	Conversation group ID (CGID) when CONTROL=ALLOC or RCVFMH5; otherwise, zero
1C–1F	Zero.

ACRC Entry for RPL6 Return Code

Entry:	**ACRC**
VIT Option:	APPC
Event:	APPCSNRC macro
VIT Processing Module:	ISTRACAC
Control is Returned to:	Issuer of the APPCSNRC macro
Applies to:	MVS VM VSE

This trace record is written when an APPC module issues a macroinstruction to set an RPL6RC non-zero return code.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ACRC				ID	0		INSTANCE ID	CONVERSATION ID				HALF-SESSION ID				MODULE NAME				CONTROL BLOCK ADDRESS				RETURN CODE				RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'ACRC'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05–06	Zero
07	Instance indentifier in invoking module
08–0B	Conversation ID
0C–0F	Half-session ID
10–13	Invoking module name
14–17	Address of control block containing return code
18–1B	Return code (RPL6RC) **18–19** Primary return code (RPL6RCPR) **1A–1B** Secondary return code (RPL6RCSC)
1C–1F	RPH address.

ACSN Entry for APPC Sense Code

Entry:	**ACSN**
VIT Option:	APPC
Event:	APPCSNRC macro
VIT Processing Module:	ISTRACAC
Control is Returned to:	Issuer of the APPCSNRC macro
Applies to:	MVS VM VSE

This trace record is written when an APPC module issues a macroinstruction to set a non-zero sense code.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ACSN				ID	0		INSTANCE ID	CONVERSATION ID				HALF-SESSION ID				MODULE NAME				CONTROL BLOCK ADDRESS				SENSE CODE OR RETURN CODE				RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'ACSN'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05–06	Zero
07	Instance indentifier in invoking module
08–0B	Conversation ID
0C–0F	Half-session ID
10–13	Invoking module name
14–17	Address of control block containing sense code
18–1B	Sense code or internal return code
1C–1F	RPH address.

ACU1 Entry for LU 6.2 User Exit (Part 1)

Entry: **ACU1**
VIT Option: APPC
Event: LU 6.2 user exit (Part 1)
VIT Processing Module: ISTRACAC
Control is Returned to: ISTAPCUE
Applies to: MVS VM VSE

This trace record is written when an LU 6.2 user exit (either ATTN or TPEND) is dispatched. If both the API and APPC trace options are active, and a TPEND user exit is dispatched, VTAM generates user exit trace records for both API and APPC.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ACU1				ID	CODE	0		EXIT TYPE DEPEND INFO				0				PLU NAME OR APPLICATION ID								SLU NAME OR 0							

Byte (hex) Contents

00–03 Record ID: C'ACU1'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Exit code for user exit:
X'0C' for ATTN
X'0D' for TPEND

06–07 Zero

08–0B The type of exit determines the value of this field.
- For ATTN exit, the exit sub-type (CNOS, FMH5, or LOSS)
- For TPEND exit, byte 8 is the reason code:

Code	Description
X'00'	Standard HALT command issued to close the network
X'04'	HALT QUICK command or VARY INACT,I or F command issued for the application program name
X'08'	HALT CANCEL command issued or VTAM terminated abnormally
X'0C'	MVS Alternate application issued an OPEN ACB for the same ACB this application has opened

0C–0F Zero

10–17 PLU name for ATTN exit; application program ID for TPEND exit

18–1F SLU name for ATTN exit; zero for TPEND exit.

ACU2 Entry for LU 6.2 User Exit (Part 2)

Entry: **ACU2**
VIT Option: APPC
Event: LU 6.2 user exit (Part 2)
VIT Processing Module: ISTRACAC
Applies to: MVS VM VSE

This trace record is a continuation of the ACU1 trace record.

0 0 0 0 0 1 2 3	0 0 0 0 0 4 5 6 7	0 0 0 0 0 0 0 0 0 8 9 A B C D E F	1 1 1 1 1 1 1 1 0 1 2 3 4 5 6 7	1 1 1 1 1 1 1 1 1 8 9 A B C D E F
ACU2	EXIT ADDRESS	MODE NAME OR 0	0	NETID OR 0

Byte (hex)	Contents
00–03	Record ID: C'ACU2'
04–07	Exit address
08–0F	Mode name for ATTN exit; zero for TPEND exit
10–17	Zero
18–1F	Network identifier for ATTN exit; zero for TPEND exit.

ADPx Entry for Adapter I/O (Part 1)

Entry:	**ADPI, ADPO, or ADPA**
VIT Option:	CIO
Event:	Adapter I/O (Part 1)
VIT Processing Module:	ISTRAC6T
Control is Returned to:	ISTTSC6B
Applies to:	VM VSE

This trace record is created when a data transfer with the Token-Ring Subsystem is started or completed.

An ADPI record is written when an inbound transfer has been received from the subsystem. An ADPI trace entry may be followed by one or more ADP2 entries.

An ADPO record is written when an outbound data transfer is complete. This entry always corresponds to a previous LICO entry. Use the TSCB field to correlate these entries. An ADPO trace entry may be followed by one or more ADP2 entries.

An ADPA record is written when the subsystem acknowledges that an outbound data transfer is complete. This entry always corresponds to a previous ADPO entry. Use the TSCB field to correlate these entries.

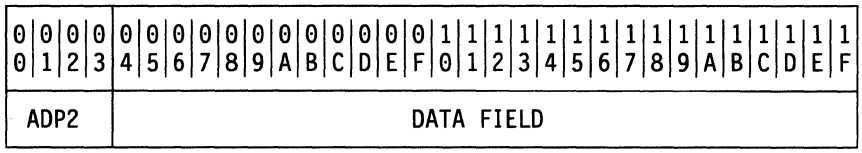
00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ADPI ADPO ADPA				ID	CUA			TSCB ADDRESS				NCB ADDRESS				CBID	ACK	INDX	PACK	FRNT	END	0		DATA FIELD							

Byte (hex)	Contents
00–03	Record ID: C'ADPA' for an acknowledge record C'ADPI' for an inbound record C'ADPO' for an outbound record
04	VM Machine ID and Task ID VSE Task ID
05–07	Channel device name in EBCDIC (either a CUA or device number)
08–0B	TSCB address
0C–0F	NCB address
10	Control block ID
11	Index value acknowledged by subsystem
12	Last index value presented to subsystem
13	Last index value acknowledged by subsystem
14	Index for the front pointer of the buffer ring
15	Index for the end pointer of the buffer ring
16–17	Zero
18–1F	Data field.

ADP2 Entry for Adapter I/O (Part 2)

Entry: **ADP2**
VIT Option: CIO
Event: Adapter I/O (Part 2)
VIT Processing Module: ISTRAC6T
Applies to: VM VSE

This trace record is a continuation of the ADPI or ADPO entries.



Byte (hex)	Contents
00–03	Record ID: C'ADP2'
04–1F	Data field.

ADSP Entry for Asynchronous Dispatch

Entry:	**ADSP**
VIT Option:	PSS
Event:	Asynchronous dispatch
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCAD
Applies to:	VM VSE

This trace record is written whenever PSS receives control because work has been scheduled (TREADY COND=VTAM).

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ADSP				ID	0	PST FLAGS		PST ADDRESS				FLAG OR 0	FLAG OR 0	0		SYNC TPPOSTD QUEUE				SYNC NORMAL QUEUE				ASYNC TPPOSTD QUEUE				ASYNC NORMAL QUEUE			

Byte (hex)	Contents
00–03	Record ID: C'ADSP'
04	VM Machine ID and Task ID VSE Task ID
05	Zero
06–07	PST Flags (PSTFLG1)
08–0B	PST address
0C	VM Zero VSE AOT Flag A
0D	VM Zero VSE AOT Flag B
0E–0F	Zero
10–13	Synchronous TPPOSTed queue (PSTSTPP)
14–17	Synchronous normal queue (PSTSNRM)
18–1B	Asynchronous TPPOSTed queue (PSTATPP)
1C–1F	Asynchronous normal queue (PSTANRM).

AI1 Entry for Authorized IO (Part 1)

Entry:	**AI1**	
VIT Option:	API	
Event:	Authorized IO (Part 1)	
VIT Processing Module:	ISTRACAP	
Control is Returned to:	ISTAICIR for:	ISTOCCSM for:
	SEND	INQUIRE
	RECEIVE	INTRPRET
	RESETSR	RCVCMD
	SESSIONC	SENDCMD
		SETLOGON
		SESSIONC (BIND)
		SIMLOGON
		OPNDST
		REQSESS
		OPNSEC
		TERMSESS
		CLSDST
Applies to:	MVS VM	

This trace record shows API requests under the authorized path or API requests not under the authorized path but issued by programs running under an authorized key. The API routine ISTAICIR or ISTOCCSM writes the entry and then queues the RPL to the correct PAB. For an explanation of the RPL fields, see *VTAM Data Areas for MVS/ESA* or *VTAM Data Areas for VM/ESA*.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
AI1			REQTY	ID		EXTDS	OPT4	OPT1	RPL ADDRESS			NIB ADDR OR CID				RH3	SRTYP	VTFL1	VTFL2	CHN	RU CNTL CODE			OPTC2				OPTC3			

Byte (hex) Contents

00–02 Record ID: C'AI1'

03 RPL request type (RPLREQ)

- X'15' SETLOGON
- X'16' SIMLOGON
- X'17' OPNDST
- X'19' CHANGE
- X'1A' INQUIRE
- X'1B' INTRPRET
- X'1F' CLSDST
- X'22' SEND
- X'23' RECEIVE
- X'24' RESETSR
- X'25' SESSIONC
- X'27' SENDCMD
- X'28' RCVCMD
- X'29' REQSESS

	X'2A' OPNSEC X'2C' TERMSESS
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM
05	Exit definition (RPLEXTDS)
06	Option code byte 4 (RPLOPT4)
07	Option code byte 1 (RPLOPT1)
08–0B	RPL address
0C–0F	NIB address or CID
10	Third byte of RH (RPLRH3)
11	Send/receive type (RPLSRTYP)
12	VTAM flags (RPLVTFL1)
13	Post/respond flags (RPLVTFL2)
14	RU chain position (RPLCHN)
15–17	RU control codes (RPLCNTRL)
18–1B	VTAM options (RPLOPTC2, which corresponds to RPLOPT5 through RPLOPT8) (See Appendix E, "Control Block Formats and DSECTs" in *VTAM Programming* for additional information.)
1C–1F	VTAM options (RPLOPTC3, which corresponds to RPLOPT9 through RPLOPT12). (See Appendix E, "Control Block Formats and DSECTs" in *VTAM Programming* for additional information.)

AI2 Entry for Authorized IO (Part 2)

Entry:	**AI2**
VIT Option:	API
Event:	Authorized IO (Part 2)
VIT Processing Module:	ISTRACAP
Applies to:	MVS VM

This trace record is a continuation of the AI1 entry. It shows additional information about the PLU and SLU in a session established or terminated by a SIMLOGON, OPNDST, CLSDST, REQSESS, OPNSEC, or TERMSESS macroinstruction. This information includes some of the parameters exchanged and the names of the PLU and SLU.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
AI2			0	RETURN ADDRESS				LOGON MODE NAME OR 0								APPLICATION LU NAME (if available)								PARTNER LU (if available)							

Byte (hex)	Contents
00–02	Record ID: C'AI2'
03	Zero
04–07	Address of the issuer of the macro
08–0F	For SIMLOGON, OPNDST, CLSDST, or REQSESS: Logon mode name (NIBLMODE) or zero if the NIB is not available For OPNSEC or TERMSESS: Zero
10–17	Application LU name if available
18–1F	Partner LU name if available.

AI3 Entry for Authorized IO (Part 3)

Entry:	**AI3**
VIT Option:	API
Event:	Authorized IO (Part 3)
VIT Processing Module:	ISTRACAP
Applies to:	MVS VM

This trace record is a continuation of the AI2 entry. It shows additional information in a session established or terminated by a SIMLOGON, OPNDST, CLSDST, REQSESS, or OPNSEC macroinstruction.

00–02	03	04–07	08–09	0A–0B	0C–0F	10–17	18–1F
AI3	0	ADDRESS OF USER DATA, BIND, OR 0	TARGET LU NETID OR NIB COUNT	(TARGET LU NETID OR) NIB RPARM COUNT	(TARGET LU NETID OR) 0	TARGET LU NAME (if available) OR 0	PARTNER LU NETID (if available)

Byte (hex)	Contents
00–02	Record ID: C'AI3'
03	Zero
04–07	For SIMLOGON, CLSDST, or REQSESS: Address of user data (RPLAREA) For OPNDST or OPNSEC: Address of BIND (NIBNDAR) or zero if the NIB is not available
08–0F	Target LU network identifier (when for CLSDST PASS).
	Byte (hex) Contents
	08–09 Count of node initialization blocks (NIBs) in NIB list (when not for CLSDST PASS).
	0A–0B Count of node initialization blocks (NIBs) in NIB list with NIBRPARM=0 (when not for CLSDST PASS).
10–17	Target logical unit name, if available, or zero
18–1F	Partner LU network identifier, if available.

ALSx Entry for Adjacent Link Station Add, Delete, Replace, Select, Validate

Entry:	**ALSA, ALSD, ALSR, ALSS, or ALSV**
VIT Option:	SSCP
Event:	ALSLIST macro
VIT Processing Module:	ISTRACSC
Control is Returned to:	Module invoking the INTRACE macro that caused the record to be produced.
Applies to:	MVS VM VSE

The adjacent link station (ALS) trace record contains information about the adding, deleting, replacing, selecting, or validating of an adjacent link station.

00	01	02	03	04	05	06	07	08 09 0A 0B 0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B 1C 1D 1E 1F
ALSA ALSD ALSR ALSS ALSV	ID	RC	CBID	FLAGS	NEWALS or SELALS or 0	RETURN ADDR	CDRADDR	OLDALS or 0 or 0 and PUADDR			

Byte (hex)

00–03 Record ID:
- C'ALSA': Add an adjacent link station
- C'ALSD': Delete an adjacent link station
- C'ALSR': Replace an adjacent link station
- C'ALSS': Select an adjacent link station
- C'ALSV': Validate an adjacent link station

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Return code from ALSLIST macro invocation

06 Session control block identifier or 0

07 Flags(from the following list):

Bit	Meaning
1...	The type of add is dynamic.
.1..	Autologon processing requested.
..1.	Verify that the add is required.
...1	Waiting autologon requests should be processed.
.... xx..	Zero
.... ..xx	Connection type of PU whose address is given in PUADDR: 00 Unknown 01 LEN 11 APPN

08–0F ALS name or zero
- For ALSA, ALSR, or ALSV: New ALS name passed on the add, replace, or validate function
- For ALSD: Zero

- For ALSS: ALS name returned on the select function

10–13 Return address of the caller of the ALSLIST function

14–17 Address of the cross-domain resource passed to the ALSLIST function

18–1F ALS name, PU address, or 0

- For ALSA: Zero
- For ALSD or ALSR: Old ALS name passed on the delete or replace function
- For ALSS or ALSV: 0 and PUADDR
  - 18–1B: Zero
  - 1C–1F: The PU address that is returned for the select or validate function.

AP Entry for Signals Passed Between Components of the APPN CP

Entry:	**AP**
VIT Option:	SSCP
Event:	Signals passed between components of the APPN CP
VIT Processing Module:	ISTRACSC
Control is Returned to:	Module invoking the INTRACE macro that caused the record to be produced.
Applies to:	MVS VM VSE

This trace record provides information about inbound request and responses which are processed by the CP. If this entry is associated with an event failure, non-zero sense code, it is generated regardless of the SSCP option. It is recognized as an exception code and is traced when the VIT is active.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
AP		TO PAB	FROM PAB	ID	MOD ID			RETURN ADDRESS				SENSE DATA OR 0				FIRST 16 BYTES OF INTERPROCESS SIGNAL (IPS)															

Byte (hex)	Contents
00–01	Record ID: C'AP'
02	To process anchor block (PAB) identification byte:

Code (hex) ID MVS

'A'	Advanced program-to-program communication (APPC)
'C'	Session services for CP-CP sessions (SSC)
'D'	APPN directory services (DR)
'H'	Half session (HS)
'I'	Disk input or output
'L'	Session services for LU-LU sessions (SSL)
'M'	Management services transport (MST)
'N'	Node operator services
'S'	System services control point (SSCP)
'T'	Topology and routing services (TRS)
'X'	Transaction programs (XP)
'Z'	Dependent LU server (DLUS)

Code (hex) ID VM VSE

'A'	Advanced program-to-program communication (APPC)
'C'	Session services for CP-CP sessions (SSC)
'D'	APPN directory services (DR)
'E'	Directory services management exit
'H'	Half session (HS)
'I'	Disk input or output
'J'	Multiple domain server router
'L'	Session services for LU-LU sessions (SSL)
'M'	Management services transport (MST)
'N'	Node operator services

	'P' I/O purge timer
	'S' System services control point (SSCP)
	'T' Topology and routing services (TRS)
	'X' Transaction programs (XP)
	'Z' Dependent LU server (DLUS)
03	From process anchor block (PAB) Identification byte. The codes are the same as byte 2.
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05–07	Last 3 characters of the module invoking APSEND. (The characters *KWN* indicate that the module name is not known.)
08–0B	Address of the issuer of the APSEND module
0C–0F	Sense code or zero
10–1F	First 16 bytes of the interprocess signal (IPS). For a list of interprocess signals, see *VTAM Data Areas for MVS/ESA*, *VTAM Data Areas for VM/ESA*, or *VTAM Data Areas for VSE/ESA*.

A2 Entry for Requests and Responses with a RUPE

Entry:	**A2**
VIT Option:	SSCP
Event:	Requests and Responses with a RUPE
VIT Processing Module:	ISTRACSC
Applies to:	MVS VM VSE

This trace record is a continuation of the AP entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
A2	30 MORE BYTES OF THE INTERPROCESS SIGNAL (IPS)																														

Byte (hex)	Contents
00–01	Record ID: C'A2'
02–1F	30 more bytes of the interprocess signal.

AREL Entry for Abend RELSTORE

Entry:	**AREL**
VIT Option:	SMS
Event:	Abend RELSTORE
VIT Processing Module:	ISTRACSM
Control is Returned to:	ISTORAPR
Applies to:	MVS VM

This trace record identifies the buffers that are released by VTAM when a VTAM application program is terminated.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
AREL				ID	CBID	0		PST ADDRESS				BUFFER ADDRESS				RETURN ADDRESS				0											

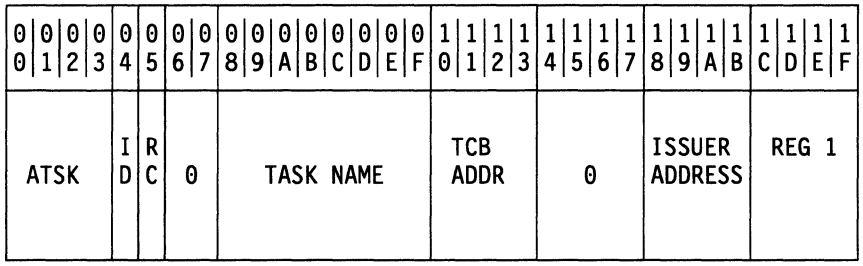
Byte (hex)	Contents
00–03	Record ID: C'AREL'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM
05	Control block ID index value
06–07	Zero
08–0B	PST address
0C–0F	Address of buffer being released
10–13	Address of the caller of the abend RELSTORE routine
14–1F	Zero.

ATSK Entry for Attach of a Subtask

Entry:	**ATSK**
VIT Option:	PSS
Event:	ATTACH of a subtask
VIT Processing Module:	ISTRACPS
Control is returned to:	The module that issued the ATTACH
Applies to:	MVS

This trace record is written when a VTAM module issues an ATTACH macro to start another subtask. This record is generated after the ATTACH completes to ensure that completion information is included in the entry. Consequently, this entry can sometimes appear after the BTSK entry that is generated by the subtask it is attaching.

Not all subtask events generate this entry.



00–03	04	05	06–07	08–0F	10–13	14–17	18–1B	1C–1F
ATSK	ID	RC	0	TASK NAME	TCB ADDR	0	ISSUER ADDRESS	REG 1

Byte (hex)	Contents
00–03	Record ID: C'ATSK'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	Return code from the ATTACH macro
06–07	Zero
08–0F	Name of the subtask being attached
10–13	TCB address returned from the ATTACH macro
14–17	Zero
18–1B	Address of the issuer of the ATTACH macro
1C–1F	Register 1.

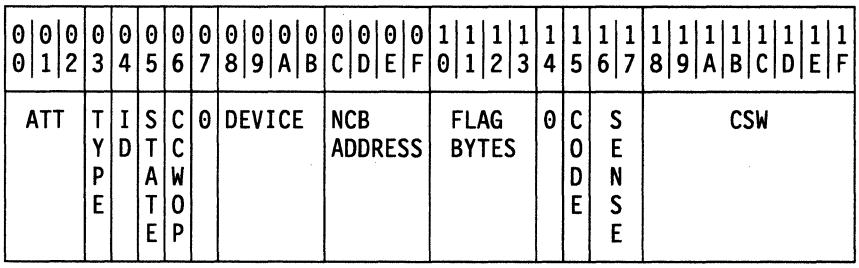
ATT Entry for Attention

Entry:	**ATT**
VIT Option:	CIO
Event:	Attention
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCCA or ISTTSCLA
Applies to:	MVS VM VSE

ATTI is written when an attention interrupt occurs for a channel-attached communication controller or an SNA cluster controller.

ATTL is written when an attention interrupt occurs for a channel-attached non-SNA device (device defined in a local non-SNA major node).

ATTX is written when an attention interrupt occurs for a channel-to-channel attachment, including an IBM 3172 Interconnect Controller.



Byte (hex)	Contents
00–02	Record ID: C'ATT'
03	NCB type: I for ICNCB, L for LDNCB, X for XCNCB
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Link station state (NCBLNKST) for ICNCB and LDNCB; station state (XCNSSFSM) for XCNCB
06	CCW opcode or channel
07	Zero
08–0B	Channel device name in EBCDIC (either a device address or device number)
0C–0F	NCB address
10–13	Flag bytes (NCBFLAGS)
14	Zero
15	I/O completion code (IOSCOD in the IOSB)
16–17	Sense data (IOSSNS of the IOSB)
18–1F	Channel status word from the IOSB.

AXIT Entry for Asynchronous Exit

Entry: **AXIT**
VIT Option: PSS
Event: Asynchronous exit
VIT Processing Module: ISTRACPS
Control is Returned to: ISTAPCSX
Applies to: VM VSE

This trace record is written whenever PSS exits to the operating system because there is no more work to do under a given task.

00	01	02	03	04	05	06 07	08 09 0A 0B	0C	0D	0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
AXIT				ID	0	PST FLAGS	PST ADDRESS	FLAG OR 0	FLAG OR 0	0	SYNC TPPOSTD QUEUE	SYNC NORMAL QUEUE	ASYNC TPPOSTD QUEUE	ASYNC NORMAL QUEUE

Byte (hex)	Contents
00–03	Record ID: C'AXIT'
04	VM Machine ID and Task ID VSE Task ID
05	Zero
06–07	PST Flags (PSTFLG1)
08–0B	PST address
0C	VM Zero VSE AOT Flag A
0D	VM Zero VSE AOT Flag B
0E–0F	Zero
10–13	Synchronous TPPOSTed queue (PSTSTPP)
14–17	Synchronous normal queue (PSTSNRM)
18–1B	Asynchronous TPPOSTed queue (PSTATPP)
1C–1F	Asynchronous normal queue (PSTANRM).

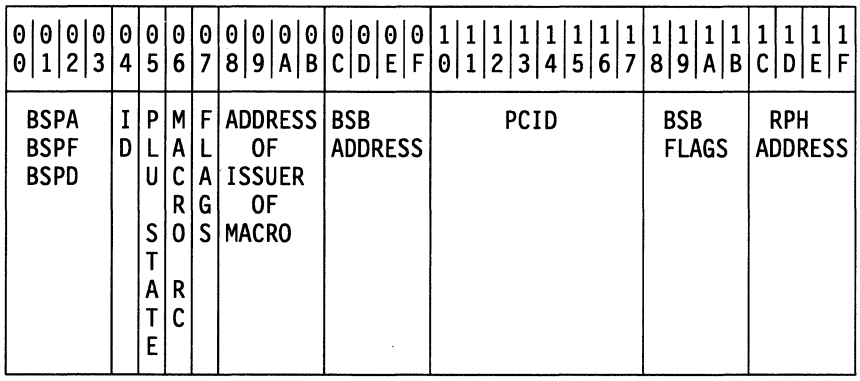
BSPx Entry for ADD, DELETE, and FIND Macros

Entry:	**BSPA, BSPD, or BSPF**
VIT Option:	NRM
Event:	BSBPCID ADD, DELETE, and FIND macros
VIT Processing Module:	ISTRACNR
Control is Returned to:	ISTTSCPU
Applies to:	MVS

This trace record is written when ADD, DELETE, and FIND operations are performed for the BSBPCID tree. It shows:

- Key used for the invocation
- Information from the boundary session block (BSB) indicating the state of the session.

Use this entry to track activity on a session.



Byte (hex)	Contents
00–03	Record ID: C'BSPA' for BSBPCID ADD C'BSPF' for BSBPCID FIND C'BSPD' for BSBPCID DELETE
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	Primary logical unit (PLU) state
06	Macro return code Code Meaning X'00' Successful invocation X'04' Boundary session block (BSB) not found X'08' Duplicate input address X'12' Address not valid
07	IP flags
08–0B	Address of the issuer of the macro
0C–0F	Boundary session block (BSB) address
10–17	Procedure-correlation identifier (PCID)
18–1B	Boundary session block (BSB) flags (FLAG1, FLAG2, FLAG3)
1C–1F	Address of request parameter header (RPH) of module issuing the call.

BSSx Entry for Add, Delete, and Find

Entry: **BSSA, BSSD, or BSSF**
VIT Option: NRM
Event: BSBSA ADD, DELETE, and FIND macros
VIT Processing Module: ISTRACNR
Control is Returned to: ISTTSCB2
Applies to: MVS VM VSE

This trace record contains information about BSBSA ADD, DELETE, or FIND macros issued by a VTAM module.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
BSSA BSSD BSSF				ID	0			RETURN ADDRESS				SUBAREA ADDRESS				SA ELEMENT		ADDRESS REX ELEMENT		BSB ADDRESS OR 0				BSB FLAGS				PLU STATE	RETURN CODE		REX ELEMENT
																															INDEX

Byte (hex) Contents

00–03 Record ID:
- C'BSSA' for BSBSA ADD
- C'BSSD' for BSBSA DELETE
- C'BSSF' for BSBSA FIND

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05–07 Zero

08–0B Address of the issuer of the macro

0C–0F Subarea address

10–11 Subarea element address

12–13 Rex element address

14–17 Boundary session block (BSB) address or zero

18–1B BSB Flags (FLAG1, FLAG2, FLAG3)

1C PLU state

1D Return code

Note: It is not required that users of the product know the meaning of this internal VTAM return code. When required, the product support organization may use it to assist in internal flow diagnosis.

1E–1F Rex element address index.

BSXx Entry for Add, Delete, and Find

Entry:	**BSXA, BSXD, or BSXF**
VIT Option:	NRM
Event:	BSBREX ADD, DELETE, and FIND macros
VIT Processing Module:	ISTRACNR
Control is Returned to:	ISTTSCB1
Applies to:	MVS VM VSE

This trace record contains information about BSBREX ADD, DELETE, or FIND macros issued by a VTAM module.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
BSXA BSXD BSXF				ID	0			RETURN ADDRESS				LFSID		PU ADDR E		CRAFL	0			BSB ADDRESS OR 0				FLAGS				STATE	RTNCD	0	

Byte (hex) Contents

00–03 Record ID:
- C'BSXA' for BSBREX ADD
- C'BSXD' for BSBREX DELETE
- C'BSXF' for BSBREX FIND

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05–07 Zero

08–0B Address of the issuer of the macro

0C–0D LFSID

0E–0F PU element address

10 CRA flags

11–13 Zero

14–17 Boundary session block (BSB) address or zero

18–1B BSB Flags (FLAG1, FLAG2, FLAG3)

1C PLU state

1D Return code

Note: It is not required that users of the product know the meaning of this internal VTAM return code. When required, the product support organization may use it to assist in internal flow diagnosis.

1E–1F Zero.

BTSK Entry for Begin a Subtask

Entry:	**BTSK**
VIT Option:	PSS
Event:	Begin a subtask
VIT Processing Module:	ISTRACPS
Control is returned to:	The subtask that is just starting
Applies to:	MVS

This trace record is written when a VTAM subtask begins execution. This record can appear in the internal trace table before the ATSK entry for the ATTACH of the subtask.

Some subtasks start without generating this entry.

00	01	02	03	04	05	06	07	08 09 0A 0B 0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
BTSK				ID	0			TASK NAME	TCB ADDR	ENTRY POINT ADDR	0	REG 1

Byte (hex)	Contents
00–03	Record ID: 'BTSK'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05–07	Zero
08–0F	Name of the subtask that was just entered
10–13	TCB address of the current subtask
14–17	Entry point into the subtask
18–1B	Zero
1C–1F	Contents of register 1 upon entry.

CCI or CCO Entry for SSCP (RUPE--Part 1)

Entry:	**CCI or CCO**
VIT Option:	SSCP
Event:	Requests/responses with a RUPE (Part 1)
VIT Processing Module:	ISTRACSC
Control is Returned to:	ISTINCCI for inbound processing ISTINCCO for outbound processing
Applies to:	MVS VM VSE

This trace record provides information about outbound processing done by ISTINCCO and inbound processing done by ISTINCCI. ISTINCCO processes a request; ISTINCCI sends a response to the requester.

If this entry is associated with an event failure (that is, the sense data is non-zero), this entry is generated whether the SSCP option is in effect or not. It is treated as an exception condition and, therefore, is traced whenever the VIT is active.

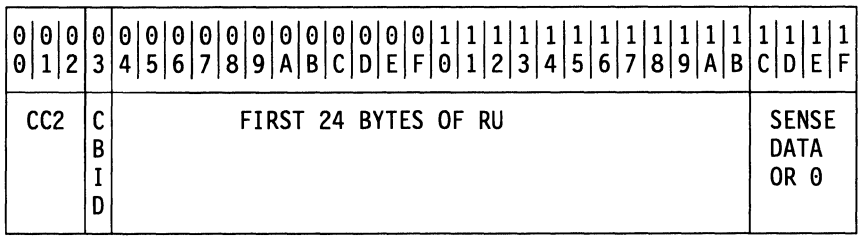
00	01	02	03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17 18 19	1A 1B 1C 1D 1E 1F
CCI OR CCO	FLAGS	ID	CBID	CPCBFL	CPCBRC	SAVE AREA ADDRESS		SAVE AREA ID	RETURN ADDRESS	ORIGIN NETWORK ADDRESS	DESTINATION NETWORK ADDRESS	

Byte (hex)	Contents
00–02	Record ID: • C'CCI' for inbound processing • C'CCO' for outbound processing
03	Flags • 80 = response RU • 40 = sense traced • 00 = request RU
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Control block ID=X'60'
06	Flags (CPCBFL)
07	Return code (CPCBRC)
08–0B	Save area address
0C–0F	Save area ID (bytes 4, 5, 7, and 8 of the module name if available)
10–13	Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)
14–19	For sender, network address at origin (RUPE)
1A–1F	For sender, network address at destination (RUPE).

CC2 Entry for SSCP (RUPE--Part 2)

Entry: **CC2**
VIT Option: SSCP
Event: Requests/responses with a RUPE (Part 2)
VIT Processing Module: ISTRACSC
Applies to: MVS VM VSE

This trace record is a continuation of the CCI or CCO (RUPE) entry.



Byte (hex)	Contents
00–02	Record ID: C'CC2'
03	Control block ID=X'54'
04–1B	First 24 bytes of RU
1C–1F	Sense data or zero.

CCI or CCO Entry for SSCP (NCSPL--Part 1)

Entry:	**CCI or CCO**
VIT Option:	SSCP
Event:	Requests/responses for a NCSPL (Part 1)
VIT Processing Module:	ISTRACSC
Control is Returned to:	ISTINCCI for inbound processing ISTINCCO for outbound processing
Applies to:	MVS VM VSE

This trace record provides information about outbound processing, which processes a request, and inbound processing, which sends a response to the requester.

If this entry is associated with an event failure (that is, the sense data is non-zero), this entry is generated whether the SSCP option is in effect or not. It is treated as an exception condition and, therefore, is traced whenever the VIT is active.

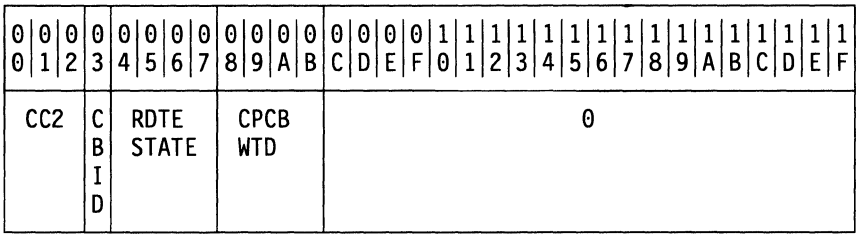
00	01	02	03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18	19	1A 1B 1C 1D 1E 1F
CCI OR CCO			FLAGS	ID	CBID	FLAGS	RTNCD	SAVE AREA ADDRESS	SAVE AREA ID	RETURN ADDRESS	CPCB OPCODE	TYPE	0	DESTINATION NETWORK ADDRESS

Byte (hex)	Contents
00–02	Record ID: • C'CCI' for inbound processing • C'CCO' for outbound processing
03	Flags • 80 = response RU • 40 = sense traced • 00 = request RU
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Control block ID=X'60'
06	Flags (CPCBFL)
07	Return code (CPCBRC)
08–0B	Save area address
0C–0F	Save area ID (bytes 4, 5, 7, and 8 of the module name if available)
10–13	Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)
14–17	CPCB operation code (see Appendix G, "Control Point/Control Block (CPCB) Operation Codes" on page 1127)
18	WTD type
19	Zero
1A–1F	Network address at destination.

CC2 Entry for SSCP (NCSPL--Part 2)

Entry: **CC2**
VIT Option: SSCP
Event: Requests/responses for a NCSPL (Part 2)
VIT Processing Module: ISTRACSC
Applies to: MVS VM VSE

This trace record is a continuation of the CCI or CCO (NCSPL) entry.



Byte (hex)	Contents
00–02	Record ID: C'CC2'
03	Control block ID=X'60'
04–07	RDTE state (in control block RPRE)
08–0B	CPCB WTD (first four bytes)
0C–1F	Zero.

CCI or CCO Entry for SSCP (not RUPE or NCSPL)

Entry:	**CCI or CCO**
VIT Option:	SSCP
Event:	Requests/responses for neither RUPE nor NCSPL
VIT Processing Module:	ISTRACSC
Control is Returned to:	ISTINCCI for inbound processing ISTINCCO for outbound processing
Applies to:	MVS VM VSE

This trace record provides information about outbound processing, which processes a request, and inbound processing, which sends a response to the requester.

If this entry is associated with an event failure (that is, the sense data is non-zero), this entry is generated whether the SSCP option is in effect or not. It is treated as an exception condition and, therefore, is traced whenever the VIT is active.

00	01	02	03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
CCI OR CCO			FLAGS	ID	CBID	CPCBF	RTNCD	SAVE AREA ADDRESS	SAVE AREA ID	RETURN ADDRESS	CPCB OPCODE	CPCB WORD 3	CPCB WORD 4

Byte (hex)	Contents
00–02	Record ID: • C'CCI' for inbound processing • C'CCO' for outbound processing
03	Flags • 80 = response RU • 40 = sense traced • 00 = request RU
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Control block ID (Control block IDs are shown in Appendix H, "Storage and Control Block ID Codes" on page 1137.)
06	CPCB flag (CPCBFL)
07	Return code (CPCBRC)
08–0B	Save area address
0C–0F	Save area ID (bytes 4, 5, 7, and 8 of the module name if available)
10–13	Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)
14–17	CPCB operation code (see Appendix G, "Control Point/Control Block (CPCB) Operation Codes" on page 1127)
18–1B	CPCB word 3 (field WTDPTR)
1C–1F	CPCB word 4 (field CPCBPH).

CFCE Entry for Coupling Facility Request Completion Exit

Entry: **CFCE**
VIT Option: CFS
Event: Complete Exit Driven for a Connection to Coupling Facility Structure
VIT Processing Module: ISTRACCF
Control is Returned to: ISTFSXCM
Applies to: MVS

This trace record is written when a previous request to access the coupling facility structure has completed asynchronously.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CFCE				ID	FLAGS	0	CONNECT ID	CFCON ADDRESS				REQUEST DATA								MVS RETURN CODE				MVS REASON CODE				RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'CFCE'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	Event flags:

Bit	Meaning
1...	This event applies to the new version of the structure.
.1..	This event pertains to an invocation of the IXLLIST macro.
..1.	This event pertains to an invocation of the IXLCACHE macro.
...1	This event pertains to an invocation of the IXLLOCK macro.
.... xxxx	Zero

06	Zero
07	The connection identifier for the target of this event
08–0B	Connection control block (ISTCFCON) address
0C–13	The request data specified on the MVS macro invocation identified in the flags at offset X'05' of CFCE
14–17	Return code for the MVS macro indicated at offset X'05'
18–1B	Reason code for the MVS macro indicated at offset X'05'
1C–1F	Request parameter header (RPH) Address.

CFCN Entry for Coupling Facility Connection Service (Part 1)

Entry: **CFCN**
VIT Option: CFS
Event: Invocation of the MVS macro IXLCONN
VIT Processing Module: ISTRACCF
Control is Returned to: ISTFSGCN
Applies to: MVS

This trace record is written when there is an attempted connection to the coupling facility structure with the IXLCONN macro.

00 01 02 03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
CFCN	ID	FLAGS	0	ID OR 0	CFCON ADDRESS	0	INVOKER ADDRESS	MVS RETURN CODE	MVS REASON CODE	RPH ADDRESS

Byte (hex) Contents

00–03 Record ID: C'CFCN'

04 ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.

05 Event flags:

Bit	Meaning
1...	The rebuild process is in progress for the structure.
.1..	The rebuild stop process is in progress for the structure.
..1.	This connection attempt caused the structure to be allocated.
...1	A previously failed connection has been re-established.
.... 1...	Full connectivity to the rebuild structure cannot be obtained. This bit is valid only if this connection attempt caused the structure to be allocated. This bit will be set only if IXLCONN was invoked during rebuild processing.
... .1..	The structure involved is volatile.
... ..1.	VTAM did not process this event.
... ...x	Zero

06 Zero

07 Connection identifier or 0. If IXLCONN was successful, this field will contain the connection identifier assigned to this connection. Otherwise this field will contain the value zero.

08–0B Connection control block (ISTCFCON) address

0C–0F Zero

10–13 The address of the invoker of IXLCONN

14–17 Return code for the MVS macro IXLCONN

18–1B Reason code for the MVS macro IXLCONN

1C–1F Request parameter header (RPH) Address.

CFC2 Entry for Coupling Facility Connection Service (Part 2)

Entry: **CFC2**
VIT Option: CFS
Event: Invocation of the MVS macro IXLCONN
VIT Processing Module: ISTRACCF
Applies to: MVS

This trace record is a continuation of the CFCN entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CFC2				ACTUAL SIZE or 0				MAXIMUM SIZE				0																			

Byte (hex)	Contents
00–03	Record ID: C'CFC2'
04–07	The actual size of the structure allocated, or zero. If IXLCONN was successful, this field contains the actual number of 4K blocks allocated to the structure.
08–0B	The maximum structure size. This field contains the structure size, in 4K blocks, defined in the active coupling facility resource management policy at the time this structure was allocated.
0C–1F	Zero.

CFDS Entry for Coupling Facility Disconnection Service (Part 1)

Entry: **CFDS**
VIT Option: CFS
Event: Invocation of the MVS macro IXLDISC
VIT Processing Module: ISTRACCF
Control is Returned to: ISTFSUDC
Applies to: MVS

This trace record is written when a connector disconnects from the coupling facility structure with the IXLDISC macro.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CFDS				ID	FLAGS	0	CONNECT ID	CFCON ADDRESS				0				RETURN ADDRESS				RETURN CODE				REASON CODE				RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'CFDS'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	Macro parameter flags Bit Meaning 1... .... REASON(FAILURE) was specified for the request .xxx xxxx Zero
06	Zero
07	Connection identifier of the invoker of IXLDISC
08–0B	Connection control block (ISTCFCON) address
0C–0F	Zero
10–13	The address of the invoker of IXLDISC
14–17	Return code for the MVS macro IXLDISC
18–1B	Reason code for the MVS macro IXLDISC
1C–1F	Request parameter header (RPH) Address.

CFD2 entry for Coupling Facility Disconnection Service (Part 2)

Entry: **CFD2**
VIT Option: CFS
Event: Invocation of the MVS macro IXLDISC
VIT Processing Module: ISTRACCF
Applies to: MVS

This trace record is a continuation of the CFDS entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CFD2				CONNECTOR SPECIFIC DISCONNECT DATA								0																			

Byte (hex) Contents

00–03 Record ID: C'CFD2'
04–0B Connector Specific disconnect data
0C–1f Zero.

CFEE Entry for Coupling Facility Connection Event Exit

Entry:	**CFEE**
VIT Option:	CFS
Event:	Event Exit Driven for a Connection to a Coupling Facility Structure
VIT Processing Module:	ISTRACCF
Control is Returned to:	ISTFSXEV
Applies to:	MVS

This trace record is written when any of the following actions occurs:

- A new connection is established to the coupling facility structure.
- A connection to the coupling facility structure is terminated.
- A connector loses read/write access to the coupling facility structure.
- A state change occurs for the coupling facility structure.
- A user sync point is set by a connector to the coupling facility structure.
- A user sync point has been confirmed by all connectors to the coupling facility structure.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CFEE				ID	FLAGS	EVENT	CONNECT ID	CFCON ADDRESS				EVENT SEQ NUMBER				SUBJECT	STATE	STARTRSN	STOPRSN	USER DATA 1				USER DATA 2				RPH ADDRESS			

Byte (hex) Contents

00–03 Record ID: C'CFEE'

04 ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.

05 The event identifier in byte 06 determines the meaning of this field.

Event flags:

- If byte 06 indicates a loss of connectivity event, use the following flags:

Bit	Meaning
1...	The rebuild process is in progress for the structure.
.1..	The rebuild stop process is in progress for the structure.
..1.	This event applies to the rebuild version of the structure.
...1	No action will be taken. MVS will determine action based on policy information.
.... xxxx	Zero

- If byte 06 indicates an existing connection event, use the following flags:

Bit	Meaning
1...	The rebuild process is in progress for the structure.
.1..	The rebuild stop process is in progress for the structure.
..1.	This event does not pertain to a real connection but indicates that all existing connection events have been received.
...1	The connection identified is active.
.... 1...	The connection identified does not have access to the structure.
.... .xxx	Zero

- If byte 06 indicates a disconnected or failed event, use the following flags:

Bit	Meaning
1...	The rebuild process is in progress for the structure.
.1..	The rebuild stop process is in progress for the structure.
..1.	This event applies to the rebuild version of the structure.
...1	The connection disconnected abnormally.
.... xxxx	Zero

- If byte 06 indicates a rebuild existing connection event, use the following flags:

Bit	Meaning
1...	The rebuild process is in progress for the structure.
.1..	The rebuild stop process is in progress for the structure.
..1.	This event does not pertain to a real connection but indicates that all rebuild existing connection events have been received.
...x xxxx	Zero

- If byte 06 indicates a recommended action, use the following flags:

Bit	Meaning
1...	The rebuild process is in progress for the structure.
.1..	The rebuild stop process is in progress for the structure.
..1.	Policy is available to determine action.
...1	Action is disconnect.
.... xxxx	Zero

- For all other events, use the following flags:

Bit	Meaning
1...	The rebuild process is in progress for the structure.
.1..	The rebuild stop process is in progress for the structure.
..1.	VTAM ignored this event.
...x xxxx	Zero

06 Event identifier (See the mapping of ISTXEEPL in *VTAM Data Areas for MVS* for an explanation of this code.)

07 The connection identifier for the target of this event

08–0B	Connection control block (ISTCFCON) address
0C–0F	Event sequence number. Each event is assigned a unique sequence number.
10	The connection identifier for the subject of this event
11	Connection FSM State. Connection states are listed in the VTAM data map ISTCFCON.
12	Reason for starting Rebuild (See the mapping of ISTXEEPL in *VTAM Data Areas for MVS* for an explanation of this code.)
13	Reason for stopping Rebuild (See the mapping of ISTXEEPL in *VTAM Data Areas for MVS* for an explanation of this code.)
14–17	User Data 1 The contents of this field depend upon the event received and the state of the structure when the event was received. If this event pertains to a user sync point, this field contains the connector-defined event for the user sync point set if a user sync point has been set. If this event pertains to a disconnect or failed connection event, then this field will contain the first 4 bytes of the connector-defined data specified in IXLDISC. If this event pertains to a rebuild event, then this field will contain the connector-defined reason for starting rebuild, if a connector-specified reason was given. Otherwise, this field will contain all zeros.
18–1B	User Data 2 The contents of this field depend upon the event received and the state of the structure when the event was received. If this event pertains to a user sync point, this field contains the connector-defined event for the user sync point confirmed if a user sync point has been confirmed. If this event pertains to a disconnect or failed connection event, then this field will contain the last 4 bytes of the connector-defined data specified in IXLDISC. If this event pertains to a rebuild stop event, then this field will contain the connector-defined reason for stopping the rebuild, if a connector-specified reason was given. Otherwise, this field will contain all zeros.
1C–1F	Request parameter header (RPH) Address.

CFER Entry for Coupling Facility Connection Event Exit Response Service

Entry: **CFER**
VIT Option: CFS
Event: Invocation of the MVS macro IXLEERSP
VIT Processing Module: ISTRACCF
Control is Returned to: ISTFSURS
Applies to: MVS

This trace record is written when a response is given to an event exit event using the IXLEERSP macro.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CFER				ID	SUBJECT	EVENT	CONNECT ID	CFCON ADDRESS				EVENT SEQ NUMBER				RETURN ADDRESS				RETURN CODE				REASON CODE				RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'CFER'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	The connection identifier of the subject of the original event. This field applies to Disconnected/Failed Events, Existing Connection Events, and Rebuild Connect Failure Events only.
06	Event code (See the mapping of ISTXEEPL in *VTAM Data Areas for MVS* for an explanation of this code.)
07	Connection identifier of the invoker of IXLEERSP
08–0B	Connection control block (ISTCFCON) address
0C–0F	The event sequence number of the event to which a response is being given. This field applies to Disconnected/Failed Events, Existing Connection Events, and Rebuild Connect Failure Events only. Otherwise, this field contains a zero.
10–13	The address of the invoker of IXLEERSP
14–17	Return code for the MVS macro IXLEERSP
18–1B	Reason code for the MVS macro IXLEERSP
1C–1F	Request parameter header (RPH) Address.

CFFC Entry for Coupling Facility IXLFORCE Service

Entry: **CFFC**
VIT Option: CFS
Event: Invocation of the MVS macro IXLFORCE
VIT Processing Module: ISTRACCF
Control is Returned to: ISTFSUFC
Applies to: MVS

This trace record is written when a failed-persistent connection is deleted using the IXLFORCE macro.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CFFC				ID	SUBJECT	FUNC	CONNECT ID	CFCON ADDRESS				0				RETURN ADDRESS				RETURN CODE				REASON CODE				RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'CFFC'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	The connection identifier of the failed-persistent connection
06	The function requested by this invocation of IXLFORCE: X'01' REQUEST(CONNECTION) has been specified.
07	Connection identifier of the invoker of IXLFORCE
08–0B	Connection control block (ISTCFCON) address
0C–0F	Zero
10–13	The address of the invoker of IXLFORCE
14–17	Return code for the MVS macro IXLFORCE
18–1B	Reason code for the MVS macro IXLFORCE
1C–1F	Request parameter header (RPH) Address.

CFLS Entry for Coupling Facility IXLLIST Service (Part 1)

Entry: **CFLS**
VIT Option: CFS
Event: Invocation of the MVS macro IXLLIST
VIT Processing Module: ISTRACCF
Control is Returned to: ISTFSKRT, ISTFSUCL, ISTFSUDL, ISTFSUEN, ISTFSUHT, ISTFSUMT, ISTFSURH, ISTFSURL, ISTFSUUL, ISTFSUWT
Applies to: MVS

This trace record is written for a request to access the coupling facility structure using the IXLLIST macro.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CFLS				ID	EVENT FLAGS	FUNC	CONNECT ID	CFCON ADDRESS				MODE	DATAOPER	LOCKOPER	LOCKCOMP	RETURN ADDRESS				RETURN CODE OR PART 1 OF REQUEST DATA				REASON CODE OR PART 2 OF REQUEST DATA				RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'CFLS'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	Event flags: Bit Meaning 1... .... Request will complete asynchronously. .xxx xxxx Zero
06	The function requested by this invocation of IXLLIST: X'01' REQUEST(READ) specified X'02' REQUEST(READ_LCONTROLS) specified X'03' REQUEST(WRITE) specified X'04' REQUEST(MOVE) specified X'05' REQUEST(DELETE) specified X'06' REQUEST(LOCK) specified
07	Connection identifier of the invoker of IXLREBLD
08–0B	Connection control block (ISTCFCON) address
0C	The value specified for the MODE parameter: X'01' MODE(SYNCSUSPEND) X'02' MODE(SYNCEXIT)
0D	The value specified for the DATAOPER parameter, or zero. This field applies to the MOVE and DELETE functions only. Otherwise it is set to zero. X'01' DATAOPER(NONE) specified or assumed by default X'02' DATAOPER(READ) specified

	X'03' DATAOPER(WRITE) specified
0E	The value specified for the LOCKOPER parameter or zero X'01' LOCKOPER(SET) specified X'02' LOCKOPER(RESET) specified
0F	The value specified for the LOCKCOMP parameter or zero. This one byte value represents the connection identifier of the connector that is assumed to own a given lock.
10–13	The address of the invoker of IXLLIST
14–17	Return code for the MVS macro IXLLIST or the first four bytes of request data. If this IXLLIST invocation will complete asynchronously, this field holds the first four bytes of the request data specified for this request. Otherwise, it holds the return code for this invocation of IXLLIST.
18–1B	Reason code for the MVS macro IXLLIST or second four bytes of request data. If this IXLLIST invocation will complete asynchronously, this field holds the first four bytes of the request data specified for this request. Otherwise, it holds the reason code for this invocation of IXLLIST.
1C–1F	Request parameter header (RPH) Address.

CFL2 Entry for Coupling Facility IXLLIST Service (Part 2)

Entry: **CFL2**
VIT Option: CFS
Event: Invocation of the MVS macro IXLLIST
VIT Processing Module: ISTRACCF
Applies to: MVS

This trace record is a continuation of the CFLS entry.

0 0 0 0 0 1 2 3	0 0 0 0 4 5 6 7	0 0 0 0 8 9 A B	0 0 0 0 C D E F	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 2 3 4 5 6 7 8 9 A B C D E F
CFL2	SOURCE LIST NUMBER	0	LOCK INDEX	ENTRYNAME, ENTRYID, OR ENTRYKEY

Byte (hex)	Contents
00–03	Record ID: C'CFL2'
04–07	The list number, if any, from which data is being retrieved. This field applies to move and read requests; it represents the list on which the data being manipulated originates.
08–0B	Zero
0C–0F	The index of the lock, if any, that is being manipulated
10–1F	The value of any ENTRYNAME, ENTRYID, or ENTRYKEY parameter specified.

CFNF Entry for Coupling Facility Event Notification

Entry: **CFNF**
VIT Option: CFS
Event: Event Notification Facility (ENF) Exit Driven for Event X'35'
VIT Processing Module: ISTRACCF
Control is Returned to: ISTFSXEF
Applies to: MVS

This trace record is written when a change in coupling facility resources or in the coupling facility structure has occurred that may now allow a connection to a structure that had previously failed.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CFNF				ID	0	FUNC	0					STRUCTURE NAME																RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'CFNF'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	Zero
06	Exit function codes pertaining to ENF Events with event code X'35': X'01' A coupling facility resource has become available that may allow a structure to successfully connect. X'02' A specific coupling facility structure has become available.
07–0B	Zero
0C–1B	If the exit function code is X'02', this will contain the structure name. Otherwise, it will contain zeros.
1C–1F	Request parameter header (RPH) Address.

CFRB Entry for Coupling Facility Structure Rebuild Service

Entry: **CFRB**
VIT Option: CFS
Event: Invocation of the MVS macro IXLREBLD
VIT Processing Module: ISTRACCF
Control is Returned to: ISTFSURB
Applies to: MVS

This trace record is written when a rebuild of the coupling facility structure is started, stopped, or completed, using the IXLREBLD macro.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CFRB				ID	FLAGS	FUNCTION	CONNECTID	CFCON ADDRESS				CONNECT REASON CODE				INVOKER ADDRESS				MVS RETURN CODE				MVS REASON CODE				RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'CFRB'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	Macro Parameter Flags
	Bit Meaning
	1... .... STARTREASON(LOSSCONN) was specified for the request.
	.1.. .... STARTREASON(STRFAILURE) was specified for the request.
	..1. .... STOPREASON(LOSSCONNOLD) was specified for the request.
	...1 .... STOPREASON(LOSSCONNEW) was specified for the request.
	.... 1... The rebuild was started or stopped for a connection-specific reason. Bytes X'0C'–X'0F' contain the connection-specific reason.
	.... .xxx Zero
06	The function requested by this invocation of IXLREBLD:
	X'01' REQUEST(START) was specified.
	X'02' REQUEST(STOP) was specified.
	X'03' REQUEST(COMPLETE) was specified.
07	Connection identifier of the invoker of IXLREBLD
08–0B	Connection control block (ISTCFCON) address
0C–0F	Connection-specific reason for starting or stopping the rebuild.
10–13	The address of the invoker of IXREBLD
14–17	Return code for the MVS macro IXLREBLD
18–1B	Reason code for the MVS macro IXLREBLD

1C–1F Request parameter header (RPH) Address.

CFUS Entry for Coupling Facility User Sync Point Service

Entry:	**CFUS**
VIT Option:	CFS
Event:	Invocation of the MVS macro IXLUSYNC
VIT Processing Module:	ISTRACCF
Control is Returned to:	ISTFSUUS
Applies to:	MVS

This trace record is written when a connector sets or confirms a user sync point for a coupling facility structure with the IXLUSYNC macro.

00	01	02	03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
CFUS				ID	0	FUNC	CONNECT ID	CFCON ADDRESS	USER EVENT	RETURN ADDRESS	RETURN CODE	REASON CODE	RPH ADDRESS

Byte (hex)	Contents
00–03	Record ID: C'CFUS'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	Zero
06	Function requested by this invocation of IXLUSYNC: X'01' SET function requested X'02' CONFIRM function requested X'03' CONFIRMSET function requested
07	Connection identifier of the invoker of IXLUSYNC
08–0B	Connection control block (ISTCFCON) address
0C–0F	The value of USEREVENT parameter for this invocation of IXLUSYNC
10–13	The address of the invoker of IXLUSYNC
14–17	Return code for the MVS macro IXLUSYNC
18–1B	Reason code for the MVS macro IXLUSYNC
1C–1F	Request parameter header (RPH) Address.

CI1 or CO1 Trace Entries

This trace record provides information about the inbound and outbound requests sent to session services to begin or end a same-network or cross-network LU-LU session.

The CI1 and CO1 traces have two formats:

- **Format 0** is used for same-network sessions or when VTAM does not know whether the request is for a cross-network session. This format contains parts 1, 2, and 3 of the CIn or COn record.
- **Format 1** is used for cross-network sessions. This format contains parts 1, 2, 3, and 4 of the CIn or COn record.

CI1 or CO1 Entry for SSCP (RUPE--Part 1)

Entry:	**CI1 or CO1**
VIT Option:	SSCP
Event:	Requests (Part 1)
VIT Processing Module:	ISTRACSC
Control is Returned to:	Module that issued the INTRACE macro
Applies to:	MVS VM VSE

Part 1 of the CIn or COn record is for both format 0 and format 1 and contains essentially the same information as the CCI or CCO entry.

If this entry is associated with an event failure (that is, the sense data is non-zero), this entry is generated whether the SSCP option is in effect or not. It is treated as an exception condition and, therefore, is traced whenever the VIT is active.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CI1 OR CO1			FLAGS	ID	CBID	ITFSM		SAVE AREA ADDRESS				SAVE AREA ID				RETURN ADDRESS				ORIGIN NETWORK ADDRESS						DESTINATION NETWORK ADDRESS					

Byte (hex) Contents

00–02 Record ID:
- C'CI1' for inbound processing
- C'CO1' for outbound processing

03 Flags

Bit	Meaning
0...	Requests
1...	Response
.0..	No sense
.1..	Sense
.... ..00	Format 0
.... ..01	Format 1

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Control block ID=X'54'

06–07 Initiation and termination finite state machines (taken from the session information block, SIB)

08–0B Save area address

0C–0F Save area ID (bytes 4, 5, 7, and 8 of the module name if available)

10–13 Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)

14–19 Network address at origin (RUPEOAF)

1A–1F Network address at destination (RUPEDAF).

CI1 or CO1 Entry for SSCP (NCSPL--Part 1)

Entry: **CI1 or CO1**
VIT Option: SSCP
Event: Requests (Part 1)
VIT Processing Module: ISTRACSC
Control is Returned to: Module that issued the INTRACE macro
Applies to: MVS VM VSE

00	01	02	03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18	19	1A 1B 1C 1D 1E 1F
CI1 OR CO1			FLAGS	ID	CBID	FLAGS	RTNCD	SAVE AREA ADDRESS	SAVE AREA ID	RETURN ADDRESS	CPCB OPCODE	TYPE	0	DESTINATION NETWORK ADDRESS

Byte (hex) Contents

00–02 Record ID:
- C'CI1' for inbound processing
- C'CO1' for outbound processing

03 Flags

Bit	Meaning
0...	Requests
1...	Response
.0..	No sense
.1..	Sense
.... ..00	Format 0
.... ..01	Format 1

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Control block ID=X'60'

06 Flags (CPCBFL)

07 CPCB return code (CPCBRC)

08–0B Save area address

0C–0F Save area ID (bytes 4, 5, 7, and 8 of the module name if available)

10–13 Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)

14–17 CPCB operation code (see Appendix G, "Control Point/Control Block (CPCB) Operation Codes" on page 1127)

18 WTD type

19 Zero

1A–1F Network address at destination (NCSPLDAF).

CI1 or CO1 Entry for SSCP (not RUPE or NCSPL--Part 1)

Entry: **CI1 or CO1**
VIT Option: SSCP
Event: Requests (Part 1)
VIT Processing Module: ISTRACSC
Control is Returned to: Module that issued the INTRACE macro
Applies to: MVS VM VSE

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CI1 OR CO1			FLAGS	ID		CBID	CPCBFC	RTNCD	SAVE AREA ADDRESS				SAVE AREA ID			RETURN ADDRESS				CPCB OPCODE				CPCB WORD 3				CPCB WORD 4		

Byte (hex)	Contents
00–02	Record ID: • C'CI1' for inbound processing • C'CO1' for outbound processing
03	Flags Bit Meaning 0... .... Requests 1... .... Response .0.. .... No sense .1.. .... Sense .... ..00 Format 0 .... ..01 Format 1
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Control block ID (see Appendix H, "Storage and Control Block ID Codes" on page 1137)
06	CPCB flag (CPCBFL)
07	Return code (CPCBRC)
08–0B	Save area address
0C–0F	Save area ID (bytes 4, 5, 7, and 8 of the module name if available)
10–13	Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)
14–17	CPCB operation code (see Appendix G, "Control Point/Control Block (CPCB) Operation Codes" on page 1127)
18–1B	CPCB word 3 (field WTDPTR)
1C–1F	CPCB word 4 (field CPCBPH).

CI2 or CO2 Entry for SSCP (RUPE--Part 2)

Entry:	**CI2 or CO2**
VIT Option:	SSCP
Event:	Requests (Part 2)
VIT Processing Module:	ISTRACSC
Applies to:	MVS VM VSE

The CI2 and CO2 trace records are continuations of the CI1 and CO1 entries.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CI2 OR CO2			CB ID	FIRST 24 BYTES OF RU																								SENSE DATA OR 0			

Byte (hex)	Contents
00–02	Record ID: • C'CI2' for inbound processing • C'CO2' for outbound processing
03	Control block ID=X'54'
04–1B	First 24 bytes of RU
1C–1F	Sense data or zero.

CI2 or CO2 Entry for SSCP (NCSPL--Part 2)

Entry: **CI2 or CO2**
VIT Option: SSCP
Event: Requests (Part 2)
VIT Processing Module: ISTRACSC
Applies to: MVS VM VSE

This trace record is a continuation of the CI1 or CO1 trace record.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CI2 OR CO2			CB ID	RDTE STATE				CPCB WTD				0																			

Byte (hex) Contents

00–02 Record ID:
- C'CI2' for inbound processing
- C'CO2' for outbound processing

03 Control block ID=X'60'

04–07 RDTE state (in control block RPRE). This value, a resource status code, is explained in "Resource Status Codes and Modifiers" in *VTAM Messages and Codes* and in control block FSM. See *VTAM Data Areas for MVS/ESA*, *VTAM Data Areas for VM/ESA*, or *VTAM Data Areas for VSE/ESA*.

08–0B CPCB WTD (first four bytes)

0C–1F Zero.

CI2 or CO2 Entry for SSCP (not RUPE or NCSPL--Part 2)

Entry: **CI2 or CO2**
VIT Option: SSCP
Event: Requests (Part 2)
VIT Processing Module: ISTRACSC
Applies to: MVS VM VSE

This trace record is a continuation of the CI1 or CO1 trace record.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CI2 OR CO2			CB ID	0																											

Byte (hex)	Contents
00–02	Record ID: • C'CI2' for inbound processing • C'CO2' for outbound processing
03	Control block ID (see Appendix H, "Storage and Control Block ID Codes" on page 1137)
04–1F	Zero.

CI3 or CO3 Entry for SSCP (RUPE, NCSPL, or Not RUPE or NCSPL--Part 3)

Entry:	**CI3 or CO3**
VIT Option:	SSCP
Event:	Requests (Part 3)
VIT Processing Module:	ISTRACSC
Applies to:	MVS VM VSE

Part 3 of the CIn or COn record is for both format 0 and format 1 and contains the address of the ISTSIB for the session, the ISTPCID, and the PLU and SLU names.

Note: If this is the last CIn or COn entry, these are the PLU and SLU names as known in this network. If this entry is followed by CI4 or CO4, these are just the PLU and SLU names. The CI4 or CO4 entry identifies the network.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CI3 OR CO3			0	SIB ADDRESS				PCID FOR THIS REQUEST								PLU NAME								SLU NAME							

Byte (hex)	Contents
00–02	Record ID: • C'CI3' for inbound processing • C'CO3' for outbound processing
03	Zero
04–07	SIB address
08–0F	PCID for this request
10–17	PLU name
18–1F	SLU name.

CI4 or CO4 Entry for SSCP (RUPE, NCSPL, or Not RUPE or NCSPL--Part 4)

Entry: **CI4 or CO4**
VIT Option: SSCP
Event: Requests (Part 4)
VIT Processing Module: ISTRACSC
Applies to: MVS VM VSE

Part 4 of the CIn or COn entry is for format 1 only.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CI4 OR CO4			0					PCID IN PREVIOUS REQUEST								PLU NETID								SLU NETID							

Byte (hex) Contents

00–02 Record ID:
- C'CI4' for inbound processing
- C'CO4' for outbound processing

03–07 Zero
08–0F PCID in previous request
10–17 PLU network ID
18–1F SLU network ID.

CNA Entry for VCNSCMD Issued by Application

Entry: **CNA**
VIT Option: VCNS
Event: VCNSCMD issued by application
VIT Processing Module: ISTRACNS
Control is Returned to: ISTAICVC
MVS VM

This CNA trace record shows VCNS command requests. The API routine ISTAICVC writes the entry and then queues the RPL to the correct PAB.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CNA				ID	EXTDS	OPT1	CNTRL	RPL ADDRESS				RPL3 ADDRESS				RPL AREA				VCNS FLAGS			0	OPERAND-DEPENDENT DATA							

Byte (hex) Contents

00–03 Record ID: C'CNA'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM

05 Exit definition (RPLEXTDS)

Bit	Meaning
1...	Indicates RPL exit was scheduled
.1..	Indicates RPL exit not specified
..1.	Indicates RPL exit was specified
...x xx..	Zero
.... ..1.	Indicates BRANCH=YES specified
....x	Zero

06 Option code byte 1 (RPLOPT1)

Bit	Meaning
xxxx	Zero
.... 1...	Asynchronous request indicator
.... .xx.	Zero
....1	External ECB indicator

07 VCNSCMD CONTROL= operand value

08	INQUIRE
10	LOGON
14	TEST
18	XID
20	LOGOFF
30	READ VM
40	STATUS
50	SET
60	OPEN VM
70	CLOSE VM
80	SEND

	90	SUSPEND VM
	A0	RESUME VM
	B0	EXPEDITE VM
	C0	CHECK
	D0	RESET VM
	E0	SETCPARM
	F0	RECEIVE
	FC	REPLY

08–0B RPL address

0C–0F RPL3 address (RPLAAREA)

10–13 Address of area into which data is to be read or from which data is to be written (RPLAREA), or zero if no area is supplied

14 VCNS flag byte

Bit	Meaning
1...	Q-bit indicator
.1..	M-bit indicator
..1.	D-bit indicator
...1	Buffer list indicator
.... xxxx	Zero

15 VCNS flag byte

Bit	Meaning
00..	Continue specific
01..	Continue any
11..	Continue same
..00	Data flow=on
..01	Data flow=off
..11	Data flow=same
.... 1...	Receive any indicator
.... .xxx	Zero

16 VCNS flag byte

Bit	Meaning
1...	STYPE=CONFIRM indicator
.xxx xxxx	Zero

17 Zero

18–1F When CONTROL = LOGON or CONTROL = INQUIRE: symbolic name of the network access point

18–1B When CONTROL does not equal LOGON and CONTROL does not equal INQUIRE and the request is an X.25 request: address into which expedited data is to be read or from which expedited data is to be written, or zero if no area is supplied

1C–1F When CONTROL does not equal LOGON and CONTROL does not equal INQUIRE: connection or resource identifiers (RID)

- RID is supplied when the VCNSCMD CONTROL value is:
  - LOGOFF
  - OPEN (STYPE=REQUEST)
  - READ
  - RECEIVE (when SMODE is CONNECTIONLESS)
  - REPLY
  - SEND (when the connection ID value is zero)
  - SET
  - SETCPARM (when the connection ID value is zero)
  - STATUS (when the RID value is not zero)
  - TEST

XID

- Connection identifier is supplied when the VCNSCMD CONTROL value is:
  CLOSE
  EXPEDITE
  OPEN (STYPE=CONFIRM)
  RECEIVE (Specific)
  RESET
  RESUME
  SEND (when the connection ID value is not zero)
  SETCPARM (when the connection ID value is not zero)
  SUSPEND
  STATUS (when the RID value is zero).

CNP1 or CNR1 Entry for ECB Posted or RPL Exit Dispatched (Part 1)

Entry: **CNP1 or CNR1**
VIT Option: VCNS
Event: ECB posted or RPL exit dispatched (Part 1)
VIT Processing Module: ISTRACNS
Control is Returned to: ISTAICPT for CNP1
ISTAPCUE for CNR1
MVS VM

The CNP1 trace record is written when an ECB is posted. The CNR1 trace record is written when an RPL exit is dispatched. These trace records signal that execution of the VCNSCMD macroinstruction is complete and show the data returned to the user application program.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CNP1 OR CNR1				ID	RTNCD	FDB2	CNTRL	RPL ADDRESS				RPL3 ADDRESS				RPL AREA				VCNS FLAGS			0	PARAMETER-DEPENDENT DATA							

Byte (hex) Contents

00–03 Record ID:
- C'CNP1' for ECB posting
- C'CNR1' for RPL exit dispatching

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM

05 Global VTAM return code (RPLRTNCD)

06 Global VTAM feedback code (RPLFDB2)

07 VCNSCMD CONTROL= operand value

08	INQUIRE
10	LOGON
14	TEST
18	XID
20	LOGOFF
30	READ VM
40	STATUS
50	SET
60	OPEN VM
70	CLOSE VM
80	SEND
90	SUSPEND VM
A0	RESUME VM
B0	EXPEDITE VM
C0	CHECK
D0	RESET VM
E0	SETCPARM
F0	RECEIVE

08–0B RPL address

0C–0F RPL3 address (RPLAAREA)

10–13 Address of area containing user data (RPLAREA), or zero if no area is supplied

14 VCNS flag byte

Bit	Meaning
1...	Q-bit indicator
.1..	M-bit indicator
..1.	D-bit indicator
...1	Buffer list indicator
.... xxxx	Zero

15 VCNS flag byte

Bit	Meaning
00..	Continue specific
01..	Continue any
11..	Continue same
..00	Data flow=on
..01	Data flow=off
..11	Data flow=same
.... 1...	Receive any indicator
.... .xxx	Zero

16 VCNS flag byte

Bit	Meaning
1...	STYPE=CONFIRM indicator
.xxx xxxx	Zero

17 Zero

18–1B Address into which expedited data is to be read or from which expedited data is to be written, or zero if no area is supplied

1C–1F Connection or resource identifier (RID)

- RID is returned when the VCNSCMD CONTROL value is:
  - LOGON
  - LOGOFF
  - READ
  - SET
  - STATUS (for network access point)
  - TEST
  - XID
- Connection ID is returned when the VCNSCMD CONTROL value is:
  - CLOSE
  - EXPEDITE
  - OPEN
  - RECEIVE
  - RESET
  - RESUME
  - SEND
  - SETCPARM
  - SUSPEND
  - STATUS (for connection)
- Zero is returned when the VCNSCMD CONTROL value is INQUIRE.

CNP2 or CNR2 Entry for ECB Posted or RPL Exit Dispatched (Part 2)

Entry: **CNP2 or CNR2**
VIT Option: VCNS
Event: ECB posted or RPL exit dispatched (Part 2)
VIT Processing Module: ISTRACNS
Applies to: MVS VM

The CNP2 trace record is a continuation of the CNP1 trace record. The CNR2 trace record is a continuation of the CNR1 trace record.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CNP2 OR CNR2				ECB ADDRESS OR EXIT ADDRESS				R C P R I		R C S E C		LENGTH OF USER DATA AREA				M A X L N		D A T L N		SENSE DATA OR 0				EXPEDITED DATA							

Byte (hex) Contents

00–03 Record ID:
- C'CNP2' for continuation of CNP1
- C'CNR2' for continuation of CNR1

04–07 ECB or RPL exit address (RPLECB)
08–09 VCNSCMD primary return code
0A–0B VCNSCMD secondary return code
0C–0F Length of area containing user data (RPLRLEN)
10–11 Length of expedited data area
12–13 Length of expedited data received
14–17 Four bytes of sense data, or zero when no sense data is returned
18–1F Eight bytes of expedited data.

CNP3 or CNR3 Entry for ECB Posted or RPL Exit Dispatched (Part 3)

Entry: **CNP3 or CNR3**
VIT Option: VCNS
Event: ECB posted or RPL exit dispatched (Part 3)
VIT Processing Module: ISTRACNS
Applies to: MVS VM

The CNP3 trace record is a continuation of the CNP2 trace record. The CNR3 trace record is a continuation of the CNR2 trace record. These records are written only when the VCNSCMD CONTROL operand value is RECEIVE or CLOSE, and RPLRLEN is greater than zero.

0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	0 A	0 B	0 C	0 D	0 E	0 F	1 0	1 1	1 2	1 3	1 4	1 5	1 6	1 7	1 8	1 9	1 A	1 B	1 C	1 D	1 E	1 F
CNP3 OR CNR3				DATA FIELD																											

Byte (hex) Contents

00–03 Record ID:
- C'CNP3' for continuation of CNP2
- C'CNR3' for continuation of CNR2

04–1F Twenty-eight bytes of user data (RPLAREA).

CONN Entry for SHM Connect

Entry:	**CONN**
VIT Option:	CIO
Event:	Connection
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSC1R
Applies to:	VSE

This trace record is generated for X.21 short-hold mode attachments. It is written when a port is connected to a station.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CONN				TIK	CFSM	PMFSM	SHSI	CUA				GROUP NCB ADDRESS				HALCB ADDRESS				PUT ADDRESS				0				PEA		0	

Byte (hex)	Contents
00–03	Record ID: C'CONN'
04	Task identification key
05	Connection finite state machine (FSM; see HALCFSM in ISTHALCB for values)
06	Port manager FSM (see HALPMFSM in ISTHALCB for values)
07	X.21 short-hold mode status indicator (see PUTFLG1 in ISTPUT)
08–0B	Channel unit address of link
0C–0F	Group network control block address (ISTGRPCB)
10–13	Link network control block address (ISTHALCB)
14–17	Physical unit table address (ISTPUT)
18–1B	Zero
1C–1D	PU element address
1E–1F	Zero.

CPPG or CPPT Entry for CPPURGE or CPPOST Event (Part 1)

Entry:	**CPPG or CPPT**
VIT Option:	SSCP
Event:	CPPURGE or CPPOST Macro (Part 1)
VIT Processing Module:	ISTRACSC
Control is Returned to:	ISTPVCWP
Applies to:	MVS VM VSE

This trace record gives information about CPPURGE or CPPOST macroinstruction processing. It may help you determine why I/O is outstanding or why an SSCP request is failing.

CPPG and CPPT are the complements of CPWT. Each CPPT is preceded by a CPWT. But a CPPG can post several events at once, as it may be concluding several CPWT entries.

00 01 02 03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
CPPG OR CPPT	ID	RTNCD	TYPE	OPT	LQAB GROUP ADDRESS	WREDATA IN WRE	WORK ELEMENT ADDRESS	OPC IN WORK ELEMENT OR 0	SAVE AREA ID OR 0	SENSE DATA OR 0

Byte (hex) Contents

00–03 Record ID:
- C'CPPG' for CPPURGE processing
- C'CPPT' for CPPOST processing

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Return code set by CPPOST or CPPURGE

06 Type flags

Bit	Meaning
....x	1 = PVI event
.... ..x.	1 = search on EID 0 = search on URC
.... .x..	For OPTIONS(POST) only, 1 indicates that return of a control block is optional. This bit corresponds to the CTLBLKOP keyword on the CPPOST macro.
.... x...	For OPTIONS(FIND) only, 1 indicates that the invoker intends to change the WRE's event ID. This bit corresponds to the CHGEID keyword on the CPPOST macro.

07 High-order 4 bits: Type of LQAB used for the search

Value	LQAB Type
0000	Global LQAB
0001	EID-related LQAB
0010	DAF-related LQAB
0011	URC-related LQAB

Low-order 4 bits: Function requested through the OPTIONS keyword on the CPPOST macro, zero for CPPG

Value	Keyword
0000	FIND
0001	DEQ
0010	POST

08–0B	LQAB group address
0C–0F	Data field in ISTWRE
10–13	Work element address
14–17	CPCB operation code that indicates the original request for this work element or zero (see Appendix G, "Control Point/Control Block (CPCB) Operation Codes" on page 1127)
18–1B	For PVI events, caller's save area ID or 0 For non-PVI events, save area ID
1C–1F	For POST or PURGE, sense data or zero For DEQ or FIND, zero.

CPP2 Entry for CPPURGE or CPPOST (Part 2)

Entry: **CPP2**
VIT Option: SSCP
Event: CPPURGE or CPPOST macro (Part 2)
VIT Processing Module: ISTRACSC
Applies to: MVS VM VSE

This trace record contains part or all of the RU for CPPURGE or CPPOST macroinstruction processing. It may help you determine why I/O is outstanding or why an SSCP request is failing. If an RU is fewer than 24 bytes long, only the bytes in the actual RU will be moved to the corresponding fields in the trace records. The leftover bytes will be set to zero.

Note: This trace record is produced only for PVI events where a RUPE is available.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPP2				RETURN ADDRESS				FIRST 24 BYTES OF RU OR 0																							

Byte (hex)	Contents
00–03	Record ID: C'CPP2'
04–07	Return Address
08–1F	First 24 bytes of RU or zero.

CPP3 Entry for CPPURGE or CPPOST (Part 3)

Entry: **CPP3**
VIT Option: SSCP
Event: CPPURGE or CPPOST macro (Part 3)
VIT Processing Module: ISTRACSC
Applies to: MVS VM VSE

This trace record is a continuation of the CPPG or CPPT trace record.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPP3				25 BYTES OF URC																									0		

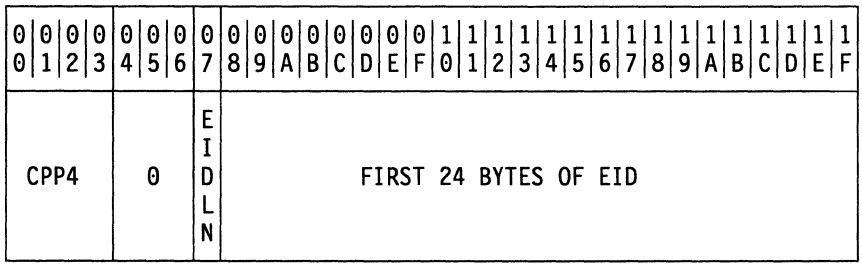
Byte (hex)	Contents
00–03	Record ID: C'CPP3'
04–1C	25 bytes of URC
1D–1F	Zero.

CPP4 Entry for CPPURGE or CPPOST (Part 4)

Entry: **CPP4**
VIT Option: SSCP
Event: CPPURGE or CPPOST macro (Part 3)
VIT Processing Module: ISTRACSC
Applies to: MVS VM VSE

This trace record contains part or all of the event ID (EID) for CPPURGE or CPPOST macroinstruction processing. If an EID is fewer than 24 bytes long, only the bytes in the actual EID will be moved to the corresponding fields in the trace records. The leftover bytes will be set to zero.

Note: This trace record is produced only when the WRE points to an EID.



Byte (hex)	Contents
00–03	Record ID: C'CPP4'
04–06	Zero
07	Length of EID
08–1F	First 24 bytes of EID.

CPRC Entry for CPRC (Part 1)

Entry:	**CPRC**
VIT Option:	SSCP
Event:	CPRC macro (Part 1)
VIT Processing Module:	ISTRACSC
Control is Returned to:	Module invoking the INTRACE macro that caused the record to be produced.
Applies to:	MVS VM VSE

This trace record gives information about CPRC macroinstruction processing. It is written when a VTAM module issues a CPRC macroinstruction to set a non-zero sense code in an RUPE.

This event is treated as an exception condition and is always traced, whether the SSCP option is in effect or not.

00 01 02 03	04	05 06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17 18 19	1A 1B 1C 1D 1E 1F
CPRC	ID	0	RUPE ADDRESS OR 0	CPCB OPCODE IN THIS RUPE	SENSE CODE	ORIGIN NETWORK ADDRESS	DESTINATION NETWORK ADDRESS

Byte (hex)	Contents
00–03	Record ID: C'CPRC'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05–07	Zero
08–0B	Address of RUPE for this macro, or 0
0C–0F	CPCB operation code for RUPE that indicates the original request for this RUPE (see Appendix G, "Control Point/Control Block (CPCB) Operation Codes" on page 1127)
10–13	Sense code
14–19	Network address at origin (RUPE)
1A–1F	Network address at destination (RUPE).

CPR2 Entry for CPRC (Part 2)

Entry: **CPR2**
VIT Option: SSCP
Event: CPRC macro (Part 2)
VIT Processing Module: ISTRACSC
Applies to: MVS VM VSE

This trace record is a continuation of the CPRC entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPR2				RETURN ADDRESS				ISSUER NAME					INSTANCE	RTNCD	0																

Byte (hex) Contents

00–03 Record ID: C'CPR2'

04–07 Address of the issuer of the CPRC macro

08–0C MVS VM Module name field from the register save area for the module that issued the CPRC macro

This field contains the abbreviated name (bytes 4, 5, 6, 7, 8) of the module that issued the CPRC macro. If the issuing module does not have its own save area, for example, modules ISTINCF1 and ISTPUCTI, this field contains the name field from the save area of the caller of the module that issued the CPRC macro.

0D Instance of the CPRC macro in the issuer

0E CPCBRC return code

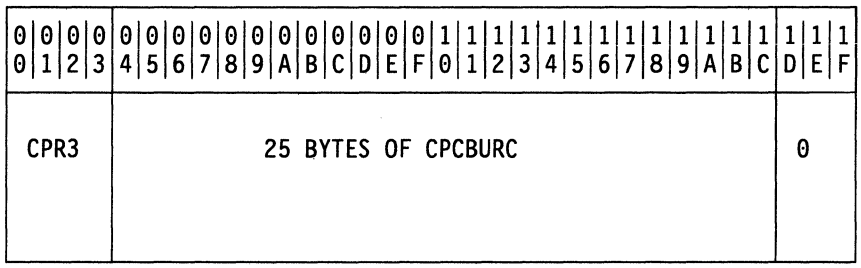
Note: It is not required that users of the product know the meaning of this internal VTAM return code. When required, the product support organization may use it to assist in internal flow diagnosis.

0F–1F Zero.

CPR3 Entry for CPRC (Part 3)

Entry: **CPR3**
VIT Option: SSCP
Event: CPRC macro (Part 3)
VIT Processing Module: ISTRACSC
Applies to: MVS VM VSE

This trace record is a continuation of the CPRC entry.



00 01 02 03	04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17 18 19 1A 1B 1C	1D 1E 1F
CPR3	25 BYTES OF CPCBURC	0

Byte (hex)	Contents
00–03	Record ID: C'CPR3'
04–1C	First 25 bytes of the user request correlator (CPCBURC) entry
1D–1F	Zero.

CPTO Entry for CPS Retry

Entry:	**CPTO**
VIT Option:	CIO
Event:	CPS Retry
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCVI
Applies to:	VSE

This trace record is generated for X.21 when a call process signal (CPS) defined in the CPS retry table is received from the line and the timer interval defined in the CPS retry table for CPS expires.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPTO				ID	CFSM	PMFSM	SHSI	CUA				GROUP NCB ADDRESS				HALCB ADDRESS				PUT ADDRESS				TIME		0		RETRY		CPS	NUM

Byte (hex)	Contents
00–03	Record ID: C'CPTO'
04	Task identification key
05	Connection finite state machine (FSM; see HALCFSM in ISTHALCB for values)
06	Port manager FSM (see HALPMFSM in ISTHALCB for values)
07	X.21 short-hold mode status indicator (see PUTFLG1 in ISTPUT)
08–0B	Channel unit address of link
0C–0F	Group network control block address (ISTGRPCB)
10–13	Link network control block address (ISTHALCB)
14–17	Physical unit table address (ISTPUT)
18–19	Time interval to wait before retrying the connection
1A–1B	Zero
1C–1D	Retries remaining
1E–1F	CPS number received from the line.

CPWT Entry for CPWAIT Events (Part 1)

Entry:	**CPWT**
VIT Option:	SSCP
Event:	CPWAIT macro (Part 1)
VIT Processing Module:	ISTRACSC
Control is Returned to:	ISTCPCPW
Applies to:	MVS VM VSE

This trace record gives information about CPWAIT macroinstruction processing. It may help you determine why I/O is outstanding or why an SSCP request is failing.

This entry is eventually followed by a CPPT entry (to resume processing) or a CPPG entry (to purge the waiting task). Use the EID field to match corresponding entries.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPWT				ID	0	TYPE	0	LQAB ADDRESS				WREDATA IN WRE				WORK ELEMENT ADDRESS				CPCB OPCODE IN WORK EL OR 0				SAVE AREA ID OR 0				SENSE CODE OR 0			

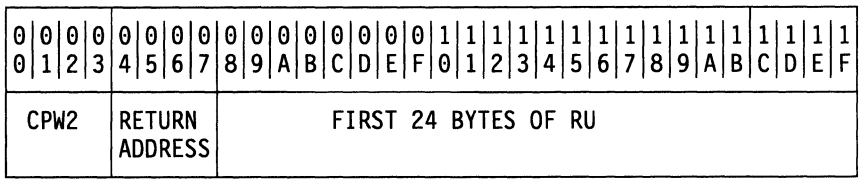
Byte (hex)	Contents
00–03	Record ID: C'CPWT'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Zero
06	Type flags • B'00000001'=PVI event • B'00000000'=non-PVI event
07	Zero
08–0B	LQAB address for this CPWAIT macro
0C–0F	Data field in ISTWRE when CPWAIT is performed
10–13	Work element address
14–17	CPCB operation code that indicates the original request for this work element or zero (see Appendix G, "Control Point/Control Block (CPCB) Operation Codes" on page 1127)
18–1B	For PVI event, caller's save area ID or zero For non-PVI event, save area ID
1C–1F	Sense code or zero.

CPW2 Entry for CPWAIT (Part 2)

Entry:	**CPW2**
VIT Option:	SSCP
Event:	CPWAIT macro (Part 2)
VIT Processing Module:	ISTRACSC
Applies to:	MVS VM VSE

This trace record is a continuation of the CPWT trace record. If an RU is fewer than 24 bytes long, only the bytes in the actual RU will be moved to the corresponding fields in the trace records. The leftover bytes will be set to zero.

Note: This trace record is produced only for PVI events where a RUPE is available.



Byte (hex)	Contents
00–03	Record ID: C'CPW2'
04–07	Return address
08–1F	First 24 bytes of RU.

CPW3 Entry for CPWAIT (Part 3)

Entry:	**CPW3**
VIT Option:	SSCP
Event:	CPWAIT macro (Part 3)
VIT Processing Module:	ISTRACSC
Applies to:	MVS VM VSE

This trace record is a continuation of the CPWAIT trace record.

0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	0 A	0 B	0 C	0 D	0 E	0 F	1 0	1 1	1 2	1 3	1 4	1 5	1 6	1 7	1 8	1 9	1 A	1 B	1 C	1 D	1 E	1 F
CPW3				RUPERC OR 0																									0		

Byte (hex)	Contents
00–03	Record ID: C'CPW3'
04–1C	RUPEURC or 0
1D–1F	Zero.

CPW4 Entry for CPWAIT (Part 4)

Entry:	**CPW4**
VIT Option:	SSCP
Event:	CPWAIT macro (Part 4)
VIT Processing Module:	ISTRACSC
Applies to:	MVS VM VSE

This trace record is a continuation of the CPW3 trace record. If an event ID (EID) is fewer than 24 bytes long, only the bytes in the actual EID will be moved to the corresponding fields in the trace records. The leftover bytes will be set to zero.

Note: This trace record is produced only when the WRE points to an EID event.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPW4				0			EIDLN	FIRST 24 BYTES OF EID																							

Byte (hex)	Contents
00–03	Record ID: C'CPW4'
04–06	Zero
07	Length of EID
08–1F	First 24 bytes of EID.

CRx Entry for ADD, DELETE, or FIND Control Block

Entry: **CRA, CRD, or CRF**
VIT Option: SSCP
Event: CRADD, CRDEL, CRFIND
VIT Processing Module: ISTRACSC
Applies to: MVS VM VSE

The correlate search control block (CR) trace record contains information about a control block add, delete, or find operation in the VTAM control point (CP).

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CRA CRD CRF			TYPE	ID	RC	MODIFIER LIST, SEARCH NUMBER, or 0										RETURN ADDR				PCID or 0								CORCB ADDR or 0			

Byte (hex)

00–02 Record ID:
C'CRA' (CRADD): Add control block
C'CRD' (CRDEL): Delete control block
C'CRF' (CRFIND): Find control block

03 Type code
0 Procedure-correlation identifier (PCID) only (6-F contain zeros)
1 PCID and modifier list (6-F contain the modifier list)
2 PCID and search number (6-7 contain the search number, 8-F contain zeros)

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Return code if specified; some FIND calls to this macro do not include a return code but use CORCB ADDR as a return code.

06–0F
- If type code = 0, this field is zero.
- If type code = 1, this field represents the PCID modifier list and is padded on the right with zeros. If search number correlation is used, this field may be zero.
- If type code = 2, this field represents the search number and is padded on the right with zeros.

10–13 Calling module's return address

14–1B PCID of the search being correlated

1C–1F Address of an ISTCORCB; bytes 4-7 of the ISTCORCB contain the address of the control block being correlated (LCB, SITCB). This field is zero if the entry type is 'CRF' and the control block was not found.

Notes:

1. A CRF entry is usually found soon after a DSP entry for LUSS or DSVC. If the CORCB ADDR is zero, the FIND failed.
2. Search number correlation is only used with DS.

DBx Entry for DBDELETE, DBQUERY, and DBUPDATE

Entry: **DBD, DBQ, DBU**
VIT Option: SSCP
Event: DBDELETE, DBQUERY, DBUPDATE
VIT Processing Module: ISTRACSC
Applies to: MVS VM VSE

This traces an operation on the APPN directory database.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
DBD DBQ DBU			0	ID		RC		FLGS			RESOURCE IDENTIFICATION					RETURN ADDRESS				DECB ADDRESS				NETID OF RESOURCE							

Byte (hex) Contents

00–02 Record ID: C'DBD', C'DBQ', C'DBU'

03 Zero

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Return code (RC)

06–07 Database information flags

Bit	Meaning
1...	Registered entry-type indicator
.1..	Cache or dynamic entry-type indicator
..1.	Suggestion or defined entry-type indicator
...1	Reserved for entry-type expansion
.... 1...	The VIT error was caused by an error in the cache data processing
.... .1..	The VIT error was caused by an error while loading the directory from storage
.... ..1.	The returned network ID on DBQUERY differs from the one queried
....x	Not used
1...	Directed failed indicator copied from directory entry
.1..	Negative cache indicator copied from directory entry
..1.	Subarea LU indicator copied from directory entry
...1	Surrogate owner indicator copied from directory entry
.... 1...	Dynamic subarea destination LU indicator copied from directory entry
.... .1..	Wildcard LU indicator copied from directory entry
.... ..1.	Non-native LU indicator copied from directory entry
....x	Not used

08–0F Resource identification

10–13 Caller's return address

14–17 Address of directory entry

18–1F Network identification of resource.

DISC Entry for SHM Disconnect

Entry:	**DISC**
VIT Option:	CIO
Event:	Disconnection
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCVI
Applies to:	VSE

This trace record is generated for X.21 short-hold mode attachments. It is written when a port is disconnected from a station.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
DISC				TIK	CFSM	PMFSM	SHSI	CUA				GROUP NCB ADDRESS				HALCB ADDRESS				PUT ADDRESS				0							

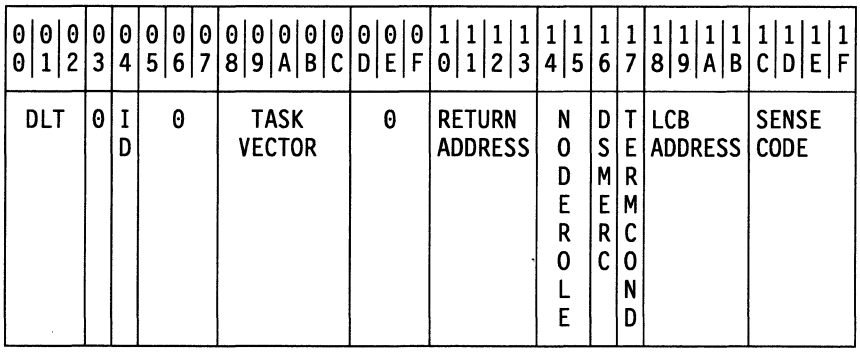
Byte (hex)	Contents
00–03	Record ID: C'DISC'
04	Task identification key
05	Connection finite state machine (FSM; see HALCFSM in ISTHALCB for values)
06	Port manager FSM (see HALPMFSM in ISTHALCB for values)
07	X.21 short-hold mode status indicator (see PUTFLG1 in ISTPUT)
08–0B	Channel unit address of link
0C–0F	Group network control block address (ISTGRPCB)
10–13	Link network control block address (ISTHALCB)
14–17	Physical unit table address (ISTPUT)
18–1F	Zero.

DLT Entry for Directory Services Locate (Part 1)

Entry: **DLT**
VIT Option: SSCP
Event: Sending out a Locate Search
VIT Processing Module: ISTRACSC
Control is Returned to: ISTDRSDL
Applies to: MVS VM VSE

This trace record is written when VTAM's directory services component forwards a locate search to some adjacent node.

Note: This trace record has been reformatted because of an increase in the size of the task vector field.



00 01 02	03	04	05 06 07	08 09 0A 0B 0C	0D 0E 0F	10 11 12 13	14 15	16	17	18 19 1A 1B	1C 1D 1E 1F
DLT	0	ID	0	TASK VECTOR	0	RETURN ADDRESS	NODEROLE	DSMERC	TERMCOND	LCB ADDRESS	SENSE CODE

Byte (hex)	Contents
00–02	Record ID: C'DLT'
03	Zero
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05–07	Zero
08–0F	The task vector bits indicate the possible tasks for this locate search. The hex values listed below are in the task vector and task vector result fields of the VIT entry.
08	**Vector** **Vector Result** X'80' Null task X'40' Directory services management exit X'20' Directory services database query X'10' Topology and routing services database query X'08' Forward to network node server X'04' One hop if directory services database is found X'02' One hop if control vector X'0E' is received for request X'01' Non-verify attempt
09	**Vector** **Vector Result** X'80' Directed due to network node destination LU hierarchy received on a search request X'40' Directed if directory services database is found

X'20' Directed if the topology and routing services valid route selection CV is returned
X'10' Directed if information is learned from scout search
X'08' Directed to a directory server
X'04' Directed to a higher function directory server
X'02' Directed to a directory server retry
X'01' Directed to a gateway node

0A

Vector Vector Result
X'80' Sequential directed search to alternate directory servers
X'40' Sequential directed search to interchange nodes
X'20' Subarea system resolution table (SRT) cache search
X'10' Subarea search after a positive cache search
X'08' Subarea search after a positive directory services directory services entry database query
X'04' Subarea search after a negative or no cache search
X'02' Domain broadcast search
X'01' Originate network broadcast search

0B

Vector Vector Result
X'80' Forward network broadcast not originated by this node
X'40' Post processing
X'20' One-hop search request due to end node destination LU hierarchy received on a search request
X'10' A cross-subnetwork directed search due to information received on the original request
X'08' A cross-network directed search due to information found in the directory services database.
X'04' A directed search due to information found in the topology and routing services database
X'02' Sequential directed search with the intent of finding the resource cross-subnetwork
X'01' Generic cache search

0C

Vector Vector Result
X'80' A subarea search due to a SESS_INIT_INFO_RPY interprocess signal
X'40' A directed search due to a SEARCH_RPY interprocess signal following a positive CACHE_SEARCH_RPY interprocess signal
X'20' A directed search due to a SEARCH_RPY interprocess signal following a positive directory services database query
X'10' Post processing

0D–0F Zero

10–13 Caller's return address

14–15 The node role for this search. More than one bit can be on.

Code Description
X'80' CP originating LU
X'40' CP destination LU
X'20' NN originating LU
X'10' NN destination LU
X'08' Owning directory server
X'04' Alternate directory server

X'02' Intermediate network server
X'01' Intermediate network node directed

Position Description

X'80' Intermediate network node broadcast

16 Return code from the directory services management exit

17 Terminating condition indicates why the search ended

Code Description

X'00' Processing can continue
X'04' Positive reply can be returned to parent
X'08' Gateway reply has been received
X'0C' Directory server reply has been received
X'10' An error was detected by a task called from the sequencer
X'14' Cleanup is pending after all replies are received from the search phase (CP session outage)
X'24' Directory services management exit routine specified no search for this request

18–1B Address of the locate control block for this search

1C–1F Sense code.

DLT2 Entry for Directory Services Locate (Part 2)

Entry: **DLT2**
VIT Option: SSCP
Event: Sending out a Locate Search
VIT Processing Module: ISTRACSC
Applies to: MVS VM VSE

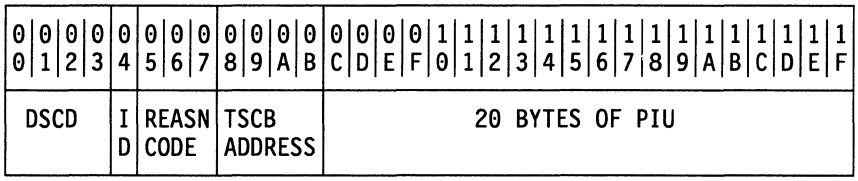
00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
DLT2				0				TASK VECTOR RESULTS								0															

Byte (hex)	Contents
00–03	Record ID: C'DLT2'
04–07	Zero
08–0F	Task Vector Results
	0 Task was not invoked.
	1 Task was invoked.
10–1F	Zero.

DSCD Entry for Discarded PIU Buffer (Part 1)

Entry:	**DSCD**
VIT Option:	PIU
Event:	Discarded PIU buffer (Part 1)
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTTSCUD
Applies to:	MVS VM VSE

This trace record is written when VTAM's TSC utility discard routine, ISTTSCUD, disposes of PIU buffers because of an error condition. The reason code and module identifier can be used to explain the reason for the discard. This entry is treated as an exception condition and is always traced, regardless of the VIT options specified, if the VIT is active.

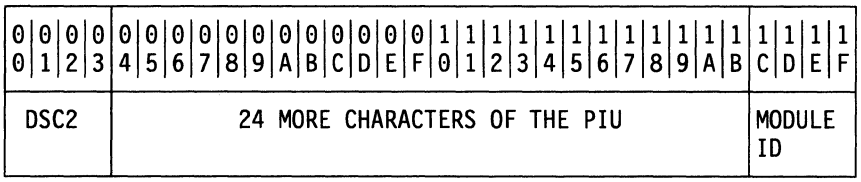


Byte (hex)	Contents
00–03	Record ID: C'DSCD'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05–07	Discard reason code in hex (see Appendix I, "Path Information Unit (PIU) Discard Reason Codes" on page 1141)
08–0B	Address of TSCB
0C–1F	First 20 characters of the PIU.

DSC2 Entry for Discarded PIU Buffer (Part 2)

Entry: **DSC2**
VIT Option: PIU
Event: Discarded PIU buffer (Part 2)
VIT Processing Module: ISTRACOT
Applies to: MVS VM VSE

This trace record is a continuation of the DSCD entry.



Byte (hex)	Contents
00–03	Record ID: C'DSC2'
04–1B	24 more characters of the PIU
1C–1F	ID of the module that is discarding the PIU.

DSP Entry for PAB Dispatch

Entry:	**DSP**
VIT Option:	PSS
Event:	PAB dispatch
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCPD
Applies to:	MVS VM VSE

The DSP entry marks the beginning of a PAB dispatch, which is VTAM's major unit of work. This unit exists until terminated by an EXIT entry. While processing, it might wait for other events to complete (WAIT). It resumes processing with a RESM entry. The unit might be interrupted by a higher priority task and continue when that task is finished.

Each PAB is identified by a unique index number. VTAM uses this number to find the called module and then creates a VIT entry for it.

To correlate all events associated with this unit, look for the request parameter header (RPH) address, which is in many trace records, and match it to the request parameter header (RPH) address in the DSP trace record. This unit has exclusive use of that request parameter header (RPH) until EXIT.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
DSP			0/WEQ	ID	CBID	FLAGS		PST ADDRESS				PAB ADDRESS				LAST WORK ELEMENT ADDRESS				CURRENT WORK ELEMENT ADDRESS				MODULE NAME OR DVT ADDRESS				RPH ADDRESS			

Byte (hex)	Contents
00–02	Record ID: C'DSP'
03	Zero, or, for very extended PABs, PAB work element queue level dispatched
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Control block ID of work element (see Appendix H, "Storage and Control Block ID Codes" on page 1137)
06	PAB flag field (PABFLAGS) Bit Meaning 1... .... PAB is unconditionally scheduled. .1.. .... PAB closedown is in progress. ..1. .... PAB is synchronous. ...1 .... PAB extension is present. .... 1... Do not dequeue work element. .... .1.. Do not detach the request parameter header (RPH). .... ..1. Indicates a very extended PAB. ....1 Indicates a slightly extended PAB.

07	PAB flag field (PABFLGS1)

Bit	Meaning
1...	Switch the PST address of this PAB's major control block to the new PST address contained in DYPNWPST.
.1..	This PAB has a data space extension.
..1.	This PAB's major control block is an FMCB.
...1	PAB can be referenced in PSW disable mode.
.... 1...	PAB is persistent.
.... .1..	APSTERM/APSINIT FMCB during PAB dispatch.
.... ..xx	Reserved.

08–0B	PST address
0C–0F	PAB address
10–13	Address of work element most recently queued to the PAB
14–17	Address of work element currently being dispatched
18–1B	Module name abbreviation (bytes 4, 5, 7, and 8 of the module name), PAB DVT address (high order bit of X'18' = 0), or one of the following abbreviations, which can represent a function or process. The module name might be unavailable if the PAB being scheduled is associated with an address space different from the current one.

Abbreviation	**Stands for**
CPSS	APPN session services for CP-CP sessions
DSVC	Directory services
DIIO	Disk I/O
LUSS	APPN session services for LU-LU sessions
MST	Management services and session initiation and termination
TRS	Topology and routing services

1C–1F	Request parameter header (RPH) address.

DTSK Entry for Detach a Subtask

Entry:	**DTSK**
VIT Option:	PSS
Event:	Detach a subtask
VIT Processing Module:	ISTRACPS
Control is returned to:	The module that issued the DETACH
Applies to:	MVS

This trace record is written when a VTAM module detaches a VTAM subtask.

Some subtasks are detached without generating this entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
DTSK				ID		RC		0				TASK NAME or 0				TCB ADDR				0				ISSUER ADDRESS				0			

Byte (hex)	Contents
00–03	Record ID: C'DTSK'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	Return code from the detach macro (ITKRC)
06–07	Zero
08–0F	Name of the subtask being detached, or 0
10–13	TCB address of the subtask being detached
14–17	Zero
18–1B	Address of the issuer of the DETACH macro
1C–1F	Zero.

ERP Entry for CIO Error Recovery

Entry:	**ERP**
VIT Option:	CIO
Event:	Error recovery for X.25 port
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	ISTAPCES
Applies to:	VM VSE

This trace record provides error recovery information for an X.25 port. An ERP entry is built when error recovery is started for an X.25 port, whether for an error, counter overflow, or shutdown recording.

The telecommunications subsystem or communication adapter uses two channel programs: a link-control channel program and a buffer-pool channel program.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ERP			TYPE	ID	CUA			0	NCB ADDR			VIRTUAL CAW				FLAG BYTES				ERCOD		SENSE		0							

Byte (hex) Contents

00–02 Record ID: C'ERP'

03 Type:
- 0C = Control link channel program
- 0D = Buffer pool channel program

04 ID is one of the following:
- VM Machine ID and Task ID
- VSE Task ID

05–07 Channel device name in EBCDIC (either a CUA or device number)

08 Zero

09–0B NCB address

0C–0F Virtual channel address word (CAW)

10–13 Flag bytes (NCBFLAGS)

14–15 Error code (ERCOD) in NCB or zero

16–17 NCB sense data in NCBSENSE or zero

18–1F Zero.

ERPx Entry for Error Recovery Procedures (Part 1)

Entry: **ERPB, ERPH, ERPI, ERPL, or ERPX**
VIT Option: CIO
Event: Error recovery for channel I/O (Part 1)
VIT Processing Module: ISTRACCI
Control is Returned to: VM ISTTSCZZ
VSE $$ABERV1
Applies to: VM VSE

This trace record is written during error recovery for channel I/O.

- ERPB is generated for error recoveries at the channel interface for a BSC line attached to the communication adapter.
- ERPH is generated for error recoveries at the channel interface for an SDLC line attached to the communication adapter.
- ERPI is generated for communication controllers and local SNA cluster controllers.
- ERPL is generated for local non-SNA cluster controllers.
- ERPX is generated for channel-to-channel-attached hosts and channel-attached IBM 3172 Interconnect Controllers.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ERPB				ID	LFSM	0			CUA			NCB ADDRESS				FLAG BYTES				ERCOD	0	SENSE		CSW							
ERPH				ID	LFSM	0			CUA			NCB ADDRESS				FLAG BYTES				0				CSW							
ERPI				ID	LNKST	0		CKEY	CUA			NCB ADDRESS				FLAG BYTES				ERCOD	0	SENSE		CSW							
ERPL or ERPX				ID	LNKST	0		CKEY	CUA			NCB ADDRESS				FLAG BYTES				0		SENSE		CSW							

Byte (hex) Contents

00–03 Record ID:
C'ERPB' for BSCLB
C'ERPH' for HALCB
C'ERPI' for ICNCB
C'ERPL' for LDNCB
C'ERPX' for XCNCB

04	ID is one of the following: • VM Machine ID and Task ID • VSE Task ID
05	Link finite state machine (LFSM) or link station state (see NCBLNKST)
06–07	VM Zero
06–08	VSE Zero
08	VM First byte of the subchannel status word
09–0B	Channel device name in EBCDIC (either a CUA or device number)
0C–0F	NCB address
10–13	Flag bytes (NCBFLAGS)
14	Error code (ERCOD) in NCB or zero
15	Return code or zero
16–17	NCB sense data (NCBSENSE) or zero
18–1F	Channel status word from NCB.

ERP2 Entry for Error Recovery Procedures (Part 2)

Entry:	**ERP2**
VIT Option:	CIO
Event:	Error recovery for channel I/O (Part 2)
VIT Processing Module:	ISTRACCI
Applies to:	VM VSE

This trace record is a continuation of the ERPB or ERPH trace entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ERP2				0				DEVICE-DEPENDENT INFORMATION																					0	ERACT	0

Byte (hex)	Contents
00–03	Record ID: C'ERP2'
04–07	Zero
08–1C	Device-dependent information shown below
1D	Zero
1E	Error action (ERACT) or zero
1F	Zero

For SDLC links (NCBTYPE=09):

08	Link error retry limit
09	Error retry count
0A	Connection finite state machine
0B	Retry operation code
0C–0F	CCW address
10	Sense byte 1
11–12	NCB sense data area
13–1C	Zero

For BSC links (NCBTYPE=0A):

08–0B	Zero
11–12	Sense data from NCBSENSE
13	Error code (NCBERCOD)
14–1C	Zero.

ERPI or ERPL Entry for Error Recovery Procedures

Entry:	**ERPI or ERPL**
VIT Option:	CIO
Event:	Error recovery
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTZBM0J for LDNCB, or ISTZBM0K for ICNCB
Applies to:	MVS

This trace record is written during error recovery for channel I/O.

- ERPI is generated for communication controllers and local SNA cluster controllers.
- ERPL is generated for local non-SNA cluster controllers.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ERPI OR ERPL				ID	STATE	0		DEVICE				NCB ADDRESS				FLAG BYTES				FLAG	CODE	SENSE		CSW							

Byte (hex)	Contents
00–03	Record ID: C'ERPI' for ICNCB C'ERPL' for LDNCB
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	Link station state (see NCBLNKST)
06–07	Zero
08–0B	Channel device name in EBCDIC (either a device address or device number)
0C–0F	NCB address
10–13	Flag bytes (NCBFLAGS)
14	Flag byte as follows: Bit Meaning ..1. .... Exception condition occurred (IOSEX flag is on) ...1 .... Error routine is in control (IOSERR flag is on)
15	I/O completion code (IOSCOD)
16–17	Sense data (IOSSNS)
18–1F	Channel status word from IOSB.

ESC Entry for the TPESC Macro

Entry:	**ESC**
VIT Option:	ESC
Event:	TPESC macro
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTESC01
Applies to:	MVS VM VSE

This trace record identifies the address of the next routine to get control in a destination vector table (DVT). The PAB address identifies the process. Using the request parameter header (RPH) address, you can correlate this trace record to the DSP entry to determine when the process was dispatched.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ESC				ID	0	PAB OF		DSP ADDRESS				PAB ADDRESS				RETURN ADDRESS				RPH WORK ELEMENT				MODULE NAME OR DVT ADDRESS				RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'ESC'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Zero
06–07	PAB offset
08–0B	Dispatch address from DVT
0C–0F	PAB address
10–13	Address of the issuer of the TPESC macro
14–17	RPH work element
18–1B	One of the following: • Module name abbreviation (bytes 4, 5, 7, and 8 of the name of the next module to get control) • PAB DVT address (high order bit of X'18' = 0)
1C–1F	Request parameter header (RPH) Address.

ETSK Entry for Exit a Subtask

Entry:	**ETSK**
VIT Option:	PSS
Event:	Exit a subtask
VIT Processing Module:	ISTRACPS
Control is returned to:	The subtask that is terminating
Applies to:	MVS

This trace record is written when a VTAM subtask is about to exit and return to the operating system supervisor.

This is not fully implemented for all subtask events in VTAM. There are a number of subtasks that terminate without this entry being generated.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ETSK				ID	0			TASK NAME								TCB ADDR				0								REG 15			

Byte (hex)	Contents
00–03	Record ID: C'ETSK'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05–07	Zero
08–0F	Name of the subtask about to end execution
10–13	TCB address of the current subtask
14–1B	Zero
1C–1F	Contents of register 15.

EXIT Entry for the TPEXIT Macro

Entry:	**EXIT**
VIT Option:	PSS
Event:	TPEXIT macro
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCTX
Applies to:	MVS VM VSE

This trace record identifies a VTAM process (PAB) that has finished executing. The PAB is rescheduled if more work elements are waiting to be processed.

In most cases, the RPH (whose address is in location 1C–1F) is freed or reused after this entry. One exception is the SMS dynamic expansion DYPAB (ATCPXPAB in module ISTORFPX), which has a dedicated RPH that is not used for anything else.

This entry is the complement of DSP. An EXIT should eventually follow every DSP. Use the RPH address to correlate the entries.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
EXIT				ID	OPTS	PAB OFF		PST ADDRESS				PAB ADDRESS				RETURN ADDRESS				WORK EL Q OR NEXT DSP Q LVL				MODULE NAME OR DVT ADDRESS				RPH ADDRESS			

Byte (hex) Contents

00–03 Record ID: C'EXIT'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 TPEXIT options

06–07 PAB offset, or zero if no PAB

Note: There is no PAB if the running process has already freed the major control block that contains the PAB.

08–0B PST address

0C–0F PAB address, or zero if no PAB

Note: There is no PAB if the running process has already freed the major control block that contains the PAB.

10–13 Address of the issuer of the TPEXIT macro

14–17 One of the following:
- PAB work element queue
- Next dispatchable queue level for a very extended PAB (see corresponding DSP entry to find the PABFLAGS to determine the PAB type)
- Zero if no PAB

Note: There is no PAB if the running process has already freed the major control block that contains the PAB.

18–1B Module name abbreviation (bytes 4, 5, 7, and 8 of the module name), PAB DVT address (high order bit of X'18' = 0), or one of the following abbreviations, which may represent a function or process. The module name might be unavailable if the PAB being scheduled is associated with an address space different from the current one.

Abbreviation	Stands for
CPSS	APPN session services for CP-CP sessions
DSVC	Directory services
DIIO	Disk I/O
LUSS	APPN session services for LU-LU sessions
MST	Management services and session initiation and termination
TRS	Topology and routing services

1C–1F Request parameter header (RPH) Address.

Note: There is no PAB if the running process has already freed the major control block that contains the PAB.

EXPN Entry for Buffer Pool Expansion

Entry:	**EXPN**
VIT Option:	SMS
Event:	Buffer pool expansion
VIT Processing Module:	ISTRACSM
Control is Returned to:	ISTORFBA or ISTORAPX
Applies to:	MVS VM VSE

This trace record is generated whenever VTAM needs to expand a buffer pool. Buffer pool expansion may occur because of a scheduled request (if done by ISTORAPX) or because additional buffers are needed to satisfy a REQSTORE macro (if done by ISTORFBA).

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
EXPN				ID	0			BPCB ADDRESS				PXB ADDRESS				EXTENT ADDRESS				PAGES		CODE	FLAGS	TOTAL NUMBER OF BUFFERS				AVAIL NUMBER OF BUFFERS			

Byte (hex) Contents

00–03 Record ID: C'EXPN'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05–07 Zero

08–0B Address of buffer pool control block (BPCB) for which expansion occurred

0C–0F Address of pool expansion block (PXB) that is expanded

10–13 Extent address (PXBSTADR)

14–15 Number of pages for an expansion in this pool. Number of pages for an expansion = (BPCBEXLN divided by 4096)

16 MVS VM Expansion failure code if expansion failed.

Note: VTAM issues codes 4 through 8 when a failure occurs during a deferred expansion.

4 Not enough CSA storage is available for the expansion.

5 VTAM cannot fix pages in storage due to insufficient page frames or some other page locking problem.

7 Storage unavailable. VTAM's CSA limit is exceeded.

8 Expansion would cause the pool to exceed its *xpanlim* specification. See "Setting Buffer Pool Allocations" in the *VTAM Network Implementation Guide* for additional information about *xpanlim.*

Note: VM VTAM issues codes 14 through 18 when a failure occurs during a dynamic expansion.

14 Not enough CSA storage is available for the expansion.

15 VTAM cannot fix pages in storage due to insufficient page frames or some other page locking problem.

VM VTAM might issue code 15 if the storage size of the virtual machine that attempts the dynamic expansion is larger than the starting address of the GCS shared system but not large enough to completely encompass all pages of the GCS shared system.

17 Storage unavailable. VTAM's CSA limit is exceeded.

18 Expansion would cause the pool to exceed its *xpanlim* specification. See "Setting Buffer Pool Allocations" in the *VTAM Network Implementation Guide* for additional information about *xpanlim*.

VSE Zero

See the explanation of IST154I in *VTAM Messages and Codes* for more information on interpreting byte 16.

17 Flag byte

Bit	Meaning
x...	1=ISTORFBA caused the expansion 0=ISTORAPX caused the expansion
.x..	1=PXB was allocated by this expansion 0=PXB already existed from prior expansion
..x.	1=Failure to obtain or fix storage 0=Success in obtaining or fixing storage

18–1B Total number of buffers in pool after this expansion (BPCBTOTL)

1C–1F Total number of available buffers in pool after this expansion.

FBLK Entry for FREEBLK Macro (Part 1)

Entry:	**FBLK**
VIT Option:	SMS
Event:	FREEBLK macro
VIT Processing Module:	ISTRACSM
Control is Returned to:	ISTORCDF
Applies to:	MVS VM VSE

This trace record shows the status of each FREEBLK request issued by VTAM components.

The FREEBLK macro is the complement of the GETBLK macro. FREEBLK must release the storage obtained by GETBLK. Each GBLK entry should eventually have a corresponding FBLK entry.

If the return code is non-zero, this entry is generated whether the SMS option is in effect or not. This event is treated as an exception condition and, therefore, is traced whenever the VIT is active.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
FBLK				ID	RTNCD	POOL	VTALC	STORAGE ADDRESS OR 0				ADDRESS OF SPTAE OR DSPSP				RETURN ADDRESS				LENGTH OF STORAGE FREED				CALLER OF UTILITY OR 0				0			

Byte (hex) Contents

00–03 Record ID: C'FBLK'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Return code

06 If FBLK is followed by FBL2:
- 00 AMUTSCBS
- 01 AMUTSCBL
- 02 CMPTAB

If FBLK is not followed by FBL2, possible storage pool types (in hex) follow (see "DISPLAY STORUSE Pools" in the *VTAM Network Implementation Guide* for more information on storage pools):

00 RUPEPRIV
01 RUPECOMM
02 SIB
03 SSCPFMCB
04 NQDAT
05 EPTDVT
06 CDRSC
07 ACDEB
08 HSQH
09 ERTE
0A WREEID
0B FMCBEXT
0C SIBEXT
0D (Not used)
0E UECB
0F IOBLOCK
10 SRTE
11 ISTTRCEL
12 UTILPVTS
13 VRPL
14 POWEPRIV
15 POWECOMM
16 PULURDTE
17 PAQ
18 RAQ
19 CPWAPVT
1A ERICPOOL
1B SIBIX
1C CDAJSCP
1D GWNAJSCP
1E IOSIB
1F DSSIB
20 UTILPVTL
21 UTILCSAS
22 UTILCSAL
23 AMU
24 HSICB
25 LMTABLE
26 SAB
27 RAB
28 PRIDBLK
29 PRIDQAB
2A AUTOLOGN
2B CPWACSA
2C PGIOBLK
2D PRDLE
2E RIBRANT MVS VM
2F CANT
30 CAB
31 CNSFACUD
32 BFRTRACE
33 DMTSQ
34 FMCB
35 PLUSFMCB MVS
36 PXBFIXED
37 PXBPAGED
38 PLUSC MVS
39 NSSCB
3A (Not used)
3B (Not used)
3C (Not used)
3D FMH5
3E OOBTSCB
3F SLD
40 NSRUS
41 NSRUL
42 RUCON
43 STB
44 WAR
45 UVRPL
46 DCX
47 PLUSDATA
48 ADJCP
49 ATGB
4A TGP
4B KEYTOKEN
4C TRSINFO
4D COS
4E NDREC
4F TGREC
50 ACPCB
51 DECB
52 NIDCB
53 CPRUPE
54 ANDCB
55 DISKIO
56 DSERVER
57 ADJNODE
58 CACHE
59 ISTSITCB
5A ISTENDEL
5B CORCB
5C LCB
5D OSCB
5E SCCB
5F UTILAREA
60 PLOCB
61 TREEBLD
62 UTILFIXD
63 POWMPRIV
64 POWMCOMM
65 POAPRIV
66 BFRTRFUL
67 SLENT
68 DYPATH
69 PCDCA

6A	BNINFO	6E	MIWKE
6B	GRINS	6F	IAP
6C	BSBEXT	70	LIA
6D	SOCKET	71	IPWKE

07	When set to 1, indicates that storage obtained through GETBLK request and converted to VTALLOC request is returned to system.
08–0B	Address of block freed (or zero if FREEBLK failed)
0C–0F	Address of storage pool anchor block (SPTAE) or, if FBLK is followed by MVS VSE FBL2, address of DSPSP
10–13	Address of the issuer of the FREEBLK macro
14–17	Length of storage actually freed not including the 8-byte header.
18–1B	Caller of utility routine or zero. If the FREEBLK macro was issued from a utility routine, the address of the utility's caller is placed here. A zero address indicates that the macro was issued directly by the caller (see return address) without a utility routine.
1C–1F	Zero.

FBL2 Entry for FREEBLK Macro (Part 2)

Entry: **FBL2**
VIT Option: SMS
Event: FREEBLK macro
VIT Processing Module: ISTRACSM
Applies to: MVS VSE

This trace record is a continuation of the FBLK entry; it is generated only if the storage that is freed is in a data space.

FBL2 returns the address of the data space descriptor control block (DSDCB) and the data space name from the DSDCB. When using IPCS, the data space name is required to look at data in the data space.

0 0 0 0 0 1 2 3	0 0 0 0 4 5 6 7	0 0 0 0 0 0 0 0 8 9 A B C D E F	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 2 3 4 5 6 7 8 9 A B C D E F
FBL2	DSDCB ADDRESS	DATA SPACE NAME	0

Byte (hex)	Contents
00–03	Record ID: C'FBL2'
04—07	Address of DSDCB
08—0F	Data space name from the DSDCB
10—1F	Zero.

GBLK Entry for GETBLK Macro (Part 1)

Entry: **GBLK**
VIT Option: SMS
Event: GETBLK macro
VIT Processing Module: ISTRACSM
Control is Returned to: ISTORCDG or ISTORCGB
Applies to: MVS VM VSE

This trace record shows the status of each GETBLK request issued by VTAM components.

The FREEBLK macro is the complement of the GETBLK macro. FREEBLK must release the storage obtained by GETBLK. Each GBLK entry should eventually have a corresponding FBLK entry.

If the return code is non-zero, this entry is generated whether the SMS option is in effect or not. It is treated as an exception condition and is always traced, regardless of the SMS option, if the VIT is active.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
GBLK				ID	RTNCD		POOL	VTALC	STORAGE ADDRESS or 0			ADDRESS OF SPTAE OR DSPSP				RETURN ADDRESS				LENGTH OF STORAGE OBTAINED				CALLER OF UTILITY OR 0				STORAGE LENGTH REQUEST OF USER			

Byte (hex) Contents

00–03 Record ID: C'GBLK'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Return code

06 If GBLK is followed by GBL2:
- 00 AMUTSCBS
- 01 AMUTSCBL
- 02 CMPTAB

If GBLK is not followed by GBL2, possible storage pool types (in hex) follow (see "DISPLAY STORUSE Pools" in the *VTAM Network Implementation Guide* for more information on storage pools):

00 RUPEPRIV
01 RUPECOMM
02 SIB
03 SSCPFMCB
04 NQDAT
05 EPTDVT
06 CDRSC
07 ACDEB
08 HSQH
09 ERTE
0A WREEID
0B FMCBEXT
0C SIBEXT
0D (Not used)
0E UECB
0F IOBLOCK
10 SRTE
11 ISTTRCEL
12 UTILPVTS
13 VRPL
14 POWEPRIV
15 POWECOMM
16 PULURDTE
17 PAQ
18 RAQ
19 CPWAPVT
1A ERICPOOL
1B SIBIX
1C CDAJSCP
1D GWNAJSCP
1E IOSIB
1F DSSIB
20 UTILPVTL
21 UTILCSAS
22 UTILCSAL
23 AMU
24 HSICB
25 LMTABLE
26 SAB
27 RAB
28 PRIDBLK
29 PRIDQAB
2A AUTOLOGN
2B CPWACSA
2C PGIOBLK
2D PRDLE
2E RIBRANT MVS VM
2F CANT
30 CAB
31 CNSFACUD
32 BFRTRACE
33 DMTSQ
34 FMCB
35 PLUSFMCB MVS
36 PXBFIXED
37 PXBPAGED
38 PLUSC MVS
39 NSSCB
3A (Not used)
3B (Not used)
3C (Not used)
3D FMH5
3E OOBTSCB
3F SLD
40 NSRUS
41 NSRUL
42 RUCON
43 STB
44 WAR
45 UVRPL
46 DCX
47 PLUSDATA
48 ADJCP
49 ATGB
4A TGP
4B KEYTOKEN
4C TRSINFO
4D COS
4E NDREC
4F TGREC
50 ACPCB
51 DECB
52 NIDCB
53 CPRUPE
54 ANDCB
55 DISKIO
56 DSERVER
57 ADJNODE
58 CACHE
59 ISTSITCB
5A ISTENDEL
5B CORCB
5C LCB
5D OSCB
5E SCCB
5F UTILAREA
60 PLOCB
61 TREEBLD
62 UTILFIXD
63 POWMPRIV
64 POWMCOMM
65 POAPRIV
66 BFRTRFUL
67 SLENT
68 DYPATH
69 PCDCA

6A	BNINFO	6E	MIWKE
6B	GRINS	6F	IAP
6C	BSBEXT	70	LIA
6D	SOCKET	71	IPWKE

07	When set to 1, actual GETBLK request converted to VTALLOC request
08–0B	Address of block obtained (or zero if GETBLK failed)
0C–0F	Address of storage pool anchor block (SPTAE) or, if GBLK is followed by MVS VSE GBL2, address of DSPSP
10–13	Address of the issuer of the GETBLK macro
14–17	Length of storage actually obtained, rounded to the correct subpool length. This length should not include the 8-byte GETBLK header. The storage address (bytes 08–0B) and this value represent the address of the next available storage.
18–1B	Caller of utility routine or zero. If the GETBLK macro was issued from a utility routine, the address of the utility's caller is placed here. A zero address indicates that the macro was issued directly by the caller (see return address) without a utility routine.
1C–1F	Length of the storage specified by the user.

GBL2 Entry for GETBLK Macro (Part 2)

Entry: **GBL2**
VIT Option: SMS
Event: GETBLK macro
VIT Processing Module: ISTRACSM
Applies to: MVS VSE

This trace record is a continuation of the GETBLK entry and provides the address of the data space descriptor control block (DSDCB). This trace record is generated only if the storage that is acquired is in a data space.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
GBL2				DSDCB ADDRESS				DATA SPACE NAME								0															

Byte (hex) Contents

00–03 Record ID: C'GBL2'
04—07 Address of DSDCB
08—0F Data space name from the DSDCB. When you are using IPCS, the data space name is required often to look at data in the data space.
10—1F Zero.

GNAM Entry for GNAME Macro Invoked (Part 1)

Entry: **GNAM**
VIT Option: SSCP
Event: GNAME macro invocation
VIT Processing Module: ISTRACSC
Control is Returned to: ISTCPCGM
Applies to: MVS

This trace record is written when generic mapping information in the coupling facility structure ISTGENERIC is created, deleted, or changed using the GNAME macroinstruction.

00	01	02	03	04	05	06	07	08 09 0A 0B 0C 0D 0E 0F	10 11 12 13	14 15 16	17	18 19 1A 1B	1C 1D 1E 1F
GNAM				ID	RTNCD	FUNC	FLAGS	GENERIC RESOURCE NAME	RETURN ADDRESS	0	RCNT	GENERIC NUMBER	RPH ADDRESS

Byte (hex)	Contents
00–03	Record ID: C'GNAM'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	Return code from the GNAME macro
06	Reason macro invoked X'01' Find generic resource mapping X'02' Find generic resource mapping or USERVAR X'03' Add the application program network name to generic mapping X'04' Delete the application program network name from generic mapping X'05' Write generic mapping to coupling facility structure X'06' Free local copy of generic mapping X'07' Find a generic resource name for an application program network name. X'08' Find the generic number for a generic resource name
07	Event flags: Bit Meaning 1... .... The addition or deletion of an application program network name is due to a change in CP-CP status. .1.. .... The addition or deletion of an application program network name is due to a SETLOGON GNAMEADD or SETLOGON GNAMEDEL. ..xx xxxx Zero
08–0F	Generic name
10–13	Address of the invoker of the GNAME macro
14–16	Zero
17	Resolution count or zero
18–1B	The generic number if available, otherwise zero
1C–1F	Request parameter header (RPH) Address.

GNA2 Entry for GNAME Macro Invoked (Part 2)

Entry: **GNA2**
VIT Option: SSCP
Event: GNAME macro invocation
VIT Processing Module: ISTRACSC
Applies to: MVS

This trace record is a continuation of the GNAM entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
GNA2				SESS COUNT or 0				PCID								NETID OF APPLICATION PROGRAM NETWORK NAME								APPLICATION PROGRAM NETWORK NAME							

Byte (hex)	Contents
00–03	Record ID: C'GNA2'
04–07	Session count or zero
08–0F	PCID if associated with a session, otherwise zero
10–17	Network ID of application program network name
18–1F	Application program network name.

HIOx Entry for Halt I/O

Entry:	**HIOB, HIOH, HIOI, or HIOX**
VIT Option:	CIO
Event:	Halt I/O
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCLR
Applies to:	MVS VM VSE

This trace record is written when a Halt I/O SVC is issued to end a currently executing channel program.

- HIOB is generated when an HIO SVC ends a channel program using a BSC line attached to the communication adapter.
- HIOH is generated when an HIO SVC ends a channel program using an SDLC line attached to the communication adapter.
- HIOI is generated when a VM GENIO HALT or an MVS HIO SVC ends a channel program to a local SNA controller as a result of VARY INACT,FORCE processing.
- HIOX is generated when an HIO SVC ends communication between channel-to-channel-attached hosts.

An HIO entry should be followed shortly by an interrupt (INT) entry. Match the CUAs in the HIO and INT entries to be sure they are for the same device. See *VTAM Data Areas for MVS/ESA* or *VTAM Data Areas for VM/ESA* for a description of the NCB, ICNCB, and XCNCB fields.

00 01 02 03	04	05	06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17 18 19 1A 1B 1C 1D 1E 1F
HIOB HIOH HIOI HIOX	ID	STATE	MODID	DEVICE	NCB ADDRESS	FLAG BYTES	0

Byte (hex)	Contents
00–03	Record ID: C'HIOB', 'HIOH', or 'HIOX'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	For HIOB, link station state (BSCLKFSM) For HIOH, link station state (HALLFSM) For HIOI, link state (NCBLNKST) For HIOX, for a channel-to-channel adaptor, station state (XCNSSFSM) For HIOX, for a LAN channel station, link state (NCBLNKST)
06–07	Module ID (last 2 characters of TSC module name)
08–0B	Channel device name in EBCDIC (either a device address or device number)
0C–0F	NCB address
10–13	Flag bytes (NCBFLAGS)

14–1F	Zero.

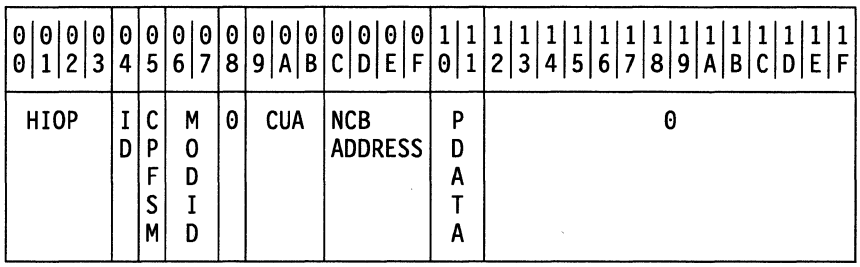
HIOP Entry for Halt I/O

Entry:	**HIOP**
VIT Option:	CIO
Event:	Halt I/O
VIT Processing Module:	ISTRAC6T
Control is Returned to:	ISTTSC6L or ISTTSC6P
Applies to:	VM VSE

This trace record is created when a Halt I/O SVC is issued to end a currently executing channel program. This entry might be followed by an interrupt (INT) entry. Match the CUAs in the HIO and INT entries to be sure they are for the same device. A Halt I/O SVC is issued against each of four CUAs during initialization or termination of the token-ring subsystem.

Notes:

1. The HIOP entry cannot be generated during initialization of a token-ring LAN attached through an IBM 3172 Interconnect Controller.
2. VSE The HIOP trace entry is generated during initialization.



Byte (hex)	Contents
00–03	Record ID: C'HIOP'
04	ID is one of the following: • VM Machine ID and Task ID • VSE Task ID
05	Channel program FSM 00 Reset 01 Sense ID 02 Interrupt port 03 Control port 04 Running 05 Sense 06 Orderly halt (HLTC) 07 Termination 08 Shutdown
06–07	Module ID (last 2 characters of TSC module name)
08	Zero
09–0B	Channel device name in EBCDIC (a device number)
0C–0F	NCB address
10	Port state (PDATA) 00 Reset 01 Halt 02 Halt complete 03 Sense ID

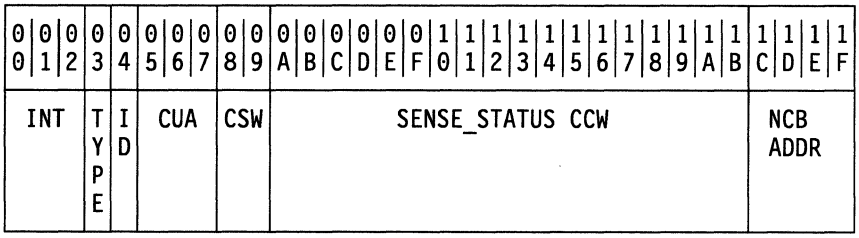
	04 Sense ID complete
	05 Interrupt pending
	06 Active
	07 Sense
11	Port type (PDATA)
	00 Interrupt
	01 Control
	02 Inbound
	03 Outbound
12–1F	Zero.

INT Entry for Channel Interrupt for X.25 Port

Entry:	**INT**
VIT Option:	CIO
Event:	Channel interrupt for X.25 port
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	ISTAPCES
Applies to:	VM VSE

This trace record provides information about interruptions at the telecommunications subsystem or communication adapter interface. An entry is built for each I/O interrupt for an X.25 port.

The telecommunications subsystem or communication adapter uses two channel programs: a control-link channel program and a buffer-pool channel program.



Byte (hex)	Contents
00–02	Record ID: C'INT'
03	Type: 0C = Control link channel program 0D = Buffer pool channel program
04	ID is one of the following: • VM Machine ID and Task ID • VSE Task ID
05–07	Channel device name in EBCDIC (either a CUA or device number)
08–09	Channel status word from NCB
0A–1B	NCB sense data in NCBSENSE
1C–1F	NCB address.

INTx Entry for Channel Interrupt (Part 1)

Entry:	**INTI, INTL, or INTX**
VIT Option:	CIO
Event:	Channel interrupt
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCLC
Applies to:	MVS

This trace record is written when a channel program interrupt occurs. Every INT entry has a previously issued SIO entry, although it may have been issued some time ago. Use the CUA field to correlate SIO and INT entries. The SIO entry provides additional information about this completing channel program.

See *VTAM Data Areas for MVS/ESA* for a description of the NCB fields. See the diagnostic manuals for your operating system for a description of the SIO fields.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
INTI INTL INTX				ID	STATE	EOPCD	0	DEVICE				NCB ADDRESS				FLAG BYTES				FLAG	CODE	SENSE	SOR 0	CSW							

Byte (hex)	Contents
00–03	Record ID: C'INTI' for ICNCB C'INTL' for LDNCB C'INTX' for XCNCB
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	Link station state (NCBLNKST) for ICNCB and LDNCB Station state (XCNSSFSM) for XCNCB
06	Ending operation code
07	Zero
08–0B	Channel device name in EBCDIC (either a device address or device number)
0C–0F	NCB address
10–13	Flag bytes (NCBFLAGS)
14	Flag byte as follows: Bit Meaning 1... .... Running in disabled interrupt exit (IOSDIE is not zero) .1.. .... ERP work area address exists (IOSERP is not zero) ..1. .... Exception condition (IOSEX flag is on) ...1 .... Error routine in control (IOSERR flag is on)
15	I/O completion code (IOSCOD)
16–17	Sense data in IOSSNS if this was a sense channel program; otherwise, zero
18–1F	Channel status word from IOSB.

INT2 Entry for Channel Interrupt (Part 2)

Entry: **INT2**
VIT Option: CIO
Event: Channel interrupt (Part 2)
VIT Processing Module: ISTRACCI
Applies to: MVS VM VSE

This trace record is a continuation of the INT entry.

00 01 02 03	04 05 06 07	08 09 0A 0B 0C 0D 0E 0F	10 11 12 13 14 15 16 17	18 19 1A 1B 1C 1D 1E 1F
INT2	0	MPC TRANSMIT BLOCK HEADER	FIRST MPC PDU HEADER	0

Byte (hex)	Contents
00–03	Record ID: C'INT2'
04–07	Zero
08–0F	MPC transmit block header
10–17	MPC PDU header for first PIU in the transmit buffer
18–1F	Zero.

INTx Entry for Channel Interrupt (Part 1)

Entry:	**INTB, INTH, INTI, INTL, or INTX**
VIT Option:	CIO
Event:	Channel interrupt (Part 1)
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCLC
Applies to:	VM

This trace record is written when a channel program interrupt occurs.

- INTB is generated for interrupts at the channel interface for a BSC line attached to the communication adapter.
- INTH is generated for interrupts at the channel interface for an SDLC line attached to the communication adapter.
- INTI is generated for interrupts between communication controllers and local SNA cluster controllers.
- INTL is generated for interrupts from local non-SNA cluster controllers.
- INTX is generated for interrupts between channel-to-channel-attached hosts.

Every INT entry has a previously issued SIO entry, although it may have been issued some time ago. Use the CUA field to correlate SIO and INT entries. The SIO entry provides additional information about this completing channel program.

See *VTAM Data Areas for VM/ESA* for a description of the NCB fields. See the diagnostic manuals for your operating system for a description of the IOS fields.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
INTB or INTH				ID	LFSM	0			CUA			NCB ADDRESS				FLAG BYTES				0	TYPE	SENSE		CSW							
INTI				ID	LNKST	0		CKEY	CUA			NCB ADDRESS				FLAG BYTES				0	TYPE	SENSE		CSW							
INTL				ID	LNKST	0		CKEY	CUA			NCB ADDRESS				FLAG BYTES				0	TYPE	0		CSW							
INTX				ID	STATE	EOPCD		0 CKEY	CUA			NCB ADDRESS				FLAG BYTES				FLAG	TYPE	SENSE or 0		CSW							

Byte (hex)	Contents
00–03	Record ID: C'INTB' for BSCTYPE C'INTH' for HALTYPE C'INTI' for ICNCB C'INTL' for LDNCB C'INTX' for XCNCB
04	Machine ID and Task ID
05	For INTB and INTH, link finite state machine (LFSM) For INTI and INTL, link station state (NCBLNKST) For INTX, station state (XCNSSFSM)
06	For INTX ending operation code; otherwise, zero
07	Zero
08	For INTB and INTH, zero For INTI, INTL, and INTX, first byte of the subchannel status word
09–0B	Channel device name in EBCDIC (either a CUA or device number)
0C–0F	NCB address
10–13	Flag bytes (NCBFLAGS)
14	For INTX, flag byte as follows; otherwise, zero Bit Meaning 1... .... Running in disabled interrupt exit (IOSDIE is not zero) .1.. .... ERP work area address exists (IOSERP is not zero) ..1. .... Exception condition (IOSEX flag is on) ...1 .... Error routine in control (IOSERR flag is on)
15	Interrupt type (ICBTYPE)
16–17	NCB sense data in NCBSENSE if this was a sense channel program; otherwise, zero
18–1F	Channel status word from NCB.

INT2 Entry for Channel Interrupt (Part 2)

Entry:	**INT2**
VIT Option:	CIO
Event:	Channel interrupt (Part 2)
VIT Processing Module:	ISTRACCI
Applies to:	VM

This trace record is a continuation of the INT entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
INT2				0				DEVICE-DEPENDENT INFORMATION								0														ERACT	0

Byte (hex)	Contents
00–03	Record ID: C'INT2'
04–07	Zero
08–13	Device-dependent information shown below
14–1D	Zero
1E	Error action (ERACT) or zero
1F	Zero

For SDLC links (NCBTYPE=09):

08	Return code
09	Buffer offset (SPLOFFST)
0A	SDLC station address (SPLADDR)
0B	Control flag (SPLCTLFG)
0C	Current SDLC number of last I-frame sent (SPLNSCUR)
0D	Last acknowledged I-frame (SPLNSACK)
0E	Next I-frame to be received (SPLNRACC)
0F	Command-in field (SCXCMDIN)
10	Count of frames sent and acknowledged (SCXCFRS)
11	Number of receive buffers used (SCXCRBUF)
12–13	Exception flags and exception code (SCXEXFCD)

For BSC links (NCBTYPE=0A):

08	First selection character (BSCSEL1)
09	First device address (BSCDEV1)
0A	Index of BPL entry for which input data was received (BSCPESV)
0B	Device address from which input data was received (BSCDEVSV)
0C	Index for last station for which there was output data (BSCISOD)
0D	First response byte (BSCRSPA1)
0E	Second response byte (BSCRSPA2)
0F	Last block type received (BSCLBR)
10	Current acknowledgment (BSCALTAK)
11	Error retry count (BSCRCNT)
12	Connection manager flags (BSCCMIO)
13	BSC flags (BSCFLAGS).

INTx Entry for Channel Interrupt (Part 1)

Entry:	**INTB, INTH, INTI, or INTL**
VIT Option:	CIO
Event:	Channel interrupt (Part 1)
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCLC
Applies to:	VSE

This trace record is written when a channel program interrupt occurs.

- INTB is generated for interrupts at the channel interface for a BSC line attached to the communication adapter.
- INTH is generated for interrupts at the channel interface for an SDLC line attached to the communication adapter.
- INTI is generated for interrupts between communication controllers and local SNA cluster controllers.
- INTL is generated for interrupts from local non-SNA cluster controllers.

Every INT entry has a previously issued SIO entry, although it may have been issued some time ago. Use the CUA field to correlate SIO and INT entries. The SIO entry provides additional information about this completing channel program.

See *VTAM Data Areas for VSE/ESA* for a description of the NCB fields. See the diagnostic manuals for your operating system for a description of the IOS fields.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
INTB or INTH				ID	LFSM		0		CUA			NCB ADDRESS				FLAG BYTES				0	CODE		SENSE		CSW						
INTI				ID	LNKST		0		CUA			NCB ADDRESS				FLAG BYTES				ERCOD	CODE		SENSE		CSW						
INTL				ID	LNKST		0		CUA			NCB ADDRESS				FLAG BYTES				0	CODE		0		CSW						

Byte (hex) Contents

00–03 Record ID:
- C'INTB' for BSCTYPE
- C'INTH' for HALTYPE
- C'INTI' for ICNCB
- C'INTL' for LDNCB

04 Task ID

05	For INTB and INTH, link finite state machine (LFSM) For INTI and INTL, link station state (NCBLNKST)
06–08	Zero
09–0B	Channel device name in EBCDIC (either a CUA or device number)
0C–0F	NCB address
10–13	Flag bytes (NCBFLAGS)
14	ERCOD or zero
15	I/O completion code (IOSCOD)
16–17	Actual sense data from the channel if this was a sense channel program; otherwise, zero
18–1F	Channel status word from NCB.

INT2 Entry for Channel Interrupt (Part 2)

Entry: **INT2**
VIT Option: CIO
Event: Channel interrupt (Part 2)
VIT Processing Module: ISTRACCI
Applies to: VSE

This trace record is a continuation of the INT entry.

0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	0 A	0 B	0 C	0 D	0 E	0 F	1 0	1 1	1 2	1 3	1 4	1 5	1 6	1 7	1 8	1 9	1 A	1 B	1 C	1 D	1 E	1 F
INT2				0				DEVICE-DEPENDENT INFORMATION												0											ERACT	

Byte (hex)	Contents
00–03	Record ID: C'INT2'
04–07	Zero
08–13	Device-dependent information shown below
14–1D	Zero
1E	Error action (ERACT) or zero
1F	Zero

For SDLC links (NCBTYPE=09):

08	Return code
09	Buffer offset (SPLOFFST)
0A	SDLC station address (SPLADDR)
0B	Control flag (SPLCTLFG)
0C	Current SDLC number of last I-frame sent (SPLNSCUR)
0D	Last acknowledged I-frame (SPLNSACK)
0E	Next I-frame to be received (SPLNRACC)
0F	Command-in field (SCXCMDIN)
10	Count of frames sent and acknowledged (SCXCFRS)
11	Number of receive buffers used (SCXCRBUF)
12–13	Exception flags and exception code (SCXEXFCD)

For BSC links (NCBTYPE=0A):

08	First selection character (BSCSEL1)
09	First device address (BSCDEV1)
0A	Index of BPL entry for which input data was received (BSCPESV)
0B	Device address from which input data was received (BSCDEVSV)
0C	Index for last station for which there was output data (BSCISOD)
0D	First response byte (BSCRSPA1)
0E	Second response byte (BSCRSPA2)
0F	Last block type received (BSCLBR)
10	Current acknowledgment (BSCALTAK)
11	Error retry count (BSCRCNT)
12	Connection manager flags (BSCCMIO)
13	BSC flags (BSCFLAGS).

INTP Entry for Channel Interrupt

Entry:	**INTP**
VIT Option:	CIO
Event:	Channel interrupt
VIT Processing Module:	ISTRAC6T
Control is Returned to:	ISTTSC6E
Applies to:	VM VSE

This trace record is created when a channel program interrupt occurs. The entry may signal the completion of a channel program; if so, it has a previously-issued SIO entry. Use the CUA field to correlate SIO and INT entries. The SIO entry provides additional information about a completing channel program.

The CUA field may contain the address specified on the PORT definition statement for a LAN major node. If so, the value in the NCB address field corresponds to the LAN major node's NCB.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
INTP				ID	CPFSM	0			CUA			NCB ADDRESS				PDATA		0		STIM	CODE	SENSE	0	ENDING CSW							

Byte (hex) Contents

00–03 Record ID: C'INTP'.

04 ID is one of the following:
- VM Machine ID and Task ID
- VSE Task ID

05 Channel program FSM
- 00 Reset
- 01 Sense ID
- 02 Interrupt port
- 03 Control port
- 04 Running
- 05 Sense
- 06 Orderly halt (HLTC)
- 07 Termination
- 08 Shutdown

06–08 Zero

09–0B Channel device name in EBCDIC (either a CUA or device number)

0C–0F NCB address

10 Port state (PDATA)
- 00 Reset
- 01 Halt
- 02 Halt complete
- 03 Sense ID
- 04 Sense ID complete
- 05 Interrupt pending
- 06 Active

	07 Sense
11	Port type (PDATA)
	00 Interrupt
	01 Control
	02 Inbound
	03 Outbound
12–13	Zero
14	I/O stimulus for FSM manager
	06 Normal completion
	07 Attention
	08 Attention and unit exception
	09 Unit check
	0A Error
15	Channel-command code for the last CCW
	04 Sense (3088 compatibility)
	08 TIC
	C1 Data buffer write
	C2 Data buffer read
	C4 Sense control unit state
	C5 Control block write
	C6 Control block read
	C7 Data synchronization
	C9 Data parameters write
	CB Set CETI mode on
	CD Interrupt parameters write
	E4 Sense ID
16	First byte of sense data
	80 Command reject
	02 Control block reject
17	Zero
18–1F	Channel status word.

IO1 Entry for TPIO Request (Part 1)

Entry:	**IO1**
VIT Option:	API
Event:	Application program request (Part 1)
VIT Processing Module:	ISTRACAP
Control is Returned to:	ISTAPCRS or ISTOCCSM
Applies to:	MVS VM VSE

This trace record shows API requests. It also provides the RPL information supplied by a VTAM application program when it issues an RPL-based macroinstruction (such as SEND or OPNDST). The API routine ISTAICIR issues the TPIO SVC. ISTAPCTI gets control from the SVC, does validity checking, and queues the RPL to the PST request PAB. After the PST request PAB is dispatched, ISTAPCRS gets control and writes the entry, or calls ISTOCCSM to write the entry.

See "RPL Fields Set by VTAM" in *VTAM Programming* for an explanation of the RPL fields.

The process that is started with this event will be completed by a post (UP), or an exit (RE or UE). To correlate the two entries, match the RPL addresses.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
IO1			REQTY	ID	EXTDS	OPT4	OPT1	RPL ADDRESS				NIB ADDRESS OR CID				RH3	SRTYP	VTFL1	VTFL2	CHN	RU CNTL CODE			OPTC2				OPTC3			

Byte (hex) Contents

00–02 Record ID: C'IO1'

03 RPL request type (RPLREQ)
- X'15' SETLOGON
- X'16' SIMLOGON
- X'17' OPNDST
- X'19' CHANGE
- X'1A' INQUIRE
- X'1B' INTRPRET
- X'1F' CLSDST
- X'22' SEND
- X'23' RECEIVE
- X'24' RESETSR
- X'25' SESSIONC
- X'27' SENDCMD
- X'28' RCVCMD
- X'29' REQSESS
- X'2A' OPNSEC
- X'2C' TERMSESS

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM

- Task ID VSE

05	Exit definition (RPLEXTDS)
06	Option code byte 4 (RPLOPT4)
07	Option code byte 1 (RPLOPT1)
08–0B	RPL address
0C–0F	NIB address or CID
10	Third byte of RH (RPLRH3)
11	Send/receive type (RPLSRTYP)
12	VTAM flags (RPLVTFL1)
13	Post/respond flags (RPLVTFL2)
14	RU chain position (RPLCHN)
15–17	RU control codes (RPLCNTRL)
18–1B	VTAM options (RPLOPTC2)
1C–1F	VTAM options (RPLOPTC3).

IO2 Entry for TPIO Request (Part 2)

Entry: **IO2**
VIT Option: API
Event: Application program request (Part 2)
VIT Processing Module: ISTRACAP
Applies to: MVS VSE

This trace record is a continuation of the IO1 entry. It shows additional information about the request.

0 0 0 1	0 2	0 3	0 4 0 5 0 6 0 7	0 8 0 9 0 A 0 B 0 C 0 D 0 E 0 F	1 0 1 1 1 2 1 3 1 4 1 5 1 6 1 7	1 8 1 9 1 A 1 B 1 C 1 D 1 E 1 F
IO2	0	RETURN ADDRESS	LOGON MODE NAME OR 0	APPLICATION LU NAME (if available)	PARTNER LU NAME (if available)	

Byte (hex)	Contents
00–02	Record ID: C'IO2'
03	Zero
04–07	Address of the issuer of the macro
08–0F	For SIMLOGON, OPNDST, CLSDST, or REQSESS: Logon mode name (NIBLMODE) or zero if the NIB is not available For OPNSEC or TERMSESS: Zero
10–17	Application LU name if available.
18–1F	Partner LU name if available.

IO2 Entry for TPIO Request (Part 2)

Entry: **IO2**
VIT Option: API
Event: Application program request (Part 2)
VIT Processing Module: ISTRACAP
Applies to: VM

This trace record is a continuation of the IO1 entry. It shows information about the PLU and SLU, such as some of the parameters exchanged and the names of the PLU and SLU.

0 0 0 0 0 1 2 3	0 0 0 0 4 5 6 7	0 0 0 0 0 0 0 0 8 9 A B C D E F	1 1 1 1 1 1 1 1 0 1 2 3 4 5 6 7	1 1 1 1 1 1 1 1 8 9 A B C D E F
IO2	0	RETURN ADDRESS	LOGON MODE NAME or 0	PLU NAME (if available)	SLU NAME (if available)

Byte (hex)	Contents
00–02	Record ID: C'IO2'
03	Zero
04–07	Address of the issuer of the macro
08–0F	Logon mode name (NIBLMODE) or zero if the NIB is not available
10–17	Primary logical unit (PLU) name or zero
18–1F	Secondary logical unit (SLU) name or zero.

IO3 Entry for TPIO Request (Part 3)

Entry: **IO3**
VIT Option: API
Event: Application program request (Part 3)
VIT Processing Module: ISTRACAP
Applies to: MVS VSE

This trace record is a continuation of the IO2 entry. It shows additional information about the request.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
IO3			0	ADDRESS OF USER DATA, BIND, OR 0				TARGET LU NETID OR N I B C O U N T RC NPO IAU BRN MT								0								TARGET LU NAME (if available) OR 0								PARTNER LU NETID (if available)							

Byte (hex) Contents

00–02 Record ID: C'IO3'

03 Zero

04–07 For SIMLOGON, CLSDST, or REQSESS: Address of user data (RPLAREA)
For OPNDST or OPNSEC: Address of BIND (NIBNDAR) or zero if the NIB is not available

08–0F Target LU network identifier (when for CLSDST PASS).

Byte (hex)	Contents
08–09	Count of node initialization blocks (NIBs) in NIB list (when not for CLSDST PASS).
0A–0B	Count of node initialization blocks (NIBs) in NIB list with NIBRPARM=0 (when not for CLSDST PASS).

10–17 Target logical unit name, if available, or zero

18–1F Partner LU network identifier, if available.

IO3 Entry for TPIO request (Part 3)

Entry: **IO3**
VIT Option: API
Event: Application program request (Part 3)
VIT Processing Module: ISTRACAP
Applies to: VM

This trace record is a continuation of the IO2 entry.

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 2 3 4 5 6 7 8 9 A B C D E F 0 1 2 3 4 5 6 7 8 9 A B C D E F			
IO3	0	ADDRESS OF BIND or 0	0

Byte (hex)	Contents
00–02	Record ID: C'IO3'
03	Zero
04–07	Address of BIND (NIBNDAR) or zero if the NIB is not available
08–1F	Zero.

IPC Entry for Socket API Call Completion

Entry: **IPC**
VIT Option: TCP
Event: When a simulated socket API call completes
VIT Processing Module: ISTRACIP
Control is Returned to: ISTIPCAS
Applies to: MVS

This trace record is written when a socket API that was previously issued completes.

The trace record indicates which call was issued and which socket the call was issued for. It also indicates which subtask the call was issued from and points to the VTAM control block that represents the socket/session association.

See Table 80 on page 1165 to determine what book contains more information about the socket API calls.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
IPC				ID	IUCV RC	0		API CALL	NUMBER	DESCR 0 / ERROR		SOCKET MAJOR CONTROL BLOCK ADDRESS				SOCKET APPL SUBTASK CONTROL BLOCK ADDRESS				SOCKET API PARM LIST ADDRESS				API RC		API ERROR	NUMBER	RPH ADDRESS			

Byte (hex) Contents

00–03 Record ID: C'IPC'

04 ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.

05 IUCV return code

Code	Meaning
X'FF'	Issue last error call to get return code
X'00'	No IUCV error
X'01'	IUCV-related error
X'02'	IUCV path severed

06–07 Zero

08–09 Socket API call number

0A–0B Socket descriptor or 0

0C–0F Socket major control block address

10–13 Socket application subtask control block address

14–17 Socket API Parameter List Address

18–19 Socket API call return code

1A–1B Socket API call error number

1C–1F Request parameter header (RPH) Address.

IPI Entry for Socket API Call

Entry: **IPI**
VIT Option: TCP
Event: When VTAM code issues a simulated socket API call
VIT Processing Module: ISTRACIP
Control is Returned to: ISTIPCAS
Applies to: MVS

This trace record is written when socket API calls are made to send/receive data on an internet protocol (IP) network.

The trace record indicates which call is being issued and which socket the call is being issued for. It also indicates which subtask the call is being issued from and points to the VTAM control block that represents the socket/session association.

See Table 80 on page 1165 to determine what book contains more information about the socket API calls.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
IPI				ID	0			API CALL	NUMBER	DESCRIPTOR		SOCKET MAJOR CONTROL BLOCK ADDRESS				SOCKET APPL SUBTASK CONTROL BLOCK ADDRESS				SOCKET API PARM LIST ADDRESS				0				RPH ADDRESS			

Byte (hex) Contents

00–03	Record ID: C'IPI'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05–07	Zero
08–09	Socket API call number
0A–0B	Socket descriptor or 0
0C–0F	Socket major control block address
10–13	Socket application subtask control block address
14–17	Socket API parameter list (ISTSOAPI) address
18–1B	Zero
1C–1F	Request parameter header (RPH) Address.

IRBD Entry for IRB Dispatch

Entry:	**IRBD**
VIT Option:	PSS
Event:	IRB dispatch
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCIE
Applies to:	MVS

This trace record provides information when an interrupt request block (IRB) is dispatched. See Table 80 on page 1165 to determine what book contains a map of the TCB.

Note: For this trace record to be created, in addition to specifying the PSS option you must specify IRB or BOTH on the PSSTRACE start option.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
IRBD				ID	0			PST ADDRESS				PST SYNC TPPOSTD QUEUE				PST SYNC NORMAL QUEUE				IRB ADDRESS				TCB ADDRESS				FLG1		0	

Byte (hex)	Contents
00–03	Record ID: C'IRBD'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05–07	Zero
08–0B	PST address
0C–0F	PST synchronous TPPOSTed queue anchor (PSTSTPP)
10–13	PST synchronous normal queue anchor (PSTSNRM)
14–17	IRB address (TCBRBP)
18–1B	TCB address (PSTTCBA)
1C–1D	PST flags (PSTFLG1)
1E–1F	Zero.

IRBX Entry for IRB Exit

Entry:	**IRBX**
VIT Option:	PSS
Event:	IRB exit
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCSD
Applies to:	MVS

This trace record provides information about an IRB exit.

Note: For this trace record to be created, in addition to specifying the PSS option you must specify IRB or BOTH on the PSSTRACE start option.

0 0 0 0 0 1 2 3	0 4	0 5 6 7	0 8 9 A B	0 C D E F	1 0 1 2 3	1 4 5 6 7	1 8 9 A B	1 C D E F
IRBX	ID	0	PST ADDRESS	NEW TCB ADDRESS	OLD TCB ADDRESS	NEW ASCB ADDRESS	OLD ASCB ADDRESS	IRB ADDRESS

Byte (hex)	Contents
00–03	Record ID: C'IRBX'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05–07	Zero
08–0B	PST address
0C–0F	New TCB address
10–13	Old TCB address
14–17	New ASCB address
18–1B	Old ASCB address
1C–1F	IRB address.

LCSx Entry for LAN Channel Station Error (Part 1)

Entry: **LCSL, LCSP, LCSS, or LCSX**
VIT Option: LCS
Event: LAN channel station error (Part 1)
VIT Processing Module: ISTRACLS
Applies to: MVS VM VSE

If the LCS trace option is specified, an LCSX trace record is created for *every* data frame VTAM receives from or sends to an IBM 3172 Interconnect Controller.

If the VIT is active and VTAM receives a frame that is not valid from an IBM 3172 Interconnect Controller, an LCSL, LCSP, or LCSS trace record is created depending on the following error conditions.

Note: These records are created as an exception condition.

- The LCSL trace record is created when VTAM receives link-related data that is not valid.
- The LCSP trace record is created when VTAM receives adapter-related data that is not valid.
- The LCSS trace record is created when VTAM receives SAP-related data that is not valid.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
LCSL LCSP LCSS LCSX				ID		DIR		ADAP		TYPE		DEVICE				CB ADDRESS				REQUEST MODULE ABBREV NAME				DATA LENGTH				DATA ADDRESS				STATE		CODE		0			

Byte (hex) Contents

00–03 Record ID:
- C'LCSL' for information about a link
- C'LCSP' for information about an adapter
- C'LCSS' for information about an SAP
- C'LCSX' for information about a channel

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Direction

Code	Meaning
C'I'	Data received inbound
C'O'	Data sent outbound

06 Adapter number

07 Adapter type
- 0 = Control communication
- 2 = 802.5 (token-ring)
- 4 = CSMA/CD–802.3

5 = 802.4 (token-bus), MAP 3.0
6 = FDDI (fiber distributed data interface)

08–0B Channel device name in EBCDIC (either a device address or device number)

0C–0F Control block address
- For LCSL, address of LSNCB
- For LCSP, address of PRTCB
- For LCSS, address of SAPCB
- For LCSX, address of XCNCB

10–13 Requesting module abbreviated name

14–17 Data length

18–1B Data address

1C FSM state

For LCSL:

00 Non-switched/switched reset
01 Non-switched add SAP User
02 Non-switched delete SAP user
03 Non-switched pending deallocation
04 Non-switched link active
05 Non-switched open station
06 Non-switched test local
07 Non-switched test global
08 Non-switched negotiable
09 Non-switched primary
0A Non-switched connect requested
0B Non-switched pending connect indication
0C Non-switched station active
0D Non-switched close station
81 Switched add SAP user
82 Switched delete SAP user
83 Switched pending deallocation
84 Switched link active
85 Switched inbound answer any
86 Switched inbound poll pending
87 Switched inbound XID pending
88 Switched inbound pending contact
89 Switched inbound open station
8A Switched outbound answer specific
8B Switched outbound open station
8C Switched outbound test local
8D Switched outbound test global
8E Switched outbound XID pending
8F Switched outbound XID3 pending
90 Switched outbound pending contact
91 Switched pending abandon connection
92 Switched pending abandon connection out
93 Switched pending abandon connection, no station
94 Switched pending abandon connection out, no station
95 Switched close station
96 Switched answer off
97 Switched negotiable 1
98 Switched negotiable 2

99 Switched negotiable 3
9A Switched primary
9B Switched secondary
9C Switched connect requested
9D Switched pending connect indication
9E Switched station active
9F Switched inbound XID negotiation-preceding pending

For LCSP
00 Reset
01 Add channel user pending
02 Adapter enable pending
03 Active
04 Adapter disable pending
05 Delete channel user pending
06 Inoperative
07 Blocked

For LCSS:
00 Reset
01 Add port user pending
02 Add SRM entry pending
03 Activate SAP pending
04 Active
05 Deactivate SAP pending
06 Delete SRM entry pending
07 Delete port user pending
08 Inoperative
09 Blocked

For LCSX:
00 Reset state
01 X-side I/O pending
02 Y-SIDE I/O pending
03 Contact wait
04 Contact wait DCM
05 XID7 1 pending
06 XID7 2 pending
07 XID bad pending
08 XID7 1 I/O pending
09 XID7 2 I/O pending
0A XID bad I/O pending
0B Ready
0C Discontact scheduled
0D HDV pending
0E HDV pending DCM
0F Discontact pending

1D Reason code
00 Successful
04 Exit PAB immediately
08 Entire primitive length is not valid
0C Buffer allocation failed
10 Length of primitive header is not valid
14 Length of data field is not valid

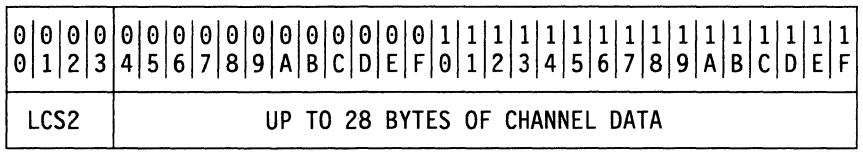
18	Incorrect target layer identifier
1C	Incorrect identifier type
24	Correlator ID mismatch
28	Incorrect parameter list version
2C	Primitive received in wrong state
30	Unrecognized primitive code
34	Incomplete primitive header
38	User_sap_id provided is not valid
58	Length of routing vector (if included) is not valid
5C	XID command/response value is not valid
60	Poll/final indicator is not valid
64	Unexpected XID type
68	Negative confirm received
6C	User class is not valid
70	Flow action value is not valid
74	SAP not net manager capable
78	Not valid for net manager SAP
7C	Vector length field is not valid in adapter-specific overlay
80	Length of adapter-specific overlay field is not valid
84	Vector identifier field is not valid
88	Destination MAC/Destination SAP reporting field is not valid
8C	Identifier (user_sap_id or user_cep_id) inconsistent with identifier type
94	Not valid for logon type
98	Test/XID remote MAC/remote SAP does not match an entry in the SAP user matrix (SUM), and SUM has no "answer any" entries available
9C	Pacing fields not zero
A0	Success count not X'FF'
A4	user_cep_id value not in SAP user matrix (SUM) range or SUM entry not in use or answer mode not specific for SUM entry
A8	Adapter inactive
AC	No match in adapter routing table found or incorrect LAN type
B0	XCNCB 8 byte header error
B4	XCNCB byte count mismatch
B8	XCNCB bad primitive offset
1E–1F	Zero.

LCS2 Entry for LAN Channel Station Error (Part 2)

Entry:	**LCS2**
VIT Option:	LCS
Event:	LAN channel station error (Part 2)
VIT Processing Module:	ISTRACLS
Applies to:	MVS VM VSE

This trace record is a continuation of the LCSL, LCSP, LCSS, or LCSX entry.

The VTAM internal trace will generate as many as 31 LCS2 records for the LCSL, LCSP, LCSS, and LCSX entries.



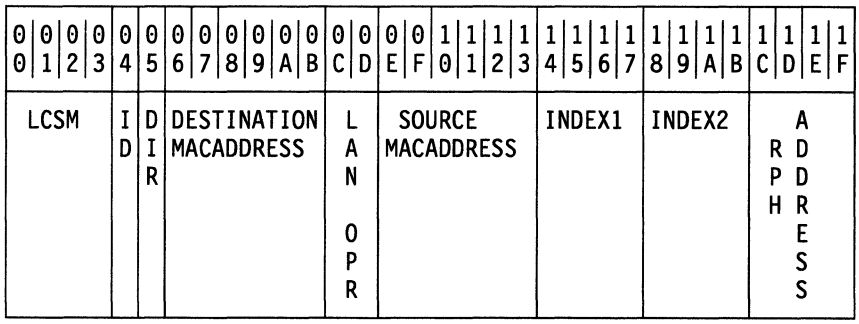
Byte (hex)	Contents
00—03	Record ID: C'LCS2'
04—1F	Up to 28 bytes of channel data.

LCSM Entry for MACaddress for DATA SEND/RECEIVE

Entry: **LCSM**
VIT Option: LCS
Event: MACaddress for DATA SEND/RECEIVE
VIT Processing Module: ISTRACLS
Applies to: MVS VM VSE

If the LCS trace option is specified, an LCSM trace record is created for *every* data frame VTAM sends/receives from an IBM 3172 Interconnect Controller.

The LCSM trace record is created in association with an LCSX entry. However, because these trace entries are created by different component elements, the entries may be separated in the trace by other component traces. For outbound data the LCSM entry will appear before the LCSX entry. For inbound data the LCSM entry will appear following the LCSX entry.



Byte (hex) Contents

00–03 Record ID:C'LCSM' for information about macaddress

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Direction

Code	Meaning
C'I'	Data received inbound
C'O'	Data sent outbound

06–0B Destination macaddress

0C–0D LAN Operation

Note: It is not required that users of the product know the meaning of this internal VTAM name. When required, the product support organization may use it to assist in internal flow diagnosis.

0E–13 Source MACaddress

14–17 Index 1

Note: It is not required that users of the product know the meaning of this internal VTAM name. When required, the product support organization may use it to assist in internal flow diagnosis.

18–1B Index 2

Note: It is not required that users of the product know the meaning of this internal VTAM name. When required, the product support organization may use it to assist in internal flow diagnosis.

1C–1F RPH address.

LICx Entry for LAN Interface Command

Entry:	**LICI, LICO, or LICS**
VIT Option:	CIO
Event:	LAN interface command
VIT Processing Module:	ISTRAC6T
Control is Returned to:	ISTTSC6R for LICI ISTTSC6W for LICO ISTTSC6K or ISTTSC6R for LICS
Applies to:	VM VSE

One of these trace records is created when a command is sent to or received from the token-ring subsystem.

An LICI record is written when a command is received from the subsystem. This entry always corresponds to a previous ADPA or ADPI entry. Use the TSCB address to correlate these entries.

An LICO record is written when a command is ready to be sent to the subsystem. If the command is transferring a PIU onto the ring, it corresponds to a previous PIU entry. The TSCB continuation address of the LICO entry correlates to the TSCB address of the PIU entry.

An LICS record is written when a special condition related to controlling the subsystem is detected.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
LICI LICO				ID	CUA			TSCB ADDRESS				NCB ADDRESS				CBID	CMD			TFSMS		VFSMS			FLAGS			TSCB CONT ADDRESS			
LICS				ID	CUA			TSCB ADDRESS				NCB ADDRESS				CBID	CMD			TFSMS		REASON									

Byte (hex)	Contents
00–03	Record ID: C'LICI' for an inbound record C'LICO' for an outbound record C'LICS' for a special record
04	ID is one of the following: • VM Machine ID and Task ID • VSE Task ID
05–07	Channel device name in EBCDIC (either a CUA or device number)
08–0B	TSCB address
0C–0F	NCB address
10	Control block ID of NCB 0E NCB for local station (VTAM) 0B NCB for remote station

11–13 Command (see "Command Values Reported in Trace Entries (VM) (VSE)" on page 547 for more information about this field)

14 LLC port FSM (TFSMS)
- 01 Logon pending
- 02 Logon completion
- 03 Logoff pending
- 04 Enable pending
- 05 Disable pending

15 Channel program FSM (TFSMS)
- 00 Reset
- 01 Sense ID
- 02 Interrupt port
- 03 Control port
- 04 Running
- 05 Sense
- 06 Orderly halt (HLTC)
- 07 Termination
- 08 Shutdown

For LICI or LICO:

16 LLC link FSM if CBID=X'0B' (VLNLFSM)
- 00 Reset
- 10 ACTLINK pending
- 20 Active
- 30 DACTLINK pending

17 LLC connection FSM if CBID=X'0B' (VLNCFSM)
- 00 Reset
- 10 XID0 pending
- 20 Connect out
- 30 Connected
- 40 XID pending
- 70 Determined primary
- 80 Non-activation XID3 pending

18 LLC station FSM if CBID=X'0B' (VLNSFSM)
- 00 Reset
- 10 XID zero pending
- 20 Set mode pending
- 30 Set mode wait
- 40 Active state
- 50 Request discontact
- 60 Disconnect pending

19–1B Remote station (VLNCB) flags if CBID=X'0B'
- 19 VLNFLAG1
- 1A VLNFLAG2
- 1B VLNFLAG3

1C–1F TSCB continuation address

For LICS:

16–1F Special reason text:

C'CLEAN UP' VTAM has completed shutdown of the token-ring subsystem. This entry is preceded by a LICS C'FINAL TERM' entry.

C'DISCARD' Subsystem input has been discarded.
C'FINAL TERM' Channel programs to the subsystem have been stopped.
C'HALT RQST' VTAM is initiating shutdown of the subsystem.

LKEX Entry for TPLOCK Exclusive

Entry:	**LKEX**
VIT Option:	LOCK
Event:	TPLOCK exclusive
VIT Processing Module:	ISTRACLK
Control is Returned to:	ISTAPC36
Applies to:	MVS VM VSE

This trace record identifies a request for exclusive control of a VTAM lock and the current status of a lock. Exclusive locks can be held by only one VTAM process. If a WAIT entry immediately follows this entry, it *could* be because a wait-for-lock condition exists. Look at the lockword in the trace output to see if the lock is already held.

Locks obtained by LKEX are eventually released by UNLK or ULKA. See Table 7 on page 80 for a list of VTAM locks.

00 01 02 03	04	05	06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17 18 19 1A 1B	1C 1D 1E 1F
LKEX	ID	0	LOCK LEVEL	LOCK ADDRESS	CRA LKACT	RETURN ADDRESS	LOCKWORD	RPH ADDRESS

Byte (hex)	Contents
00–03	Record ID: C'LKEX'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Zero
06–07	Level of lock to be obtained
08–0B	Address of lock to be obtained
0C–0F	CRA lock account word (CRALKACT; see step 11 on page 79)
10–13	Address of the issuer of the TPLOCK macro
14–1B	Lockword pointed to by address in bytes 08-0B
1C–1F	Request parameter header (RPH) Address.

LKSH Entry for TPLOCK Shared

Entry:	**LKSH**
VIT Option:	LOCK
Event:	TPLOCK shared
VIT Processing Module:	ISTRACLK
Control is Returned to:	ISTAPC35
Applies to:	MVS VM VSE

This trace record identifies a request for a shared VTAM lock and the current status of a lock. Shared locks can be held by more than one VTAM process at a time. If a WAIT entry immediately follows this entry, it may be because a wait-for-lock condition exists. Look at the lockword in the trace output to see if the lock is already held.

Locks obtained by LKEH are eventually released by UNLK or ULKA. See Table 7 on page 80 for a list of VTAM locks.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
LKSH				ID	STATUS	LOCK	LEVEL	LOCK ADDRESS				CRA LKACT				RETURN ADDRESS				LOCKWORD								RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'LKSH'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Indicates lock wait status: X'00' WAIT X'80' NOWAIT
06–07	Level of lock to be obtained
08–0B	Address of lock to be obtained
0C–0F	CRA lock account word (CRALKACT; see step 11 on page 79)
10–13	Address of the issuer of the TPLOCK macro
14–1B	Lockword pointed to by address in bytes 08-0B
1C–1F	Request parameter header (RPH) Address.

LNKx Entry

Entry:	**LNKA, LNKB, LNKH, LNKI, LNKL, LNKV, or LNKX**
VIT Option:	CIO
Event:	Change to link use count
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTPUCFL, ISTPUCGA, ISTTSCAF, ISTTSCDN, or ISTTSCQA
Applies to:	MVS VM VSE

This trace record is written when the link use count in the node control block (NCB) is incremented or decremented.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
LNKA LNKB LNKH LNKI LNKL LNKV LNKX				ID	REASON	EALEDMRENTS		LINK USE COUNT				NCB ADDRESS				ADDRESS OF CONTROL BLOCK				0				MODULE or CALLER ADDRESS				0			

Byte (hex) Contents

00–02 Record ID: C'LNK'

03 NCB type:
- C'LNKA' for AHNCB
- C'LNKB' for BSCNCB VM VSE
- C'LNKH' for HALCB VM VSE
- C'LNKI' for ICNCB
- C'LNKL' for LDNCB
- C'LNKV' for VLNCB VM
- C'LNKX' for XCNCB

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Reason for change to use count (DNPFUNC):
- X'01' Increment for boundary SLU element address (BLB)
- X'02' Increment for boundary PLU element address (BPAT)
- X'03' Increment for boundary SNA half-session (BSB)
- X'04' Increment for boundary non-SNA half-session (LUST)
- X'05' Increment for application half-session (FMCB)
- X'F1' Decrement for boundary SLU element address (BLB)
- X'F2' Decrement for boundary PLU element address (BPAT)
- X'F3' Decrement for boundary SNA half-session (BSB)
- X'F4' Decrement for boundary non-SNA half-session (LUST)
- X'F5' Decrement for application half-session (FMCB)
- X'F6' Decrement for line (NCB)

06–07 Element address (for boundary element address reasons)

08–0B Link use count after the increment or decrement

0C–0F Node control block (NCB) address

10–13	Address of control block associated with the increment or decrement
14–17	Zero
18–1B	Module address or address of the issuer of the TSCDN caller
1C–1F	Zero.

LOST Entry for Lost Trace Record

Entry:	**LOST**
VIT Option:	None
Event:	Lost trace record
VIT Processing Module:	ISTRACRR
Control is Returned to:	The location shown in register 14
Applies to:	MVS VM VSE

This trace record indicates that the internal trace table is incomplete. Information from events that were supposed to be traced was lost because there were not enough save areas available for the tracing routine to record the information.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
LOST				ID	0			REG 14				'LOST TRACE RECORD'																			

Byte (hex)	Contents
00–03	Record ID: C'LOST'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05–07	Zero
08–0B	Register 14 (return address)
0C–1F	C'LOST TRACE RECORD'.

LSNA Entry for Local SNA Record (Part 1)

Entry:	**LSNA**
VIT Option:	CIO
Event:	Read/Write from APPN PU Channel
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSC86
Applies to:	MVS VM VSE

This trace record is written each time data is read or written across an APPN host-to-host channel.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
LSNA				ID	CB ID	FLAGS	TYPE	TSCB ADDRESS				DAT LEN		18 BYTES OF LOCAL SNA DATA																	

Byte (hex) Contents

00–03 Record ID: C'LSNA'

04 ID is one of the following:

- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Control block ID of work element being written or read across the channel

06 Event Flags

Bit	Meaning
1...	The data unit being traced (FID2 data or control data) is the last data unit in the multipath channel (MPC) transmit block header just received. This bit only has meaning for inbound data.
.1..	This data is control data. If off, this data is normal FID2 data traffic.
..1.	This trace entry contains the beginning of a new PIU/control data. If off, this data unit is the middle or end portion of a PIU. For outbound data, this bit will always be on (only the first part of PIU/Control data outbound is traced).

07 A 1-byte field that indicates the direction of the data unit flow:

Code	Meaning
C'I'	Data received inbound
C'O'	Data sent outbound

08–0B TSCB address

0C–0D Length of data being sent or received
0E–1F This 18-byte field will map as follows:

- For control data
  - The 2-byte QLLC Header (indicating the type of the control data)
  - Up to 16 bytes of the remaining control data
- For FID2 data
  - The 6-byte FID2 TH header
  - A 3-byte RH if it exists
  - Up to 9 bytes of the remaining FID2 data (12 bytes if no RH).

LSN2 Entry for Local SNA Record (Part 2)

Entry: **LSN2**
VIT Option: CIO
Event: Read/Write from APPN Host PU Channel
VIT Processing Module: ISTRACCI
Applies to: MVS VM VSE

This trace record is a continuation of the LSNA entry. It contains 28 more bytes of data.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
LSN2				28 MORE BYTES OF LOCAL SNA DATA																											

Byte (hex)	Contents
00–03	Record ID: C'LSN2'
04–1F	Up to 28 more bytes of data.

MIC Entry for MVS IUCV Macro Completion

Entry:	**MIC**
VIT Option:	TCP
Event:	When a macro issued across the MVS-simulated IUCV API completes
VIT Processing Module:	ISTRACIP
Control is Returned to:	ISTIPCII or ISTIPCIS
Applies to:	MVS

This entry is issued when a macro that was issued to the MVS simulated IUCV interface completes.

The trace record indicates which macro was issued and which IUCV path it was issued on. It also points to the parameter list used on the macro and shows the completion audit fields from the macro.

00	01	02	03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11	12 13	14 15 16 17 18 19 1A 1B	1C 1D 1E 1F
MIC		ID	IUCV TYPE	IUCV RC	0	SOCKET MAJOR CONTROL BLOCK ADDRESS	IUCV PARM LIST ADDRESS	IUD CV PATH	AFUL DAIG TS OR 0	0			RPH ADDRESS

Byte (hex)	Contents
00–03	Record ID: C'MIC'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	IUCV external interrupt type: X'01' Pending connection X'02' Connection complete X'03' Path has been severed X'04' Path has been quiesced X'05' Path has been resumed X'06' Priority message completion pending (REPLY) X'07' Non-priority message completion pending (REPLY) X'08' Priority message pending X'09' Non-priority message pending
06	Return code from issuing the IUCVMINI SET macro. Otherwise, this field contains a zero.
07	Zero
08–0B	Socket major control block address
0C–0F	IUCV IPARML parameter list address
10–11	IUCV Path ID
12–13	Audit flags or zero
14–1B	Zero
1C–1F	Request for parameter header (RPH) address.

MIE Entry for MVS IUCV External Interrupt

Entry:	**MIE**
VIT Option:	TCP
Event:	When MVS IUCV drives the external interrupt exit to signal the completion of an MVS IUCV macro
VIT Processing Module:	ISTRACIP
Control is Returned to:	ISTIPCIE
Applies to:	MVS

This trace record is written when MVS IUCV drives the external interrupt exit to signal the completion of an MVS IUCV macro.

The trace record copies the first 27 bytes of the external interrupt buffer.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
MIE				ID	FIRST 27 BYTES OF EXTERNAL INTERRUPT EXIT																											

Byte (hex)	Contents
00–03	Record ID: C'MIE'
04	ID is the primary address space !D (ASID). This field is zero if the ASID is greater than X'FF'.
05–1F	First 27 bytes of the External Interrupt Exit.

MII Entry for IUCV API Call

Entry:	**MII**
VIT Option:	TCP
Event:	IUCV API Call
VIT Processing Module:	ISTRACIP
Control is Returned to:	ISTIPCII or ISTIPCIS
Applies to:	MVS

This trace record is written when a VTAM module issues an IUCV API call.

The trace record indicates which macro is issued and which IUCV path it is issued on. It also points to the parameter list used on the macro and shows the return code from issuing the macro.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
MII				ID	MACRO TYPE	RETURN CODE		0	SOCKET MAJOR CONTROL BLOCK ADDRESS			IUCV PARM LIST ADDRESS				IUCV PATH ID		0										RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'MII'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	Macro type: 1 IUCVMINI SET 2 IUCVMCOM CONNECT 3 IUCVMCOM SEND 4 IUCVMCOM SEVER 5 IUCVMINI CLEAR 6 IUCVMCOM SEVER(ALL)
06	Return code from issuing macro
07	Zero
08–0B	Socket major control block address
0C–0F	IUCV IPARML (ISTIPARM) address
10–11	IUCV path ID (ISTMIPTH)
12–1B	Zero
1C–1F	Request parameter header (RPH) Address.

MPT Entry for IP Component Read or Write (Part 1)

Entry: **MPT**
VIT Option: TCP
Event: IP component reads in or writes out MPTN format
VIT Processing Module: ISTRACIP
Control is Returned to: Calling module (ISTIPCRP, ISTIPCWR, ISTIPCDR, or ISTIPCDW)
Applies to: MVS

This trace record is written when an MPTN format is read in or written out by the IP component. It contains information about the MPTN format being traced.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
MPT				ID	FLAGS	MPTN ID	MIG OR 0	MPTN FORMAT LENGTH				TSCB ADDRESS				RETURN ADDRESS OF CALLER				0								RPH ADDRESS			

Byte (hex) Contents

00–03 Record ID: C'MPT'

04 ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.

05 Flag field (TSCFLAG1)

06 MPTN format ID
- X'00' Header: No associated compensation
- X'01' Header: Expedited data
- X'03' Header: Expedited-data acknowledgement
- X'10' Header: Duplex-abortive termination
- X'20' Header: Segmented record
- X'80' MPTN connect
- X'82' MPTN datagram probe
- X'83' MPTN out-of-band datagram
- X'84' MPTN KEEPALIVE datagram

07 If an MPTN command is being traced, the MTN_COMM_MIG_SPEC field is traced in this field. Otherwise, the field is zero.

08–0B MPTN format length

0C–0F Address of transmission subsystem control block (TSCB) chain containing the MPTN format

10–13 Return address of module issuing INTRACE

14–1B Zero

1C–1F Request parameter header (RPH) address.

MPT2 Entry for IP Component Read or Write (Part 2)

Entry: **MPT2**
VIT Option: TCP
Event: IP component reads in or writes out MPTN format
VIT Processing Module: ISTRACIP
Applies to: MVS

This trace record is a continuation of the MPT entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
MPT2					28 BYTES OF THE MPTN FORMAT																										

Byte (hex)	Contents
00–03	Record ID: C'MPT2'
05–1F	28 bytes of the MPTN format.

MSG Entry for Message Issued

Entry:	**MSG**
VIT Option:	MSG
Event:	Message issued (Part 1)
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTCFCTM
Applies to:	MVS VM VSE

This trace record is written each time a message is issued by a VTAM module. You can use it to find the originator of a message and to see whether the message was solicited or unsolicited.

MSG trace entries are generated for all messages, even those that are suppressed by message-flooding prevention (described in Appendix B, "Message Flooding Prevention" in *VTAM Messages and Codes*), by the MODIFY SUPP command (described in "MODIFY SUPP Command" in *VTAM Operation*), or by the SUPP start option (described in "SUPP" in the *VTAM Resource Definition Reference*). MSG trace entries thus provide information that may be missing from the operator's console.

The VTAM Internal Trace (VIT) does not trace the following messages:

- MVS Logon manager messages (beginning with ELM)
- MVS Sockets-over-SNA messages (beginning with ISU)
- MVS TSO/VTAM messages (beginning with IKT)
- VM VSCS messages (beginning with DTI).

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
MSG				ID	0	S or U	MSG ID OR 'USS'				MODULE ID					SAVE AREA ADDRESS				RETURN ADDRESS				DESTINATION OR 0								

Byte (hex) Contents

00–03 Record ID: C'MSG'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Zero

06 'S' if the message was solicited (issued in response to a VTAM command); 'U' if the message was unsolicited

07–0A Message ID or 'USS'. If USS, the message ID appears in the variable text field in the MSG2 entry. If message ID, the 4-digit message number (or 3-digit message number and 1-character type code) appears in this field.

0B–0F Issuing module ID (4th, 5th, 6th, 7th, and 8th characters of the issuing module name). See "Module Names in Internal Trace Records" on page 405 for information on determining the module name from the module ID.

10–13 Save area address

14–17 Address of the CPMSG macro

- If the high order bit in the return address field is on, CPMSG has been issued from a utility module. The module name and return address are that of the caller of the utility module.
- If the high order bit in the return address field is off, CPMSG has not been issued from a utility module. The module name and return address are that of the module that issued the CPMSG.

18–1F Destination

- If byte 06 is **S**, this field contains one of the following:

  SYSTEMxx — *xx* is the ID of the system console.

  NLDM — Message is destined for the session monitor component of the NetView program or NLDM.

  URC — Message is to be sent to the system console using routing codes, instead of to one console in particular.

  POA — Message is to be sent to a program operator application (POA) and the POA name is not available yet.

- If byte 06 is **U**, this field is zero.

MSGS Entry for Message Sent (Part 1)

Entry:	**MSGS**
VIT Option:	MSG
Event:	Message sent (Part 1)
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTLUCRC or ISTMSCCN
Applies to:	MVS VM VSE

This trace record is written each time VTAM sends a message to its destination. It can be used to correlate the console log with the internal trace. The destination may indicate that the message was sent to a system console (via a WTO macro) or to a program operator application, such as the NetView program (via a RCVCMD macro).

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
MSGS				ID	0			DESTINATION								MESSAGE HEADER				FIRST 12 CHARACTERS OF MESSAGE TEXT											

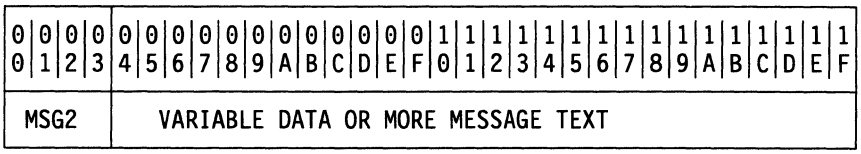
Byte (hex)	Contents
00–03	Record ID: C'MSGS'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05–07	Zero
08–0F	Destination. If 'SYSTEMxx', the message was sent to system console 'xx', via a WTO macro. If an application name, the message was sent to a program operator application, via a RCVCMD macro.
10–13	Message header (in hex). If destination is 'SYSTEMxx', this will be the WPL header. If destination is a program operator application, this will be a program operator header (ISTDPOHD).
14–1F	First 12 characters of the message text. If the message text is longer than 12 characters, it will continue in the MSG2 entry.

MSG2 Entry for Message Sent (Part 2)

Entry:	**MSG2**
VIT Option:	MSG
Event:	Message issued or message sent (Part 2)
VIT Processing Module:	ISTRACOT
Applies to:	MVS VM VSE

This trace record is a continuation of the MSG or MSGS trace record. It contains variable data for the MSG entry, or more message text for the MSGS entry.

Up to two MSG2 entries can follow the MSG entry; however, only one MSG2 entry can follow the MSGS entry.



Byte (hex)	Contents
00–03	Record ID: C'MSG2'
04–1F	Variable data, or more message text.

MU1 Entry for LU 6.2 Message Unit (Part 1)

Entry:	**MU1**
VIT Option:	APPC
Event:	LU 6.2 message unit (Part 1)
VIT Processing Module:	ISTRACAC
Control is Returned to:	Module invoking the INTRACE macro that caused the record to be produced.
Applies to:	MVS VM VSE

This trace record shows information about an LU 6.2 message unit. It is generated whenever a message unit is queued to another component.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
MU1				ID	MU ID			MU ADDRESS				HALF SESS ID				CORR VALUE				RTNCD	0			AMU FLAGS				SENSE DATA OR 0			

Byte (hex) Contents

00–03 Record ID: C'MU1'

04 ID is one of the following:

- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05–07 Message unit ID

X'010101' (END_CONVERSATION)
X'010102' (SESSION_FLOW_RESUMED)
X'010103' (FLUSH_DATA)
X'010201' (PS_COPR_FMH5_RCVD)
X'010301' (SEND_EXPEDITED_DATA)
X'010302' (CONFIRMED)
X'010303' (REQUEST_TO_SEND)
X'010304' (SEND_DATA_RECORD)
X'010305' (SEND_ERROR)
X'010306' (SEND_PACING_RSP)
X'010307' (RSP_TO_EXPEDITED_DATA)
X'010501' (ALLOCATE_RCB)
X'010502' (DEALLOCATE_RCB)
X'010503' (GET_SESSION)
X'010504' (UNBIND_PROTOCOL_ERROR)
X'010505' (REJECT_SESSION)
X'010506' (SUSPEND_SESSION)
X'010507' (RESUME_SESSION)
X'010701' (ATTACH_TP)
X'010702' (END_XP)
X'020101' (CNOS_ABORT)
X'020201' (CNOS_CLEANUP)
X'020501' (CNOS_COMPLETE)

X'020502' (CHANGE_SESSIONS)
X'030101' (RECEIVE_EXPEDITED_DATA)
X'030102' (CONFIRMED)
X'030103' (REQUEST_TO_SEND)
X'030104' (RECEIVE_DATA)
X'030105' (RECEIVE_ERROR)
X'030106' (PACING_RSP_RCVD)
X'030107' (RSP_TO_REQUEST_TO_SEND)
X'030108' (INITIAL_PACING_COUNT)
X'030109' (DEALLOCATE_ABEND_REJECTED)
X'030401' (ABORT_HS)
X'030501' (ATTACH_HEADER)
X'030502' (FREE_SESSION)
X'030503' (BID)
X'030504' (BID_RSP)
X'030505' (BIS_RQ)
X'030506' (BIS_REPLY)
X'030507' (RTR_RQ)
X'030508' (RTR_RSP)
X'030509' (SECURITY_HEADER)
X'040101' (RESTORE_SESSION)
X'040401' (LOAD_PROFILES)
X'040402' (RSP_LOAD_PROFILES)
X'040403' (DELETE_PROFILES)
X'040501' (SESSION_ACTIVATED)
X'040502' (SESSION_DEACTIVATED)
X'040503' (ACTIVATE_SESSION_RSP)
X'040504' (CTERM_DEACTIVATE_SESSION)
X'040505' (LNS_LRM_FREE_AMU)
X'050101' (RCB_ALLOCATED)
X'050103' (SESSION_ALLOCATED)
X'050104' (ATTACH_RECEIVED)
X'050105' (CONVERSATION_FAILURE)
X'050106' (SESSION_REJECTED)
X'050107' (SESSION_SUSPENDED)
X'050108' (SESSION_RESUMED)
X'050109' (RESUME_SESSION_FLOW)
X'050301' (HS_PS_CONNECTED)
X'050302' (YIELD_SESSION)
X'050303' (BID_WITHOUT_ATTACH)
X'050304' (BID_RSP)
X'050305' (BIS_RQ)
X'050306' (BIS_REPLY)
X'050307' (RTR_RQ)
X'050308' (RTR_RSP)
X'050309' (ENCIPHERED_RD2)
X'050401' (ACTIVATE_SESSION)
X'050402' (DEACTIVATE_SESSION)
X'060201' (MODIFY_CNOS)
X'060202' (MODIFY_DEFINE)
X'060203' (DISPLAY_CNOS)
X'060204' (DISPLAY_LUS)
X'060205' (DISPLAY_MODES)

	X'060206' (DISPLAY_CONVS)
	X'060401' (MODIFY_PROFILES)
	X'070101' (TERMINATE_TP)
	X'070301' (SEND_1WAY_FASTPATH)
08–0B	Message unit address
0C–0F	Half session ID
10–13	Address of correlator value
14	LU 6.2 message unit return code (AMURETCD)
15–17	Zero
18–1B	LU 6.2 message unit flags (AMUFLAGS)
1C–1F	LU 6.2 message unit sense data (AMUSENSE) or zero.

MU2 Entry for LU 6.2 Message Unit (Part 2)

Entry: **MU2**
VIT Option: APPC
Event: LU 6.2 message unit (Part 2)
VIT Processing Module: ISTRACAC
Applies to: MVS VM VSE

This trace record is a continuation of the MU1 trace record.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
MU2				0	MU ID			DATA ADDRESS OR 0				DATA LENGTH OR 0				RAB ADDRESS OR 0				SES CNT OR 0		SES LIM OR 0		CONV ID OR 0				VARI-ABLE OVERLAY FIELD			

Byte (hex) Contents

00–03 Record ID: C'MU2'
04 Zero
05–07 Message unit ID shown in MU1 entry
08–0B Data address or zero
0C–0F Data length or zero
10–13 Resource allocation block address or zero
14–15 For MU ID X'020502': Session count
For all other MU IDs: Zero
16–17 For MU ID X'020502': Session limit
For all other MU IDs: Zero
18–1B For MU IDs associated with an active conversation (X'010102', X'010201', X'010506', X'030108', X'050107', X'050109', X'050301'): Conversation ID
For all other MU IDs: Zero
1C–1F Variable overlay field

For MU ID X'020201': COPR control block address

For MU ID X'020502':
1C–1D Change in the CNOS value
1E–1F Zero

For MU IDs X'030106' and X'030108': Pacing count increment

For MU IDs X'030501', X'030503', X'030504', and X'030507': RPH address for inbound half session

For MU IDs X'040402' and X'040403': ACEE address

For MU IDs X'010701' Address of the associated data for the MU.

For all other MU IDs: Zero.

MU3 Entry for LU 6.2 Message Unit (Part 3)

Entry: **MU3**
VIT Option: APPC
Event: LU 6.2 message unit (Part 3)
VIT Processing Module: ISTRACAC
Applies to: MVS VM VSE

This trace record is a continuation of the MU2 trace record. It is generated for only those message units which contain the following MU IDs:

X'010102' X'040403' X'060202'
X'010201' X'040501' X'060203'
X'010501' X'050106' X'060204'
X'010505' X'050107' X'060205'
X'010506' X'050108' X'060206'

X'020502' X'050109' X'060401'
X'030509' X'050309'
X'040401' X'050401'
X'040402' X'060201'

0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	0 A	0 B	0 C	0 D	0 E	0 F	1 0	1 1	1 2	1 3	1 4	1 5	1 6	1 7	1 8	1 9	1 A	1 B	1 C	1 D	1 E	1 F
MU3				0	MU ID			MODE NAME OR SESSION ID OR 0								LOCAL LU NAME								PARTNER LU NAME							

Byte (hex) Contents

00–03 Record ID: C'MU3'

04 Zero

05–07 Message unit ID shown in the MU1 and MU2 entries

08–0F Mode name for the following MU IDs:

X'020502'
X'040501'
X'050401'
X'060202'
X'060203'

Mode name or zero for the following MU IDs:

X'010501'
X'060201'
X'060205'
X'060206'

Session instance identifier for the following MU IDs:

X'010102'
X'010506'
X'010507'
X'050106'
X'050107'
X'050108'
X'050109'

	Session instance identifier or zero for the following MU ID: X'010505'
	Zero for all other MU IDs
10–17	Local logical unit name
18–1F	Partner logical unit name or zero; for X'060204', zero.

MU4 Entry for LU 6.2 Message Unit (Part 4)

Entry:	**MU4**
VIT Option:	APPC
Event:	LU 6.2 message unit (Part 4)
VIT Processing Module:	ISTRACAC
Applies to:	MVS VM VSE

This trace record is a continuation of the MU3 trace record. It is generated for only those message units which contain the following MU IDs:

X'010102' X'040403' X'060202'
X'010201' X'040501' X'060203'
X'010501' X'050106' X'060204'
X'010505' X'050107' X'060205'
X'010506' X'050108' X'060206'
X'020502' X'050109' X'060401'
X'030509' X'050309'
X'040401' X'050401'
X'040402' X'060201'

0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	0 A	0 B	0 C	0 D	0 E	0 F	1 0	1 1	1 2	1 3	1 4	1 5	1 6	1 7	1 8	1 9	1 A	1 B	1 C	1 D	1 E	1 F
MU4				0																				PARTNER LU NETID OR 0							

Byte (hex) Contents

00–03	Record ID: C'MU4'
04–17	Zero
18–1F	Partner network identifier or zero; for X'060204', zero.

NMPT Entry for an Exception Condition(Part 1)

Entry:	**NMPT**
VIT Option:	TCP
Event:	Negative MPTN format or duplex-abortive header is generated in response to an exception condition
VIT Processing Module:	ISTRACIP
Control is Returned to:	Module invoking the INTRACE macro that caused the record to be produced.
Applies to:	MVS

This trace record is written when a negative MPTN format or duplex-abortive header is generated in response to an exception condition. It contains information about the MPTN format being traced.

This record is treated as an exception entry and is always traced regardless of the VIT options specified.

00	01	02	03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
NMPT		ID	FLAGS	MPTN ID	MIG OR 0	MPTN FORMAT LENGTH		SENSE DATA OR 0	RETURN ADDRESS OF CALLER	PRIMARY RETURN CODE OR 0	SECOND RETURN CODE OR 0	RPH ADDRESS	

Byte (hex) Contents

00–03 Record ID: C'NMPT'

04 ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.

05 Flag field (TSCFLAG1)

06 MPTN format ID
- X'10' Header: Duplex-abortive termination
- X'80' MPTN connect
- X'82' MPTN datagram probe
- X'83' MPTN out-of-band datagram
- X'84' MPTN KEEPALIVE datagram

07 If an MPTN command is being traced, the MTN_COMM_MIG_SPEC field is traced in this field. Otherwise, the field is zero.

08–0B MPTN format length

0C–0F If present, this field contains sense data from the UNBIND or negative BIND response. Otherwise, the field is zero.

10–13 Return address of module issuing INTRACE

14–17 If present, this field contains the primary return code from the diagnostic vector. Otherwise, the field is zero.

18–1B If present, this field contains the secondary return code from the diagnostic vector. Otherwise, the field is zero.

1C–1F Request parameter header (RPH) address.

NMP2 Entry for Exception Condition (Part 2)

Entry: **NMP2**
VIT Option: TCP
Event: Negative MPTN format or duplex-abortive header is generated in response to an exception condition
VIT Processing Module: ISTRACIP
Applies to: MVS

This trace record is a continuation of the NMPT entry and will contain the actual MPTN format. VTAM will generate as many NMP2 records as necessary to trace up to 868 bytes of the MPTN format.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
NMP2					28 BYTES OF THE MPTN FORMAT																										

Byte (hex)	Contents
00–03	Record ID: C'NMP2'
04–1F	28 bytes of the MPTN format.

NRSP Entry for Negative Response to PIU Request (Part 1)

Entry:	**NRSP**
VIT Option:	PIU
Event:	Negative response to PIU request (Part 1)
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTTSCGR
Applies to:	MVS VM VSE

This trace record is written when VTAM generates a negative response to a PIU request. Because this entry is associated with an event failure, it is treated as an exception condition and is always traced, regardless of the VIT options specified, if the VIT is active.

00	01	02	03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F
NRSP				ID	CBID	FLAGS	0	TSCB ADDRESS	20 BYTES OF PIU

Byte (hex)	Contents
00–03	Record ID: C'NRSP'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Control block ID
06	Flag field (TSCFLAG1)
07	Zero
08–0B	Address of TSCB
0C–1F	First 20 bytes of the PIU.

NRS2 Entry for Negative Response to PIU Request (Part 2)

Entry: **NRS2**
VIT Option: PIU
Event: Negative response to PIU request (Part 2)
VIT Processing Module: ISTRACOT
Applies to: MVS VM VSE

This trace record is a continuation of the NRSP entry.

0 0 0 0 0 1 2 3	0 0 0 0 0 0 0 0 0 0 0 0 4 5 6 7 8 9 A B C D E F	1 1 1 1 1 1 1 1 1 1 1 1 0 1 2 3 4 5 6 7 8 9 A B	1 1 1 1 C D E F
NRS2	24 MORE CHARACTERS OF THE PIU		RETURN ADDRESS

Byte (hex)	Contents
00–03	Record ID: C'NRS2'
04–1B	24 more characters of the PIU
1C–1F	Return address of the issuer.

NSD Entry for NSIND Exit

Entry:	**NSD**
VIT Option:	VCNS
Event:	NSIND exit scheduled
VIT Processing Module:	ISTRACNS
Control is Returned to:	ISTAPCUE
Applies to:	MVS VM

This trace record is written when the NSIND exit routine is scheduled. The exit is scheduled when VTAM receives a VCNSCMD CONTROL=LOGON request from a VCNS user. This trace record shows information about the exit invocation.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
NSD				ID	0			ACB ADDR				NETWORK ACCESS POINT								EXIT REASON				STORAGE ADDRESS				ERROR INFO			

Byte (hex) Contents

00–03 Record ID: C'NSD'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM

05–07 Zero

08–0B Address of the ACB associated with the NSIND exit

0C–13 Symbolic name of the network access point, defined with USER=VCNS in the VM packet major node or the interconnect major node, representing a line to which the application program is logged on
Symbolic name of the network access point, defined with USER=VCNS in the interconnect major node, representing a line to which the application program is logged on

14–17 Reason the exit was scheduled:
- 0 As a result of a network management-related event
- 4 As a result of a termination of the VCNS line
- 8 As a result of an error detected by VTAM which resulted in the termination of the LOGON request (for example, an abend)

18–1B If exit reason in bytes 14–17 is zero, address of storage area containing specific error information about the exit event; otherwise, zero.

1C–1F If bytes 18-1B contain a storage area address, first 4 bytes of specific error information from the storage area.

OPER Entry for Operator Command (Part 1)

Entry:	**OPER**
VIT Option:	MSG
Event:	Operator command (Part 1)
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTLUCMD
	MVS ISTCFF3D, ISTINCF9
Applies to:	MVS VM VSE

This trace record is written each time a VTAM command is issued at the operator console. This can help you match the console log to a surge of activity shown in the VIT. This trace record is also useful when the console log is unavailable. Additionally, it provides a record of POA commands that have been issued. These do not appear on the console log.

The variable data in bytes 08–1F is the command text, with minor modifications. If this data is longer than 24 characters, it will continue in the OPE2 entry. The command always appears in abbreviated form:

D DISPLAY command
F MODIFY command
V VARY command
Z HALT command
? Unknown command passed to VTAM.

NET and procname are excluded from the command text. Because the VTAM internal trace is not active when VTAM is started, the START command is not traced.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
OPER				ID	FLAG	0		FIRST 24 CHARACTERS OF OPERATOR COMMAND																							

Byte (hex) Contents

00–03 Record ID: C'OPER'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Flag byte

Bit	Meaning
1...	A POA command was issued
0...	Not a POA command

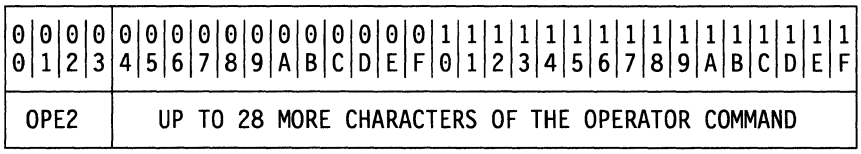
06–07 Zero

08–1F First 24 characters of the operator command.

OPE2 Entry for Operator Command (Part 2)

Entry:	**OPE2**
VIT Option:	MSG
Event:	Operator command (Part 2)
VIT Processing Module:	ISTRACOT
Applies to:	MVS VM VSE

This trace record is a continuation of the OPER entry.

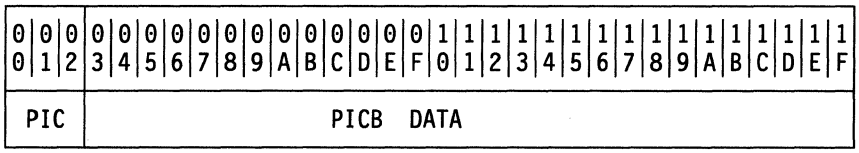


Byte (hex)	Contents
00–03	Record ID: C'OPE2'
04–1F	Up to 28 more characters of the operator command.

PIC Entry for Packet Interface

Entry:	**PIC**
VIT Option:	CIO
Event:	Packet interface
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	ISTTSC5F, ISTTSC5G
Applies to:	VM VSE

This trace record provides information about internal control flows of the telecommunications subsystem or communication adapter.



Byte (hex)	Contents
00–02	Record ID: C'PIC'
03–1F	29 bytes of PICB data.

PIU Entry (Part 1)

Entry:	**PIU**
VIT Option:	PIU
Event:	PIU record (Part 1)
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTLSC6V ISTTSCPR, ISTTSCLS, ISTTSCRI, ISTTSCSC, ISTTSCSR, ISTTSCWS, ISTTSCPD, ISTTSCLE, ISTTSCXS (or ISTINCF1 and INTINCS1 for RUPE)
Applies to:	MVS VM VSE

This trace record provides information about external and internal FID4 PIUs.

For most PIUs, only the first 48 bytes are traced (a 26-byte TH, a 3-byte RH, and the first 19 bytes of RU). However, certain classes of PIUs, such as session control RUs and VTAM RUs, are traced in their entirety.

If the control block is a RUPE, the TH is reconstructed by the internal trace using what information is available. For TSCB entries, the PIU is recorded as is.

If the PIU is a response with sense data, this entry will be generated whether the PIU option is in effect or not. It is treated as an exception condition and is always traced, regardless of the VIT options specified, if the VIT is active.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
PIU				ID	CBID	FLAGS	0	TSCB OR RUPE ADDRESS				20 BYTES OF PIU																			

Byte (hex) Contents

00–03 Record ID: C'PIU'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Control block ID: X'54'=RUPE, X'99'=TSCB

06 Flags (TSCFLAG1) or zero if control block is a RUPE

07 Zero

08–0B TSCB or RUPE address (see byte 5)

0C–1F Twenty bytes of PIU.

PIU2 Entry (Part 2)

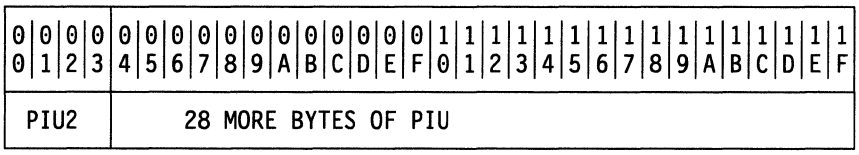
Entry:	**PIU2**
VIT Option:	PIU
Event:	PIU record (Part 2)
VIT Processing Module:	ISTRACOT
Applies to:	MVS VM VSE

This trace record is a continuation of the PIU entry and contains 28 more bytes of FID4 PIU.

The VTAM internal trace will generate as many as 31 PIU2 records in the following situations:

- For PIUs that are formatted and contain session control RUs
- For RUs to or from the SSCP or PUs
- When the VIT LCS option was specified and the PIU contains LCS data.

The number of records generated depends on the data count field in the TH.



Byte (hex)	Contents
00–03	Record ID: C'PIU2'
04–1F	Up to 28 more bytes of the FID4 PIU (padded at right with zeros).

PKTI or PKTO Entry for X.25 Packet Sent or Received (Part 1)

Entry:	**PKTI or PKTO**
VIT Option:	CIO
Event:	X.25 packet sent or received (Part 1)
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	ISTTSC5B
Applies to:	VM VSE

This trace entry provides information about packets received at (PKTI) or sent from (PKTO) an X.25 port.

A PKTI trace record is written whenever the buffer control channel program completes with packets received.

A PKTO trace record is written whenever a packet is transferred to the telecommunications subsystem or communication adapter by inserting it into the buffer control channel program.

Note: If the packet is longer than 20 bytes (including the packet header), up to 27 additional bytes of data are contained in the PKTX trace record.

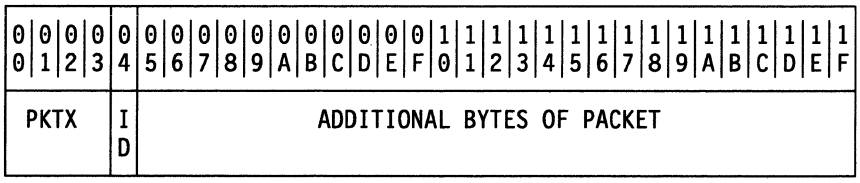
00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
PKTI PKTO				ID	CUA			PFSM	INDX	LNGTH		FIRST 20 BYTES OF THE PACKET																			

Byte (hex)	Contents
00–03	Record ID: C'PKTI' for inbound packet C'PKTO' for outbound packet
04	ID is one of the following: • VM Machine ID and Task ID • VSE Task ID
05–07	Channel device name in EBCDIC (either a CUA or device number)
08	Packet finite state machine (PFSM)
09	Index to slot that packet arrived on
0A–0B	Packet length
0C–1F	First 20 bytes of the packet.

PKTX Entry for X.25 Packet Sent or Received (Part 2)

Entry:	**PKTX**
VIT Option:	CIO
Event:	X.25 packet sent or received (Part 2)
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	ISTTSC5B
Applies to:	VM VSE

This trace record is a continuation of the PKTI or PKTO trace record.



00–03	04	05–1F
PKTX	ID	ADDITIONAL BYTES OF PACKET

Byte (hex)	Contents
00–03	Record ID: C'PKTX'
04	ID is one of the following: • VM Machine ID and Task ID • VSE Task ID
05–1F	Up to 27 additional bytes of the packet.

POST Entry for Post Waiting Event

Entry:	**POST**
VIT Option:	PSS
Event:	Post waiting event
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCTP
Applies to:	MVS VM VSE

This trace record identifies an RPH that is being posted (via the TPPOST macro) for restart after a TPWAIT macro. Usually the POST entry is followed sometime later by a RESM entry. However, when a TPPOST precedes the TPWAIT, the TPWAIT returns to the caller immediately without waiting. Then POST is not followed by RESM.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
POST				ID		0		PABOF				PST ADDRESS				PAB ADDRESS				RETURN ADDRESS				WORK ELEMENT ADDRESS				DVT ADDRESS				RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'POST'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Zero
06—07	PAB offset
08–0B	PST address
0C–0F	PAB address
10–13	Address of the issuer of the TPPOST macro
14–17	Work element address (from RPHWEA)
18–1B	PAB DVT address
1C–1F	Address of the RPH being posted.

PROA or PROD Entry for Profile Add or Delete

Entry:	**PROA** or **PROD**
VIT Option:	NRM
Event:	Profile Add or Delete
VIT Processing Module:	ISTRACNR
Control is Returned to:	ISTSDCPM
Applies to:	MVS VM VSE

This trace record give information about PROFILE macroinstruction processing. It is written when a VTAM module issues a PROFILE macroinstruction to add or delete an RDTE profile.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
PROA PROD				ID	KEY INDEX or 0			INPUT PROFILE ADDRESS				OUTPUT PROFILE ADDRESS				RETURN ADDRESS				RDTE ADDRESS				RESOURCE NAME							

Byte (hex) Contents

00–03 Record ID:
- C'PROA': Add a profile
- C'PROD': Delete a profile

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05–07 Key index into the RDTE profile table or 0. Note that an index value of 0 is valid and represents the first slot in the profile table. The index might also be 0 for PROA entries created when a profile is first associated with an RDTE during SYSDEF processing.

08–0B Input profile address

0C–0F Output profile address

10–13 Return address of the module calling the ISTSDCRP module or the address of the module issuing PROFILE macro

14–17 RDTE address

18–1F Resource name.

QREQ Entry for Queued REQSTORE

Entry:	**QREQ**
VIT Option:	SMS
Event:	Queued storage request
VIT Processing Module:	ISTRACSM
Control is Returned to:	ISTORFBQ
Applies to:	MVS VM

This trace record identifies a REQSTORE request that was waiting for one or more buffers, and which is now satisfied.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
QREQ				ID	CBID	0		PST ADDRESS				BUFFER ADDRESS				0				NUM BUF REQ		0						RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'QREQ'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM
05	Control block ID index value
06–07	Zero
08–0B	PST address
0C–0F	Address of buffer obtained
10–13	Zero
14–15	Number of buffers requested
16–1B	Zero
1C–1F	Request parameter header (RPH) Address.

QRYL Entry for Query Language

Entry:	**QRYL**
VIT Option:	MSG
Event:	Query language error
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTINCUL
Applies to:	MVS

This trace record is generated when a non-zero return code is received from the QRYLANG macro. When the return code and reason code indicate that the MVS message service is not active, no QRYL trace record is generated.

0 0 0 0 0 1 2 3	0 4	0 0 0 0 5 6 7	0 0 0 0 8 9 A B	0 0 0 0 C D E F	1 1 1 1 0 1 2 3	1 1 1 1 4 5 6 7	1 1 1 1 1 1 1 1 8 9 A B C D E F
QRYL	ID	0	LANG QUERY BLOCK ADDRESS	LANG QUERY BLOCK LENGTH	RETURN CODE	REASON CODE	FIRST 8 BYTES OF LANGUAGE NAME USED IN QUERY

Byte (hex)	Contents
00–03	Record ID: C'QRYL'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05–07	Zero
08–0B	Language query block address
0C–0F	Language query block length (LQBSIZE)
10–13	Return code
14–17	Reason code
18–1F	First 8 bytes of language name used in query (LQBINLNG).

QUE Entry for Work Element Queued to PAB

Entry:	**QUE**
VIT Option:	PSS
Event:	Work element queued to PAB
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCTQ
Applies to:	MVS VM VSE

This trace record shows a work element queued to a PAB to allow another VTAM routine to do further processing with the work element. This entry may be followed shortly by a DSP entry, representing the PSS dispatch of this PAB. However, if the PAB is already running, it might not be dispatched again. In this case, no DSP entry follows the TPQUE.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
QUE			CBID		ID		STAT		FLAGS			PST ADDRESS				PAB ADDRESS				RETURN ADDRESS				WORK ELEMENT ADDRESS				MODULE NAME OR DVT ADDRESS				RPH ADDRESS OR 0			

Byte (hex) Contents

00–02 Record ID: C'QUE'

03 Control block ID of work element (see Appendix H, "Storage and Control Block ID Codes" on page 1137)

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Status

Bit	Meaning
xx..	Type of scheduling request
	Bit 00 — TPQUE none
	Bit 01 — TPQUE normal
	Bit 10 — TPQUE delay
..x.	Zero
...1	Queue is in last-in-first-out (LIFO) order.
...0	Queue is in first-in-first-out (FIFO) order.
.... 1...	Registers are saved in the RPH control block.
.... .x..	Zero
.... ..1.	PAB work element queue gate bit (PABWEQG).
....0	PAB change bit (PABCHNG). PAB **is not** scheduled to run.
....1	PAB **is** scheduled to run.

06 PAB flag field (PABFLAGS)

Bit	Meaning
1...	PAB is unconditionally scheduled.
.1..	PAB closedown is in progress.
..1.	PAB is synchronous.
...1	PAB extension is present.
.... 1...	Do not dequeue work element.
.... .1..	Do not detach the RPH.
.... ..1.	Indicates a very extended PAB.
....1	Indicates a slightly extended PAB.

07 PAB flag field (PABFLGS1)

Bit	Meaning
1...	Switch the PST address of this PAB's major control block to the new PST address contained in DYPNWPST.
.1..	This PAB has a data space extension.
..1.	This PAB's major control block is an FMCB.
...1	PAB can be referenced in PSW disable mode.
.... 1...	PAB is persistent.
.... .1..	APSTERM/APSINIT FMCB during PAB dispatch.
.... ..xx	Reserved.

08–0B PST address

0C–0F PAB address

10–13 Address of the issuer of the TPQUE macro

14–17 Address of work element to be queued

18–1B Module name abbreviation (bytes 4, 5, 7, and 8 of the module name), PAB DVT address (high order bit of X'18' = 0), or one of the following abbreviations, which may represent a function or process. The module name might be unavailable if the PAB being scheduled is associated with an address space different from the current one.

Abbreviation	**Meaning**
CPSS	APPN session services for CP-CP sessions
DSVC	Directory services
DIIO	Disk I/O
LUSS	APPN session services for LU-LU sessions
MST	Management services and session initiation and termination
TRS	Topology and routing services

1C–1F Request parameter header (RPH) Address or zero.

QUEN Entry for Work Element Queued to Any Control Block

Entry: **QUEN**
VIT Option: PSS
Event: Work element queued to any control block
VIT Processing Module: ISTRACPS
Control is Returned to: ISTAPCTQ
Applies to: MVS VM VSE

This trace record shows a work element queued to any control block (not just a PAB) to allow another VTAM routine to further process the work element.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
QUEN				ID	CBID	0		PST ADDRESS OR 0				QUEUE ADDRESS				RETURN ADDRESS				WORK ELEMENT ADDRESS				0				RPH ADDRESS OR 0			

Byte (hex) Contents

00–03 Record ID: C'QUEN'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Control block ID of work element (see Appendix H, "Storage and Control Block ID Codes" on page 1137)

06–07 Zero

08–0B Address of PST or zero

0C–0F Address of the control block field to which the work element is queued

10–13 Address of the issuer of the TPQUE NONE macro

14–17 Address of work element to be queued

18–1B Zero

1C–1F Request parameter header (RPH) Address or zero.

RACR Entry for LU 6.2 Invocation of RACROUTE

Entry: **RACR**
VIT Option: APPC
Event: Resource access control
VIT Processing Module: ISTRACAC
Control is Returned to: ISTNSCSI
Applies to: MVS VM

This trace record shows information about the completion of a RACROUTE macro. This trace record is written during security processing for session establishment.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
RACR				ID	REQ	TYPE	0	ACEE ADDRESS				HALF SESS ID				0				RACR COMPLET CODE				REQUEST RETURN CODE				REQUEST REASON CODE			

Byte (hex) Contents

00–03 Record ID: C'RACR'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM

05 RACROUTE request
X'01' AUDIT
X'02' EXTRACT
X'03' LIST
X'04' VERIFY

06 RACROUTE type
For AUDIT: Zero
For EXTRACT, LIST, or VERIFY:
X'01' CREATE
X'02' DELETE
X'03' ENCRYPT
X'04' EXTRACT

07 Zero

08–0B Address control environment element (ACEE) address

0C–0F Half-session ID

10–13 Zero

14–17 RACROUTE completion code
X'00' Request successfully completed
X'04' Request completed with non-zero return/reason code
X'08' RACROUTE failure

18–1B Request return code from security management product[8]

1C–1F Request reason code from security management product[8].

[8] See the appropriate manual for your security management product for an explanation of the return/reason codes.

RCEx Entry for RCE Macro

Entry:	**RCEA, RCEC, RCED, or RCEF**
VIT Option:	NRM
Event:	RCE macro
VIT Processing Module:	ISTRACNR
Control is Returned to:	ISTRACSC
Applies to:	MVS VM VSE

This trace record contains information about an RCEADD, RCECHG, RCEDEL, or RCEFIND macroinstruction issued by a VTAM module.

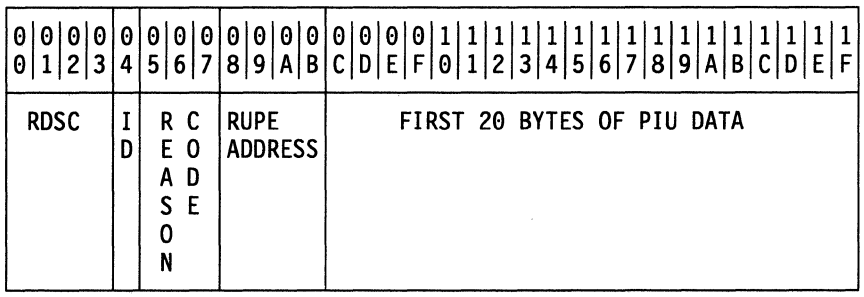
00	01	02	03	04	05	06	07	08 09 0A 0B 0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
RCEA RCEC RCED RCEF				ID	RTNCD	TYPE	MODE	KEY VALUE	RETURN ADDRESS	OUT1 OR NEW TABLE NAME	OUT2 OR NEW TABLE NAME	DATA ADDRESS

Byte (hex)	Contents
00–03	Record ID: C'RCEA' (RCEADD), C'RCEC' (RCECHG), C'RCED' (RCEDEL), C'RCEF' (RCEFIND)
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Return code
06	Type of RCE entry
07	Mode of RCE entry (in hex) • 01=Base entity mode • 02=Reference entity mode
08–0F	Key value (base entity name, base index, or reference index, depending on mode)
10–13	Address of the issuer of the macro
14–17	For RCEADD, RCEDEL, and RCEFIND: Value in RCEPOUT1 For RCECHG: First half of new table name
18–1B	For RCEADD, RCEDEL, and RCEFIND: Value in RCEPOUT2 For RCECHG: Second half of new table name
1C–1F	Data address.

RDSC Entry for RUPE Discard (Part 1)

Entry:	**RDSC**
VIT Option:	PIU
Event:	Discarding a RUPE
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTDLCDI
Applies to:	MVS VM VSE

This trace record is written when VTAM's dependent LU server receives an erroneous signal either from the VTAM configuration services component or from the dependent LU requester node across the CPSVRMGR session.



Byte (hex) Contents

00–03 Record ID: C'RDSC'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05–07 Reason code (decimal)
- 1 The dependent LU server abend recovery routine, DLCRR, performs clean-up.
- 2 CV X'60' is not found.
- 3 Dependent LU requester element is not found.
- 4 PU element is not found.
- 5 LU element is not found.
- 6 Storage shortage has occurred.
- 7 Dependent LU requester finite state machine is not valid.
- 8 Dependent LU requester finite state machine is pending inactive.
- 9 CPSVRMGR session pipe activation failure has occurred.
- 10 Unrecognized request unit processing element (RUPE) detected.
- 11 Start transaction program (STARTTP) failure has occurred.
- 12 CPSVRMGR session pipe has already been deactivated.
- 13 PU finite state machine is not valid.
- 14 SEND_ENCAP_TP process anchor block (PNB) does not exist.

08–0B Address of the RUPE being discarded

0C–1F First 20 bytes of PIU data (inbound) or RUPE's RU data (outbound) in the RUPE being discarded.

RDS2 Entry for RUPE Discard (Part 2)

Entry: **RDS2**
VIT Option: PIU
Event: Discarding a RUPE
VIT Processing Module: ISTRACOT
Applies to: MVS VM VSE

This trace record is a continuation of the RDSC entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
RDS2				NEXT 20 BYTES OF PIU DATA																				RETURN		ADDRESS		MODULE			NAME

Byte (hex)	Contents
00–03	Record ID: C'RDS2'
04–17	Bytes 21 through 40 of the PIU data (inbound) or RUPE's RU data (outbound) in the RUPE being discarded
18–1B	Return address
1C–1F	The module name that initiates the RUPE discard.

RE Entry for RPL Exit

Entry: **RE**
VIT Option: API
Event: RPL exit
VIT Processing Module: ISTRACAP
Control is Returned to: ISTAPCSX, ISTAPCUE, or ISTAICPT
Applies to: MVS VM VSE

This trace record contains RPL information passed from VTAM to the application program. If the return code or feedback is non-zero, this entry will be generated whether the API option is in effect or not. It is treated as an exception condition and is always traced, regardless of the VIT options specified, if the VIT is active.

See "RPL-Based Macroinstruction Errors and Special Conditions" in *VTAM Programming* for a description of RPL return codes.

00	01	02	03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
RE	REQTY	EXTDS	ID	RTNCD	FDB2	FDB3		RPL ADDRESS	EXIT ADDRESS	RPL AREA	RPL RLEN	CID OR 0	RPL FDBK2

Byte (hex) Contents

00–02 Record ID: C'RE'

03 RPL request type (RPLREQ)
- X'15' SETLOGON
- X'16' SIMLOGON
- X'17' OPNDST
- X'19' CHANGE
- X'1A' INQUIRE
- X'1B' INTRPRET
- X'1F' CLSDST
- X'22' SEND
- X'23' RECEIVE
- X'24' RESETSR
- X'25' SESSIONC
- X'27' SENDCMD
- X'28' RCVCMD
- X'29' REQSESS
- X'2A' OPNSEC
- X'2C' TERMSESS

03 Exit definition (RPLEXTDS)

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Return code (RPLRTNCD) (See "RPL-Based Macroinstruction Errors and Special Conditions" in *VTAM Programming*)

06	Feedback code (RPLFDB2) (See "RPL-Based Macroinstruction Errors and Special Conditions" in *VTAM Programming*)
07	Feedback data flag (RPLFDB3) (See "RPL-Based Macroinstruction Errors and Special Conditions" in *VTAM Programming*, under INQUIRE macro with OPTCD=APPSTAT)
08–0B	RPL address
0C–0F	Exit address
10–13	RPL data area pointer (RPLAREA)
14–17	Record length (RPLRLEN)
18–1B	CID from NIB or from RPL (RPLARG) or zero
1C–1F	Second RPL feedback area - Sense code (RPLFDBK2).

RELS Entry for Release Storage

Entry:	**RELS**
VIT Option:	SMS
Event:	Release storage
VIT Processing Module:	ISTRACSM
Control is Returned to:	MVS ISTORMBD VM ISTORFBD VSE ISTORCRT
Applies to:	MVS VM VSE

This trace record provides the status of each RELSTORE request issued by VTAM components to release fixed-length buffers to one of the predefined buffer pools.

RELSTORE is the complement of REQSTORE. Storage obtained by REQSTORE must be released by RELSTORE. However, because more than one buffer can be obtained and released at a time, there may not be a RELS for every REQS, and there may not be a REQS for each RELS. Use the "next buffer address" in RELS and the "number of buffers" in REQS to help determine the correlation between buffers requested and buffers released.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
RELS				ID	CBID	0	RTNCD	PST ADDRESS				BUFFER ADDRESS				RETURN ADDRESS				NEXT BUFFER ADDRESS OR 0				REG 1				CALLER OF UTILITY OR 0			

Byte (hex) Contents

00–03 Record ID: C'RELS'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 MVS VM Control block ID index value or zero. A zero is returned if the buffer is not allocated for a specific control block.

For the index values and the control block IDs they represent, see *VTAM Data Areas for MVS/ESA* or *VTAM Data Areas for VM/ESA*.
VSE First character of buffer pool ID

06 MVS VM Zero
VSE Last character of buffer pool ID. See "SMS Trace Record Output (VSE)" on page 379 for a listing of the 2-character values.

07 Return code

08–0B PST address

0C–0F Address of buffer to be released

10–13 Address of the issuer of the RELSTORE macro

14–17 Pointer to the next buffer to be released if more than one buffer to be released; otherwise, zero

18–1B Register 1 (normally RPH address)

1C–1F Caller of utility routine, or zero

MVS Address of the utility's caller if the RELSTORE macro was issued from a utility routine; otherwise, zero if the macro was issued directly by the caller without a utility routine (See bytes 10–13 for the address of the issuer of the RELSTORE macro).

VM VSE Zero.

REQS Entry for Request Storage (Part 1)

Entry: **REQS**
VIT Option: SMS
Event: Request storage (Part 1)
VIT Processing Module: ISTRACSM
Control is Returned to: MVS ISTORMBA
VM ISTORFBA
VSE ISTORCRQ
Applies to: MVS VM VSE

This trace record provides the status of each REQSTORE request issued by VTAM components to obtain fixed-length buffers from one of the predefined buffer pools. The storage obtained by REQSTORE is released by RELSTORE.

If this entry is associated with an event failure (that is, the return code is non-zero), this entry will be generated whether the SMS option is in effect or not. It is treated as an exception condition and is always traced, regardless of the VIT options specified, if the VIT is active.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
REQS				ID	CBID	0		PST ADDRESS OR 0				BUFFER ADDRESS OR 0				RETURN ADDRESS				NUM BUF REQ		0		REG 1				RETURN CODE			

Byte (hex) Contents

00–03 Record ID: C'REQS'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 MVS VM Control block ID index value or zero. A zero is returned if the buffer is not allocated for a specific control block or if the buffer allocation failed.

For the index values and the control block IDs they represent, see *VTAM Data Areas for MVS/ESA* or *VTAM Data Areas for VM/ESA.*
VSE First character of buffer pool ID

06–07 MVS VM Zero
VSE Last character of buffer pool ID See "SMS Trace Record Output (VSE)" on page 379 for a listing of the 2-character values.

08–0B Pointer to the PST, or zero

0C–0F Address of buffer obtained, or zero if buffer not obtained

10–13 Address of the issuer of the REQSTORE macro

14–15 Number of buffers requested

16–17 Zero

18–1B Register 1 (normally RPH address)

1C–1F Return code
- 00 Successful.
- 04 Storage obtained, but slowdown threshold exceeded.

06 Storage request queued, but buffer pool has to be expanded before buffers can be allocated.

08 The number of buffers requested could not be allocated immediately. Depending on the conditions specified on the REQSTORE request and the state of the buffer pool that is requested, the request may or may not be queued for future processing.

The request is queued for future processing if one of the following sets of conditions is met:

- The REQSTORE request specifies that it can be queued for future processing regardless of the state of the buffer pool from which buffers are requested.
- The REQSTORE request specifies that it can be queued for future processing if specific conditions are met. These conditions are as follows:
  - Number of buffers already allocated plus the number of buffers for which there are outstanding requests does not exceed the maximum size of the buffer pool.
  - Buffer pool is defined to be expandable.
  - Buffer pool expansion has never failed for the buffer pool or the buffer pool has contracted since it failed to expand.

The request is not queued for future processing if any of the following conditions apply:

- The REQSTORE request specifies that it is not to be queued under any circumstances.
- The REQSTORE request specifies that it is to be queued according to one or more of the conditions described previously and the specified conditions are not met.

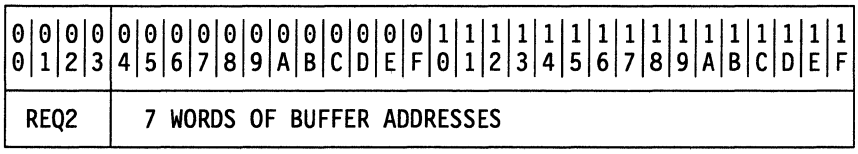
16 This return code applies only to non-expandable buffer pools. The number of buffers requested exceeds the maximum number which can be allocated in one request. This return code does not indicate that the number of buffers requested is not available; it indicates that the number of buffers exceeds the maximum allowed in one request.

REQ2 Entry for Request Storage (Part 2)

Entry: **REQ2**
VIT Option: SMS
Event: Request storage (Part 2)
VIT Processing Module: ISTRACSM
Applies to: MVS VM VSE

This trace record is a continuation of the REQS entry. It is generated when REQSTORE macro was issued for more than one buffer. The number of records generated depends on the number of buffers obtained by the REQSTORE request.

VTAM generates up to 31 REQ2 trace records. Each REQ2 entry contains addresses for up to 7 buffers. If more than 218 buffers are requested, the last four bytes, 1C–1F, in the last REQ2 entry is set to X'FFFF' indicating that not all of the requested buffers are traced.



Byte (hex)	Contents
00–03	Record ID: C'REQ2'
04–1F	Up to 7 more words of buffer addresses (padded on the right with zeros).

RESM Entry for Resume Event Processing

Entry:	**RESM**
VIT Option:	PSS
Event:	Resume event processing
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCPD
Applies to:	MVS VM VSE

This trace record gives information about which VTAM process is about to be redispatched after a TPWAIT. A RESM entry follows a POST, although not every POST has a RESM. Usually the POST entry is followed sometime later by a RESM entry. However, when a TPPOST precedes the TPWAIT, the TPWAIT returns to the caller immediately without waiting. Then POST is not followed by RESM. Use the RPH address to find the original DSP entry, which may have occurred quite some time earlier.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
RESM				ID	CBID	FLAGS		PST ADDRESS				PAB ADDRESS				WORK EL ADDR or NEXT DSP Q LVL				WORK ELEMENT QUEUE				PAB DVT ADDRESS or NAME				RPH ADDRESS			

Byte (hex) Contents

00–03 Record ID: C'RESM'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Control block ID of work element (see Appendix H, "Storage and Control Block ID Codes" on page 1137)

06 PAB flag field (PABFLAGS)

Bit	Meaning
1...	PAB is unconditionally scheduled.
.1..	PAB closedown is in progress.
..1.	PAB is synchronous.
...1	PAB extension is present.
.... 1...	Do not dequeue work element.
.... .1..	Do not detach the RPH.
.... ..1.	Indicates a very extended PAB.
....1	Indicates a slightly extended PAB.

07 PAB flag field (PABFLGS1)

Bit	Meaning
1...	Switch the PST address of this PAB's major control block to the new PST address contained in DYPNWPST.
.1..	This PAB has a data space extension.
..1.	This PAB's major control block is an FMCB.
...1	PAB can be referenced in PSW disable mode.
.... 1...	PAB is persistent.

.... .1..	APSTERM/APSINIT FMCB during PAB dispatch.
.... ..xx	Zero.
08–0B	PST address
0C–0F	PAB address
10–13	Work element address or next dispatchable queue level for a very extended PAB
14–17	PAB work element queue
18–1B	PAB DVT address or name
1C–1F	Request parameter header (RPH) Address.

SCHD Entry for Schedule of PAB Dispatch

Entry: **SCHD**
VIT Option: PSS
Event: Schedule of PAB dispatch
VIT Processing Module: ISTRACPS
Control is Returned to: ISTAPCTS
Applies to: MVS VM VSE

This trace record shows the schedule of a PAB to be dispatched and the last work element, if any, that was queued to the PAB. This entry should be followed shortly by a DSP entry, representing the PSS dispatch of this PAB.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SCHD				ID	STAT	FLAGS		PST ADDRESS				PAB ADDRESS				RETURN ADDRESS				WORK EL Q OR NEXT DSP Q LVL				MODULE NAME OR DVT ADDRESS				RPH ADDRESS			

Byte (hex) Contents

00–03 Record ID: C'SCHD'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Status

Bit	Meaning
xx..	Zero
..1.	TPSCHED unconditional
...x	Zero
.... 1...	Registers are saved in the RPH control block.
.... .x..	Zero
.... ..1.	PAB work element queue gate bit (PABWEQG).
....0	PAB change bit (PABCHNG) = 0 if the PAB **is not** scheduled to run.
....1	PABCHNG=1 if the PAB **is** scheduled to run.

06 PAB flag field (PABFLAGS)

Bit	Meaning
1...	PAB is unconditionally scheduled.
.1..	PAB closedown is in progress.
..1.	PAB is synchronous.
...1	PAB extension is present.
.... 1...	Do not dequeue work element.
.... .1..	Do not detach the RPH.
.... ..1.	Indicates a very extended PAB.
....1	Indicates a slightly extended PAB.

07	PAB flag field (PABFLGS1)	
	Bit	Meaning
	1...	Switch the PST address of this PAB's major control block to the new PST address contained in DYPNWPST.
	.1..	This PAB has a data space extension.
	..1.	This PAB's major control block is an FMCB.
	...1	PAB can be referenced in PSW disable mode.
	.... 1...	PAB is persistent.
	.... .1..	APSTERM/APSINIT FMCB during PAB dispatch.
	xx	Zero.
08–0B	PST address associated with PAB	
0C–0F	PAB address	
10–13	Address of the issuer of the TPSCHED macro	
14–17	PAB work element queue or next dispatchable queue level for a very extended PAB	
18–1B	Module name abbreviation (bytes 4, 5, 7, and 8 of the module name), or PAB DVT address (high order bit of X'18' = 0). The module name might be unavailable if the PAB being scheduled is associated with an address space different from the current one	
1C–1F	Request parameter header (RPH) Address.	

SIO Entry for Start I/O for X.25 Port

Entry:	**SIO**
VIT Option:	CIO
Event:	Start I/O
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	ISTAPCES
Applies to:	VM VSE

This trace record provides information about Start I/O at the channel interface for an X.25 port.

The telecommunications subsystem or communication adapter uses two channel programs: a control-link channel program and a buffer-pool channel program. An SIO entry is built whenever one of these channel programs is started. These channel programs run continuously and stop only in case of error or when the adapter stops.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SIO			TYPE	ID	CUA			PFSM	0	DUMP OF WRITE CONTROL AND READ CONTROL DATA AREAS																					

Byte (hex)	Contents
00–02	Record ID: C'SIO'
03	Type: 0C = Control link channel program 0D = Buffer pool channel program
04	ID is one of the following: • VM Machine ID and Task ID • VSE Task ID Channel device name in EBCDIC (either a CUA or device number)
08	Packet finite state machine (PFSM)
09	Zero
0A–1F	Dump of Write Control and Read Control data areas.

SIOx Entry for Start I/O (Part 1)

Entry:	**SIOB, SIOH, SIOI, SIOL, or SIOX**
VIT Option:	CIO
Event:	Start I/O (Part 1)
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCIO or ISTTSCLC
Applies to:	MVS VM VSE

This trace record is written when a Start I/O SVC is issued for a channel-attached communication controller or SNA cluster controller.

- SIOI is generated for communication controllers and local SNA cluster controllers.
- SIOL is generated for local non-SNA cluster controllers.
- SIOX is generated for channel-to-channel-attached hosts and channel-attached IBM 3172 Interconnect Controllers.

Eventually, an INT entry follows the SIO trace record. Use the CUA field to correlate the entries. See *VTAM Data Areas for MVS/ESA*, *VTAM Data Areas for VM/ESA*, or *VTAM Data Areas for VSE/ESA* for a description of the NCB fields.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SIOB or SIOH				ID	LFSM	MODID		CUA DEVICE				NCB ADDRESS				FLAG BYTES				CAW				0							

SIOI or SIOL				ID	LNKST	MODID		CUA DEVICE				NCB ADDRESS				FLAG BYTES				CAW				0							
SIOX				ID	STATE	MODID		CUA DEVICE				NCB ADDRESS				FLAG BYTES				CAW				CCW or OUTPUT CONTROL AREA (STATUS INFORMATION)							

Byte (hex) Contents

00–03 Record ID:
- C'SIOB' for BSCLB VM VSE
- C'SIOH' for HALCB VM VSE
- C'SIOI' for ICNCB
- C'SIOL' for LDNCB
- C'SIOX' for XCNCB

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM

	• Task ID VSE
05	For SIOB or SIOH: link finite state machine (LFSM) VM VSE For SIOI or SIOL: link station state (NCBLNKST) For SIOX: station state (XCNSSFSM)
06–07	Module ID (last two characters of TSC module name)
08–0B	Channel device name in EBCDIC (either a CUA or device number)
0C–0F	NCB address
10–13	Flag bytes (NCBFLAGS)
14–17	Virtual channel address word (NCBCAW)
18–1F	For SIOB and SIOH: Zero VM VSE For SIOI and SIOL: Zero For SIOX: • If link is active, output control area (status information) • If link is not active, first channel command word (CCW).

SIO2 Entry for Start I/O (Part 2)

Entry: **SIO2**
VIT Option: CIO
Event: Start I/O (Part 2)
VIT Processing Module: ISTRACCI
Applies to: MVS VM VSE

This trace record is a continuation of the SIO entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SIO2				VIRTUAL BUFFER ADDRESS				MPC TRANSMIT BLOCK HEADER								FIRST MPC PDU HEADER										0				ERACT	

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SIO2				ERACT	TECT	ERCOD	RCODE	MODID		DEVICE-DEPENDENT INFORMATION or 0																0				ERACT	

Byte (hex) Contents

00–03 Record ID: C'SIO2'

04–07

- When multipath channel (MPC),
  - 04–07 Virtual buffer address of the first READ CCW, or zero
- When device other than MPC,
  - 04 ERACT or zero
  - 05 Temporary error counter (TECT)
  - 06 Error code (NCBERCOD)
  - 07 Return code or zero

08–1F

- When multipath channel (MPC),
  - 08–0F MPC transmit block header
  - 10–19 First MPC PDU header
  - 1A–1D Zero
  - 1E VM VSE Error action (ERACT) or zero
  - 1F Zero
- When device other than MPC,
  - 08–09 Module ID
  - 0A–19 Device-dependent information shown below, or zero
  - 1A–1D Zero
  - 1E VM VSE Error action (ERACT) or zero
  - 1F Zero

For SDLC links (NCBTYPE=09):

0A Index to current PUT
0B Station flags

0C	Station finite state machine
0D	Connection finite state machine
0E–0F	Connection manager flags
10–19	First 10 bytes of CCW command code

For BSC links (NCBTYPE=0A):

0A	Current channel program finite state machine (BSCCCFSM)
0B	Original channel program finite state machine (BSCOCFSM)
0C	Previous channel program finite state machine (BSCPCFSM)
0D	Original failing channel program finite state machine (BSCFCFSM)
0E	First selection character (BSCSEL1)
0F	First device address (BSCDEV1)
10	Index of BPL entry for which input data was received (BSCPESV)
11	Device address from which input data was received (BSCDEVSV)
12	Index for last station for which there was output data (BSCISOD)
13	First response byte (BSCRSPA1)
14	Second response byte (BSCRSPA2)
15	Last block type received (BSCLBR)
16	Current acknowledgment (BSCALTAK)
17	Error retry count (BSCRCNT)
18	Connection manager flags (BSCCMIO)
19	BSC flags (BSCFLAGS).

SIOP Entry for Start I/O

Entry: **SIOP**
VIT Option: CIO
Event: Start I/O for token-ring subsystem (Part 1)
VIT Processing Module: ISTRAC6T
Control is Returned to: ISTTSC6F
Applies to: VM VSE

This trace record is created when a Start I/O SVC is issued for the Token-Ring Subsystem. Channel programs started for the subsystem do not always complete. If the channel program completes, this entry is eventually followed by an INT entry. Use the CUA field to correlate the entries.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SIOP				ID	CPFSM	MODID		0	CUA			NCB ADDRESS				PDATA		0		CAW				CODE	0						

Byte (hex)	Contents
00–03	Record ID: C'SIOP'
04	ID is one of the following: • VM Machine ID and Task ID • VSE Task ID
05	Channel program FSM 00 Reset 01 Sense ID 02 Interrupt port 03 Control port 04 Running 05 Sense 06 Orderly halt (HLTC) 07 Termination 08 Shutdown
06–07	Module ID (last 2 characters of TSC module name)
08	Zero
09–0B	Channel device name in EBCDIC (either a CUA or device number)
0C–0F	NCB address
10	Port state (PDATA) 00 Reset 01 Halt 02 Halt complete 03 Sense ID 04 Sense ID complete 05 Interrupt pending 06 Active 07 Sense
11	Port type (PDATA) 00 Interrupt

	01 Control
	02 Inbound
	03 Outbound
12–13	Zero
14–17	Channel address word
18	Channel-command code for the first CCW
	C4 Sense control unit state
	CB Set CETI mode on
	E4 Sense ID
19–1F	Zero.

SOCx Entry for SOCTREE ADD, DELETE, FIND, and RELEASE Macros

Entry: **SOCA, SOCD, SOCF, or SOCR**
VIT Option: TCP
Event: SOCTREE ADD, DELETE, FIND, and RELEASE macros
VIT Processing Module: ISTRACIP
Control is Returned to: ISTIPCMT
Applies to: MVS

This trace record is written when a VTAM module issues a SOCTREE ADD, DELETE, FIND, or RELEASE.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SOCA SOCD SOCF SOCR				ID	LOCK	RC	0	SOCKET CONTROL BLOCK ADDRESS				0				RETURN ADDRESS				CORRELATOR FROM THE MPTN KEY								RPH ADDRESS			

Byte (hex) Contents

00–03 Record ID:
- C'SOCA' for SOCTREE ADD
- C'SOCD' for SOCTREE DELETE
- C'SOCF' for SOCTREE FIND
- C'SOCR' for SOCTREE RELEASE

04 ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.

05 Locking level on TREELOCK

Code	Meaning
X'01'	Exclusive
X'02'	Share

06 SOCTREE return code

Code	Meaning
X'00'	Successful invocation
X'01'	Socket already exists in the tree
X'02'	Socket was not found in the tree

07 Zero

08-0B Address of the socket control block during an add, a successful find, or a delete. Otherwise, this field is zero.

0C-0F Zero

10-13 Address to which the macro returns control

14-1B Procedure correlation identifier (PCID)

1C-1F Request parameter header (RPH) address.

SPT Entry for SPT Macro Invoked (Part 1)

Entry: **SPT**
VIT Option: SSCP
Event: SPT macro invocation
VIT Processing Module: ISTRACSC
Control is Returned to: ISTCPMSP
Applies to: MVS

This trace record is written when partner LU mapping information in the coupling facility structure ISTGENERIC is created, deleted, or changed using the SPT macroinstruction.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SPT				ID	RTNCD	FUNC	FLAGS	GENERIC NAME								RETURN ADDRESS				REAL RESOURCE NAME								RPH ADDRESS			

Byte (hex) Contents

00–03 Record ID: C'SPT'

04 ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.

05 Return code from the SPT macro

06 Reason macro invoked:

- X'01' Find an SPT entry
- X'02' Add a session pair to the SPT entry
- X'03' Decrement the session count for a session pair
- X'04' Increment the session count for a session pair
- X'05' End the affinity between the generic resource and its partner LU
- X'06' Start an affinity between the generic resource and its partner LU
- X'07' Delete a session pair in a SPT entry
- X'08' Free local SPT entry that was read from the coupling facility structure

07 Flags:

- x... .... Name type
  - B'0' Real names only
  - B'1' Generic names allowed
- .1.. .... This SPT will persist until the application issues the CHANGE OPTCD=ENDAFFIN macroinstruction. See Chapter 13, "Conventions and Descriptions of VTAM Macroinstructions" in *VTAM Programming* for additional information.
- ..x. .... Ownership indicator
  - B'0' VTAM-owned
  - B'1' Application-owned
- ...x xxxx Zero

08–0F	Generic resource name
10–13	Address of the issuer of the SPT macro
14–1B	Application program network name
1C–1F	Request parameter header (RPH) Address.

SPT2 Entry for SPT Macro Invoked (Part 2)

Entry: **SPT2**
VIT Option: SSCP
Event: SPT macro invocation
VIT Processing Module: ISTRACSC
Applies to: MVS

This trace record is a continuation of the SPT entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SPT2				SIB COUNT				PCID								NETID OF PARTNER LU								PARTNER LU NAME							

Byte (hex) Contents

00–03 Record ID: C'SPT2'
04–07 Count of number of session information blocks (SIB) that have been created for this SPT entry
08–0F PCID if associated with a session, otherwise zero
10–17 Partner NETID
18–1F Partner resource name.

SRBD Entry for SRB Dispatch

Entry: **SRBD**
VIT Option: PSS
Event: SRB dispatch
VIT Processing Module: ISTRACPS
Control is Returned to: ISTAPCAD
Applies to: MVS

This trace record provides information when PSS is dispatched under an SRB.

Note: For this trace record to be created, in addition to specifying the PSS option, you must specify SRB or BOTH on the PSS TRACE start option.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SRBD				ID	0			PST ADDRESS				PST LIFO ASYNC DSP QUEUE				PST FIFO ASYNC DSP QUEUE				PST ASYNC N-DSP QUEUE (ANDSP)				TCB ADDRESS				F L G 1		0	

Byte (hex)	Contents
00–03	Record ID: C'SRBD'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05–07	Zero
08–0B	PST address
0C–0F	PST LIFO asynchronous dispatchable queue (PSTADSP)
10–13	PST FIFO asynchronous dispatchable queue (PSTALIST)
14–17	PST asynchronous nondispatchable queue (PSTANDSP)
18–1B	TCB address (PSTTCBA)
1C–1D	PST flags (PSTFLG1)
1E–1F	Zero.

SRBX Entry for SRB Exit

Entry:	**SRBX**
VIT Option:	PSS
Event:	SRB exit
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCSX, ISTAPCTX, or ISTAPCAD
Applies to:	MVS

This trace record provides information when PSS exits an SRB. For this trace record to be created, you must specify the PSS VIT option, and you must specify SRB or BOTH on the PSSTRACE start option.

00 01 02 03	04	05	06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
SRBX	ID	CODE	0	PST ADDRESS	PST LIFO ASYNC DSP QUEUE	PST FIFO ASYNC DSP QUEUE	NEW ASCB ADDRESS	OLD ASCB ADDRESS	RETURN ADDRESS

Byte (hex)	Contents
00–03	Record ID: C'SRBX'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05	Reason code for SRB exit
06–07	Zero
08–0B	PST address
0C–0F	PST LIFO asynchronous dispatchable queue (PSTADSP)
10–13	PST FIFO asynchronous dispatchable queue (PSTALIST)
14–17	New ASCB address
18–1B	Old ASCB address
1C–1F	Address of the module issuing the SRBX trace event.

SRTx Entry for SRTADD, SRTCHG, SRTDEL, SRTFIND Macros

Entry:	**SRTA, SRTC, SRTD, or SRTF**
VIT Option:	NRM
Event:	SRTADD, SRTCHG, SRTDEL, SRTFIND macros
VIT Processing Module:	ISTRACNR
Control is Returned to:	ISTNRCSA for SRTADD ISTNRCSD for SRTCHG, SRTDEL, and SRTFIND
Applies to:	MVS VM VSE

This trace record contains information about a SRTADD, SRTCHG, SRTDEL, or SRTFIND macroinstruction issued by a VTAM module.

00	01	02	03	04	05	06	07	08 09 0A 0B 0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B 1C 1D 1E 1F
SRTA SRTC SRTD SRTF				ID	RTNCD	0	TYPE	HASH NAME (NAME OR NETWORK ADDRESS)	RETURN ADDRESS	SRT ENTRY ADDRESS	NETWORK ID

Byte (hex) Contents

00–03 Record ID: C'SRTA' (SRTADD), C'SRTC'(SRTCHG), C'SRTD' (SRTDEL), or C'SRTF' (SRTFIND)

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Return code. See the SRT control block in *VTAM Data Areas for MVS/ESA*, *VTAM Data Areas for VM/ESA*, or *VTAM Data Areas for VSE/ESA*.

06 Zero

07 Type of SRT entry. For the types, see the SRT control block in *VTAM Data Areas for MVS/ESA*, *VTAM Data Areas for VM/ESA*, or *VTAM Data Areas for VSE/ESA*.

08–0F Network name or network address (padded on left with zeros) passed to the hashing algorithm

10–13 Address of the issuer of the SRTADD, SRTCHG, SRTDEL, or SRTFIND macroinstruction

14–17 Address of SRT entry for the resource

18–1F Network ID of the resource.

SRT2 Entry for the SRTADD Macro

Entry: **SRT2**
VIT Option: NRM
Event: SRTADD
VIT Processing Module: ISTRACNR
Applies to: MVS VM VSE

This trace record is a continuation of the SRTADD entry. It is written after an SRTADD failure if the duplicate SRTE is contained in or points to an RDTE.

00	01	02	03	04	05	06	07	08 09 0A 0B 0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B 1C 1D 1E 1F
SRT2				SRTUSELN	FLAGS	RPRENTRY	TYPE	RPRNAME	SRTDATA OR 0		NETWORK ID

Byte (hex) Contents

00–03 Record ID: C'SRT2'

04 SRTUSELN from the duplicate SRTE; this is a 4-byte field in the SRTE and will be set to X'FF' in the SRT2 if SRTUSELN is greater than X'FF'.

05 The first bit is SRTSPECE; the other 7 are not used.

06 RPRENTRY of the duplicate RDTE

07 SRTTYP from the duplicate SRTE
- X'00' RDTE
- X'03' Shadow
- X'09' Alias name
- X'0A' Network address
- X'FF' SRTE

08–0F RPRNAME of the duplicate RDTE

10–13 SRTDATA from the duplicate SRTE or 0.

14–17 Not used

18–1F Network ID of the duplicate RDTE. If RPRENTRY is RPRENTRC, it is RCDNETID; otherwise, it is ATCNETID.

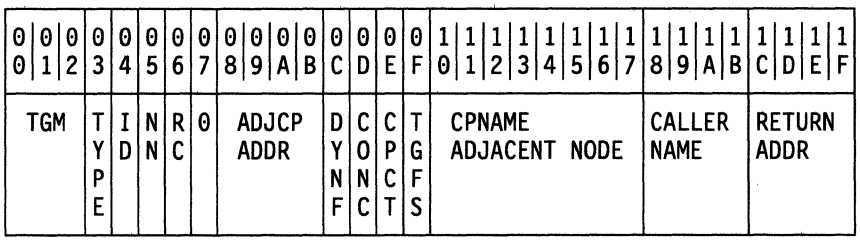
TGM Entry for APPN TG Management (Part 1)

Entry:	**TGM**
VIT Option:	SSCP
Event:	APPN TG management event
VIT Processing Module:	ISTRACSC
Control is Returned to:	ISTATCTR
Applies to:	MVS VM VSE

This record is generated when one of the following APPN TG management events has occurred:

- Add an ATGB to an ATGB chain
- Remove an ATGB from an ATGB chain
- Find an ATGB
- Activate a TG number
- Deactivate a TG number
- Negotiate a TG number.

If TGFIND fails to locate an ATGB, a trace entry will still be recorded.



00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
TGM			TYPE	ID	NNC	RC	0	ADJCP ADDR				DYNF	CONC	CPCT	TGFS	CPNAME ADJACENT NODE								CALLER NAME				RETURN ADDR			

Byte (hex) Contents

00–02 Record ID: C'TGM'

03 Type field

Code	Meaning
C	Activate transmission group number (TGN) or APPN transmission group block (ATGB)
D	Deactivate TGN or ATGB
A	Add an ATGB
R	Remove an ATGB
F	Find an ATGB
N	Negotiate a TGN

04 ID is one of the following:

- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 APPN TG Management function values

- Network node specification
  - 0: Not a network node
  - 1: Network node
- RU being processed
  - 0: Request Contact
  - 1: Contacted
- TG number coded on macroinstruction
  - 0: Not specified

1: Specified
- Display CV47
- APPN connection pending, not reported to topology (LIMBO) specification
  0: No
  1: Yes
- Vector format
  0: Key-length format
  1: Length-type format
- Assigned chain specification
  0: Assigned not specified
  1: Assigned specified
- Unassigned chain specification
  0: Unassigned not specified
  1: Unassigned specified

06	The calling module's return code. Zero if no return address is provided.
07	Unused available
08–0B	Adjacent control point (ADJCP) address
0C	Dynamic indicators received in XID3 exchange (ADJDYNFL)
0D	Number of active connections controlled (ADJCONCT), last digit
0E	Connections controlled by the ADJCP (ADJCPCT), last digit
0F	Transmission group number state map (ADJSTMP), indexed by TGN
10–17	CPNAME of the adjacent node
18–1B	Calling module name
1C–1F	Return address.

TGM2 Entry for APPN TG Management (Part 2)

Entry: **TGM2**
VIT Option: SSCP
Event: APPN TG management event (Part 2)
VIT Processing Module: ISTRACSC
Applies to: MVS VM VSE

This trace record is a continuation of the TGM trace record.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
TGM2				PU of ATGB OR RCV TGN				ATGB ADDRESS				TG NU	TO PR	DY NF	PO S	PUPTR ATGB TOP				NXTPT ATGB TOP				PUPTR ATGB NEXT				NXTPT ATGB NEXT			

Byte (hex) Contents

00–07 Overlay 1
- 00–03 Record ID: C'TGM2'
- 04–07 PU of current ATGB

Overlay 2
- 00–03 Record ID: C'TGM2'
- 04–06: Unused, available
- 07: Received TGN

08–0B ATGB address
0C ATGTGNUM
0D ATGTOPR
0E ATGDYNFL
0F ATGPOS
10–13 ATGPUPTR: Top ATGB in the chain
14–17 ATGNXTPT: Top ATGB in the chain
18–1B ATGPUPTR: Next ATGB in the chain
1C–1F ATGNXTPT: Next ATGB in the chain.

TREx Entry for Routing Tree Build or Update

Entry:	**TREM, TRED, TREI, or TREB**
VIT Option:	SSCP
Event:	Routing tree Build or Update
VIT Processing Module:	ISTRACSC
Control is Returned to:	Module invoking the INTRACE macro that caused the record to be produced.
Applies to:	MVS VM VSE

A routing tree operations (TRE) VIT entry is created by topology and routing services (TRS) whenever one of the following events is encountered:

- TRS accesses or builds a routing tree to satisfy a request for an APPN route.
- An existing routing tree is marked for modification due to a topological change in the APPN network.
- TRS learns of the availability or loss of a central directory server or interchange node.

00–03	04	05	06	07	08–0B	0C–0F	10–13	14–17	18–1B	1C–1F
TREM TRED TREI TREB	ID	MARK FLAGS	ADDR or DEL	TREE FLAGS	PATH WEIGHT OR RES SEQ NUMBER	TREE HEADER POINTER OR TOPO DB RES POINTER	RETURN ADDRESS	ORIGIN TREE RECORD POINTER OR COS POINTER	DEST TREE RECORD POINTER OR CALLING MODULE NAME	TREE BUILD EXEC TIME

Byte (hex)

00–03 Record ID:
- C'TREM' Mark tree
- C'TRED' Add or change directory server
- C'TREI' Add or change interchange node
- C'TREB' Build tree

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Tree marking flags

Bit	Meaning
x...	1=Operable resource state 0=Inoperable resource state
.xxx	Reason 000=Transmission group update 001=Topology database update

010=Garbage collection
011=Overuse
100=Modify topology
110=Uncache tree

.... xxxx Zero.

06 Indicates that a directory server or interchange node is added ('A') or deleted ('D')

07 Tree building flags

Bit	Meaning
x...	Indicates whether existing tree is used 1=Existing tree is used 0=New tree is built; existing tree is not available
.x..	Indicates whether clean path is used 1=Clean path is used 0=Clean path is not available
..x.	Indicates number of destinations 1=Multiple 0=Single

08–0B Indicates the entry type.

Entry Type	Meaning
B	Path weight
M	Resource sequence number

Note: If entry type is TREB and the destination tree record pointer indicates multiple destinations, the path weight field is not valid.

0C–0F

Entry Type	Meaning
B	Tree header pointer
M	Topology database resource pointer

10–13 Return address of caller

14–17 Origin tree record pointer or COS pointer
- Origin tree, if entry type is B
- COS pointer, if tree marking flags are ('110')

18–1B Destination tree record pointer or calling module
- Destination tree record pointer, if entry type is B
- Last 4 characters of calling module name, if tree marking flags are ('110')

1C–1F Tree building execution time (microseconds), if entry type is B.

TRNM Entry for Translate Message (Part 1)

Entry:	**TRNM**
VIT Option:	MSG
Event:	Translate message error (Part 1)
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTUSCMS or IKTMSIFR
Applies to:	MVS

This trace record is generated when a return code greater than the value 4 is received from the TRANMSG macro. When the return code and reason code indicate that the MVS message service is not active, no TRNM trace record is generated. A single message is translated each time the TRANMSG macro is invoked.

00 01 02 03	04	05 06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
TRNM	ID	LANG CODE	MESSAGE I/O BLOCK ADDRESS	MESSAGE I/O BLOCK LENGTH	RETURN CODE	REASON CODE	MPB ADDRESS	MTB ADDRESS

Byte (hex)	Contents
00–03	Record ID: C'TRNM'
04	ID is the primary address space ID (ASID). This field is zero if the ASID is greater than X'FF'.
05–07	Three-character language code or blanks if language is not supported

Hex Value	Language Code	Language Name
X'02'	ARA	Arabic
X'03'	CHT	Traditional Chinese
X'04'	CHS	Simplified Chinese
X'05'	DAN	Danish
X'06'	DEU	German
X'07'	DES	Swiss German
X'08'	ELL	Greek
X'09'	ENG	UK English
X'00'		US English (default)
X'01'	ENU	US English (specified)
X'0A'	ESP	Spanish
X'0B'	FIN	Finnish
X'0C'	FRA	French
X'0D'	FRB	Belgian French
X'0E'	FRC	Canadian French
X'0F'	FRS	Swiss French
X'10'	HEB	Hebrew
X'12'	ISL	Icelandic
X'13'	ITA	Italian
X'14'	ITS	Swiss Italian
X'11'	JPN	Japanese
X'15'	KOR	Korean
X'16'	NLD	Dutch
X'17'	NLB	Belgian Dutch
X'18'	NOR	Norwegian
X'19'	PTG	Portuguese
X'1A'	PTB	Brazil Portuguese

Hex Value	Language Code	Language Name
X'1B'	RMS	Rhaeto-Romanic
X'1C'	RUS	Russian
X'1D'	SVE	Swedish
X'1E'	THA	Thai
X'1F'	TRK	Turkish
X'3F'		Unknown language code

08–0B Message input/output block address

0C–0F Message input/output block length

10–13 Return code

14–17 Reason code

18–1B Input message parameter block address (MIOINPTP). If a message text block or text is supplied instead, 18–1B is the address of the text block or text.

1C–1F Output message text block address (MIOBUFFP).

TRN2 Entry for Translate Message (Part 2)

Entry: **TRN2**
VIT Option: MSG
Event: Translate message error (Part 2)
VIT Processing Module: ISTRACOT
Applies to: MVS

This trace record is a continuation of the TRNM entry. It is generated when a TRNM entry is produced and the input to the TRANMSG macro is the address of a message parameter block.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
TRN2				IST, IKT, or USS MESSAGE NUMBER				FIRST MESSAGE VARIABLE OR 0								SECOND MESSAGE VARIABLE OR 0								THIRD MESSAGE VARIABLE OR 0							

Byte (hex)	Contents
00–03	Record ID: C'TRN2'
04–07	Last four digits of the message ID (for example, 0200 for message IKT0200I, or SG02 for USSMSG02)
08–0F	First 8 bytes of the first message variable, if present; otherwise, zero
10–17	First 8 bytes of the second message variable, if present; otherwise, zero
18–1F	First 8 bytes of the third message variable, if present; otherwise, zero.

UE Entry for USER Exit (DFASY or RESP)

Entry:	**UE**
VIT Option:	API
Event:	User exit (DFASY or RESP)
VIT Processing Module:	ISTRACAP
Control is Returned to:	ISTAPCUE
Applies to:	MVS VM VSE

This trace record identifies the exit dispatched and data passed back to the user application program.

If this entry is associated with an event failure (that is, the return code is non-zero), this entry will be generated whether or not the API option is in effect. It is treated as an exception condition and is always traced, regardless of the VIT options specified, if the VIT is active.

For DFASY and RESP exits (type codes = 07 and 08), the user exit trace record has the following format:

00 01	02	03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
UE	CODE	EXTDS	ID	RTNCD	FDB2	FDB3	RPL ADDRESS	EXIT ADDRESS	RPL AREA	RPL RLEN	CID OR 0	RPL FDBK2

Byte (hex)	Contents
00–01	Record ID: C'UE'
02	Exit type code (hex 7=DFASY, hex 8=RESP)
03	Exit definition (RPLEXTDS)
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Return code (RPLRTNCD) (See "RPL-Based Macroinstruction Errors and Special Conditions" in *VTAM Programming*)
06	Feedback code (RPLFDB2) (See "RPL-Based Macroinstruction Errors and Special Conditions" in *VTAM Programming*)
07	Feedback code (RPLFDB3) (See "RPL-Based Macroinstruction Errors and Special Conditions" in *VTAM Programming*)
08–0B	RPL address
0C–0F	Exit address
10–13	RPL data area pointer (RPLAREA) or zero
14–17	Record length (RPLRLEN)
18–1B	CID from NIB or RPL (RPLARG) or zero
1C–1F	Second RPL feedback area — Sense code (RPLFDBK2).

UE1 Entry for USER Exit (not DFASY or RESP, Part 1)

Entry:	**UE1**
VIT Option:	API
Event:	User exit (not DFASY or RESP, Part 1)
VIT Processing Module:	ISTRACAP
Control is Returned to:	ISTAPCUE
Applies to:	MVS VM VSE

The user exit trace record (not DFASY or RESP, Part 1) has the following format:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
UE1				ID	CODE	0		EXIT-TYPE DEPENDENT INFORMATION								PRIMARY (APPL) LU NAME								SECONDARY LU NAME OR 0							

Byte (hex)	Contents
00–03	Record ID: C'UE1'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Exit type code (see Table 35)
06–07	Zero
08–0F	Exit-type-dependent information (see Table 35)
10–17	Primary (application program) logical unit name
18–1F	Secondary logical unit name, or zero if this is a TPEND exit.

Table 35 (Page 1 of 2). Exit-Type-Dependent Information for a UE1 Entry

Exit Type Code	Exit Type	Byte (hex)	Contents
05	SCIP (BIND)	08–09	RPL control flags (RPLCNTDC, RPLCNTSC)
		0A–0F	Session parameters
05	SCIP (STSN)	08–09	RPL control flags (RPLCNTDC, RPLCNTSC)
		0A–0B	RPLOBSQV
		0C–0D	RPLIBSQV
		0E	RPLOBSQ
		0F	RPLIBSQ
05	SCIP (other)	08–09	RPL control flags (RPLCNTDC, RPLCNTSC)
		0C–0F	CID
06	LOGON	08–0B	Logon data length
		0C–0F	CID
09	LOSTERM	08–0B	Reason code
		0C–0F	CID

Table 35 (Page 2 of 2). Exit-Type-Dependent Information for a UE1 Entry

Exit Type Code	Exit Type	Byte (hex)	Contents
0A	RELREQ	08–0F	Unused
0D	TPEND	08–0B	Reason code
		0C–0F	Unused
0E	NSEXIT	08–0F	NS RU bytes 0–7

UE2 Entry for USER Exit (not DFASY or RESP, Part 2)

Entry: **UE2**
VIT Option: API
Event: User exit (not DFASY or RESP, Part 2)
VIT Processing Module: ISTRACAP
Applies to: MVS VM VSE

This trace record is a continuation of the UE1 entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
UE2				EXIT ADDRESS				0																NETID OR 0							

Byte (hex) **Contents**
00–03 Record ID: C'UE2'
04–07 Exit address
08–17 Zero
18–1F Network identifier, or zero if this is a TPEND exit.

ULKA Entry for Release All Locks

Entry:	**ULKA**
VIT Option:	LOCK
Event:	Release all locks
VIT Processing Module:	ISTRACLK
Control is Returned to:	ISTAPC39
Applies to:	MVS VM VSE

This trace record contains information about a routine releasing all locks it currently holds.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ULKA				ID	0	0		PST ADDRESS				CRA LOCK ACCOUNT WORD				RETURN ADDRESS				0								RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'ULKA'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Zero
06–07	Zero
08–0B	PST address
0C–0F	CRA lock account word (CRALKACT; see step 12 on page 79)
10–13	Address of the issuer of the TPUNLOCK macro
14–1B	Zero
1C–1F	Request parameter header (RPH) Address.

UNLK Entry for Release a Lock

Entry:	**UNLK**
VIT Option:	LOCK
Event:	Release a lock
VIT Processing Module:	ISTRACLK
Control is Returned to:	ISTAPC33
Applies to:	MVS VM VSE

This trace record contains information for a routine releasing a lock. It is the complement of a LKEX or LKSH entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
UNLK				ID	0	LOCK		LEVEL		LOCK ADDRESS		CRA LOCK ACCOUNT WORD				RETURN ADDRESS				LOCKWORD								RPH ADDRESS			

Byte (hex)	Contents
00–03	Record ID: C'UNLK'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Zero
06–07	Level of lock to be released. (See Table 7 on page 80 for a listing of VTAM locks.)
08–0B	Address of lock to be released
0C–0F	CRA lock account word (CRALKACT; see step 12 on page 79)
10–13	Address of the issuer of the TPUNLOCK macro
14–1B	Lockword pointed to by address in bytes 08–0B
1C–1F	Request parameter header (RPH) Address.

UP Entry for User Application Program Post

Entry:	**UP**
VIT Option:	API
Event:	User application program post
VIT Processing Module:	ISTRACAP
Control is Returned to:	ISTAICPT or ISTAPCSX
Applies to:	MVS VM VSE

This trace record provides the RPL information that VTAM passes to the application program. The RPL information and the API TPIO request record (AIn or IOn) show VTAM activity for a given API request.

See "RPL-Based Macroinstruction Errors and Special Conditions" in *VTAM Programming* for a description of the RPL return codes.

If this entry is associated with an event failure (that is, the return code is non-zero), this entry will be generated whether or not the API option is in effect. It is treated as an exception condition, and is always traced, regardless of the VIT options specified, if the VIT is active.

00 01	02	03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
UP	REQTY	EXTDS	ID	RTNCD	FDB2	FDB3	RPL ADDRESS	EXIT OR ECB ADDRESS	RPL AREA	RPL RLEN	CID OR 0	RPL FDBK2

Byte (hex)	Contents
00–01	Record ID: C'UP'
02	RPL request type (RPLREQ) X'15' SETLOGON X'16' SIMLOGON X'17' OPNDST X'19' CHANGE X'1A' INQUIRE X'1B' INTRPRET X'1F' CLSDST X'22' SEND X'23' RECEIVE X'24' RESETSR X'25' SESSIONC X'27' SENDCMD X'28' RCVCMD X'29' REQSESS X'2A' OPNSEC X'2C' TERMSESS
03	Exit definition (RPLEXTDS)
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM

- Task ID VSE

05	Return code (RPLRTNCD) (See "RPL-Based Macroinstruction Errors and Special Conditions" in *VTAM Programming*.)
06	Feedback code (RPLFDB2) (See "RPL-Based Macroinstruction Errors and Special Conditions" in *VTAM Programming*.)
07	Feedback data flag (RPLFDB3) (See "RPL-Based Macroinstruction Errors and Special Conditions" in *VTAM Programming* under INQUIRE macro with OPTCD=APPSTAT.)
08–0B	RPL address
0C–0F	Exit address or ECB address
10–13	RPL data area pointer (RPLAREA)
14–17	Record length (RPLRLEN)
18–1B	CID from NIB or from RPL (RPLARG), or zero
1C–1F	Second RPL feedback area — Sense code (RPLFDBK2).

VCC1 Entry for CNCB Queued (Part 1)

Entry: **VCC1**
VIT Option: VCNS
Event: CNCB queued (Part 1)
VIT Processing Module: ISTRACNS
Applies to: MVS VM

The VCC1 trace record is written when a CNS protocol control block (CNCB) is queued between CNS and LSA components or VM between CNS and CNSMAP. The VCC2 trace record displays data contained in the CNCB.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
VCC1				ID	CBID	STATE	CCODE	MAJOR CONTROL BLOCK ADDRESS				CNCB ADDRESS				REQSTNG MODULE ABBREV NAME				12 BYTES OF CNCB DATA											

Byte (hex) Contents

00–03 Record ID: C'VCC1'

04 ID is one of the following:

- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM

05 Control block identifier

X'0B' ISTVLNCB
X'7B' ISTRIB

06 Process state of the control block processing the dequeued work element:

LSNCB states when CBID=X'0B' (ISTVLNCB)

X'00' Reset
X'01' Pending active
X'02' Active without application logged on to line
X'03' Active with application logged on to line
X'04' Pending inactive phase 1
X'05' Pending inactive phase 2
X'06' Pending inactive phase 3
X'07' Blocked

States when CBID=X'7B' (ISTRIB)

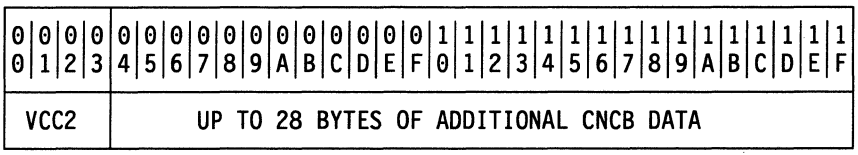
X'00' Idle
X'01' Logon in progress
X'02' Active
X'03' Logoff in progress
X'04' DACTLINK in progress
X'05' Forced termination in progress

07	CNCB primitive code X'01' CNS_INQUIRE_REQUEST X'02' CNS_INQUIRE_CONFIRM X'03' CNS_LOGON_REQUEST X'04' CNS_LOGON_CONFIRM X'05' CNS_LOGOFF_REQUEST X'06' CNS_LOGOFF_CONFIRM X'07' CNS_LOGOFF_RESPONSE X'08' CNS_LOGOFF_INDICATION X'09' CNS_CLOSEACB_REQUEST X'0A' CNS_CLOSEACB_CONFIRM X'0B' CNS_CLOSEACB_COMPLETION_INDICATION
08–0B	Address of the major control block
0C–0F	CNCB address
10–13	Requesting module abbreviated name
14–1F	First 12 bytes of CNCB data.

VCC2 Entry for CNCB Queued (Part 2)

Entry: **VCC2**
VIT Option: VCNS
Event: CNCB queued (Part 2)
VIT Processing Module: ISTRACNS
Applies to: MVS VM

This trace record is a continuation of the VCC1 trace record. Up to two VCC2 trace records might follow a VCC1 trace record, depending on the length of the primitive that is traced.



Byte (hex)	Contents
00–03	Record ID: C'VCC2'
04–1F	Up to 28 bytes of additional CNCB data.

VCDQ Entry for Work Element Dequeued

Entry: **VCDQ**
VIT Option: VCNS
Event: Work element dequeued
VIT Processing Module: ISTRACNS
Control is Returned to: Module invoking the INTRACE macro that caused the record to be produced.
Applies to: MVS VM

This trace record is written whenever a work element is dequeued for processing. The VCDQ trace record identifies the work element being dequeued.

0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	0 A	0 B	0 C	0 D	0 E	0 F	1 0	1 1	1 2	1 3	1 4	1 5	1 6	1 7	1 8	1 9	1 A	1 B	1 C	1 D	1 E	1 F
VCDQ				ID	STATE	CBID	WK EL	WORK ELEMENT TYPE				RU TYPE OR 0				WORK ELEMENT ADDRESS				CAB ADDRESS OR 0				0				RPH ADDRESS			

Byte (hex) Contents

00–03 Record ID: C'VCDQ'

04 ID is one of the following:

- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM

05 Process state of the associated connection or of the control block processing the dequeued work element:

States when CBID=X'0B' (ISTVLNCB)

X'00' Reset state
X'10' ACTLINK pending
X'20' Active state
X'30' DACTLINK pending

States when CBID=X'0C' (ISTPCLCB)

X'00' Reset
X'01' Logon complete
X'02' Logon pending
X'03' Logoff pending
X'04' Adapter termination pending

States when CBID=X'25' (ISTVCCB)

X'01' P1 Ready
X'02' P2 DTE waiting
X'03' P3 DCE waiting
X'04' P4 Data transfer
X'05' P5 Call collision
X'06' P6 DTE clear request
X'07' P7 DCE clear indication
X'11' D1 Flow control ready
X'12' D2 DTE reset request

X'13' D3 DCE reset indication
X'21' I1 No interrupt pending
X'22' I2 DTE interrupt pending
X'31' E1 No error reset
X'32' ER CNS error reset
X'33' CF CNS cancel forward

States when CBID=X'66' (ISTCAB)

X'00' Idle
X'01' Connect requested
X'02' Connect indicated
X'03' Flow control ready
X'04' Error reset
X'05' Cancel hold
X'06' Hold purge
X'07' Reset requested
X'08' Reset indicated
X'09' Error disconnect
X'0A' Cleanup
X'0B' Disconnect requested
X'0C' Disconnect indicated
X'0D' Pending error disconnect

States when CBID=X'6B' (ISTCAR)

No states are traced when ISTCAR is the major control block.

States when CBID=X'7B' (ISTRIB)

X'00' Idle
X'01' Logon in progress
X'02' Active
X'03' Logoff in progress
X'04' DACTLINK in progress
X'05' Forced termination in progress

06 Control block identifier
X'0B' ISTVLNCB
X'0C' ISTPCLCB
X'25' ISTVCCB
X'66' ISTCAB
X'6B' ISTCAR
X'7B' ISTRIB

07 If work element is an 'RPL' (see bytes 08-0B)
X'08' CONTROL=INQUIRE
X'10' CONTROL=LOGON
X'14' CONTROL=TEST
X'18' CONTROL=XID
X'20' CONTROL=LOGOFF
X'30' CONTROL=READ
X'40' CONTROL=STATUS
X'50' CONTROL=SET
X'60' CONTROL=OPEN
X'70' CONTROL=CLOSE
X'80' CONTROL=SEND
X'90' CONTROL=SUSPEND

	X'A0'	CONTROL=RESUME
	X'B0'	CONTROL=EXPEDITE
	X'C0'	CONTROL=CHECK
	X'D0'	CONTROL=RESET
	X'E0'	CONTROL=SETCPARM
	X'F0'	CONTROL=RECEIVE
	X'FC'	CONTROL=REPLY
07	If work element is a 'PICB' (see bytes 08-0B)	
	X'01'	LOGON_REQUEST
	X'02'	LOGON_CONFIRM
	X'03'	LOGOFF_REQUEST
	X'05'	CNS_LOGON_REQUEST
	X'06'	CNS_LOGON_CONFIRM
	X'07'	CNS_LOGOFF_REQUEST
	X'08'	CNS_LOGOFF_CONFIRM
	X'09'	CNS_LOGOFF_RESPONSE
	X'0A'	CNS_LOGOFF_INDICATION
	X'0B'	CNS_CLOSEACB_REQUEST
	X'0C'	CNS_CLOSEACB_CONFIRM
	X'0D'	CNS_CLOSEACB_COMPLETION_INDICATION
	X'11'	CONNECT_REQUEST
	X'12'	CONNECT_CONFIRM
	X'13'	CONNECT_ACCEPT
	X'14'	CONNECT_INDICAT
	X'15'	DISCONNECT_REQUEST
	X'16'	DISCONNECT_INDICAT
	X'17'	DISCONNECT_CONFIRM
	X'18'	ERROR_DISCONNECT_INDICAT
	X'19'	RESTART_INDICAT
	X'1A'	CNS_CLEAR
	X'1B'	ERROR_DISCONNECT_INDICATION (TIMER)
	X'21'	RESET_REQUEST
	X'22'	RESET_CONFIRM
	X'23'	RESET_ACCEPT
	X'24'	RESET_INDICAT
	X'25'	ERROR_RESET_INDICAT
	X'31'	DATA_OUT (LEVEL1)
	X'32'	DATA_IN (LEVEL1)
	X'33'	DATA_OUT (LEVEL2)
	X'34'	DATA_IN (LEVEL2)
	X'35'	INTERRUPT_OUT
	X'36'	INTERRUPT_CONFIRM
	X'37'	INTERRUPT_INDICATION
	X'42'	STATUS_INDICAT FINAL TERMINAT
	X'53'	SUSPEND_REQUEST
	X'54'	RESUME_REQUEST
	X'55'	RNR_IN
	X'56'	RR_IN
	X'57'	ROTATE
	X'62'	TIMER_EVENT
	X'63'	RESTART_VC_TRIGGER
	X'71'	RESET_PENDING_TRIGGER
	X'72'	RESET_COMPLETE_TRIGGER

X'73' CLEAR_PENDING_TRIGGER
X'74' CLEAR_COMPLETE_TRIGGER
X'75' CALL_PENDING_TRIGGER
X'76' CALL_COMPLETE_TRIGGER
X'77' INTERRUPT_PENDING_TRIGGER
X'78' ERROR_RESET_TRIGGER
X'79' ERROR_DISCONNECT_TRIGGER
X'82' PACKET_IN
X'91' TERMINATE_PORT
X'E0' CNS_NMVT
X'F0' READ_REQ
X'F1' READ_CONFIRM
X'F2' STATUS_PHYSICAL_REQUEST
X'F3' STATUS_PHYSICAL_CONFIRM
X'F4' STATUS_VIRTUAL_REQ
X'F5' STATUS_VIRTUAL_CONFIRM

If work element is a 'CNCB' (see bytes 08-0B)
X'01' CNS_INQUIRE_REQUEST
X'02' CNS_INQUIRE_CONFIRM
X'03' CNS_LOGON_REQUEST
X'04' CNS_LOGON_CONFIRM
X'05' CNS_LOGOFF_REQUEST
X'06' CNS_LOGOFF_CONFIRM
X'07' CNS_LOGOFF_RESPONSE
X'08' CNS_LOGOFF_INDICATION
X'09' CNS_CLOSEACB_REQUEST
X'0A' CNS_CLOSEACB_CONFIRM
X'0B' CNS_CLOSEACB_COMPLETION_INDICATION

08–0B Work element type:
C'CNCB' CNCB
C'RPL' RPL
C'PICB' PICB
C'RUPE' RUPE
C'VCCB' VCCB
C'CAB' CAB
C'RIB' RIB
C'RPH' RPH
C'TSCB' TSCB
C'UECB' UECB
C'????' None of the above

0C–0F If work element is a 'RUPE', one of the following; otherwise, zero
X'0801020A' ACTLINK
X'0801020B' DACTLINK

10–13 Work element address

14–17 If work element is 'PICB', CAB address; otherwise, zero

18–1B Zero

1C–1F Request parameter header (RPH) address.

VCP1 Entry for PICB Queued (Part 1)

Entry:	**VCP1**
VIT Option:	VCNS
Event:	PICB queued (Part 1)
VIT Processing Module:	ISTRACNS
Control is Returned to:	Modules invoking the INTRACE macro that caused the record to be produced
Applies to:	VM

This trace record is written whenever packet-level interface control blocks (PICBs) are queued between components. The VCP2 trace record displays data contained in the PICB.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
VCP1				ID	STATE	CBID	TYPE	SEQNO	FLAGS			FORMAT DEPEND DATA				PICB ADDRESS				CAB ADDRESS				CONT POINTER				LEN			0

Byte (hex) Contents

00–03 Record ID: C'VCP1'

04 Machine ID and Task ID

05 Process state of associated connection or of control block process queuing PICB

States when CBID=X'0B' (ISTVLNCB)

X'00' Reset state
X'10' ACTLINK pending
X'20' Active state
X'30' DACTLINK pending

States when CBID=X'0C' (ISTPCLCB)

X'00' Reset
X'01' Logon complete
X'02' Logon pending
X'03' Logoff pending
X'04' Adapter termination pending

States when CBID=X'25' (ISTVCCB)

X'01' P1 Ready
X'02' P2 DTE waiting
X'03' P3 DCE waiting
X'04' P4 Data transfer
X'05' P5 Call collision
X'06' P6 DTE clear request
X'07' P7 DCE clear indication
X'11' D1 Flow control ready
X'12' D2 DTE reset request
X'13' D3 DCE reset indication
X'21' I1 No interrupt pending

X'22' I2 DTE interrupt pending
X'31' E1 No error reset
X'32' ER CNS error reset
X'33' CF CNS cancel forward

States when CBID=X'66' (ISTCAB)

X'00' Idle
X'01' Connect request
X'02' Connect indicated
X'03' Flow control ready
X'04' Error reset
X'05' Cancel hold
X'06' Hold purge
X'07' Reset requested
X'08' Reset indicated
X'09' Error disconnect
X'0A' Cleanup
X'0B' Disconnect requested
X'0C' Disconnect indicated
X'0D' Pending error disconnect

States when CBID=X'6B' (ISTCAR)

No states are traced when ISTCAR is the major control block.

States when CBID=X'7B' (ISTRIB)

X'00' Idle
X'01' Logon in progress
X'02' Active
X'03' Logoff in progress
X'04' DACTLINK in progress
X'05' Forced termination in progress

06 Control block identifier
X'0B' ISTVLNCB
X'0C' ISTPCLCB
X'25' ISTVCCB
X'66' ISTCAB
X'6B' ISTCAR
X'7B' ISTRIB

07 PICB type
X'01' LOGON_REQUEST
X'02' LOGON_CONFIRM
X'03' LOGOFF_REQUEST
X'05' CNS_LOGON_REQUEST
X'06' CNS_LOGON_CONFIRM
X'07' CNS_LOGOFF_REQUEST
X'08' CNS_LOGOFF_CONFIRM
X'09' CNS_LOGOFF_RESPONSE
X'0A' CNS_LOGOFF_INDICATION
X'0B' CNS_CLOSEACB_REQUEST
X'0C' CNS_CLOSEACB_CONFIRM
X'0D' CNS_CLOSEACB_COMPLETION_INDICATION
X'11' CONNECT_REQUEST
X'12' CONNECT_CONFIRM

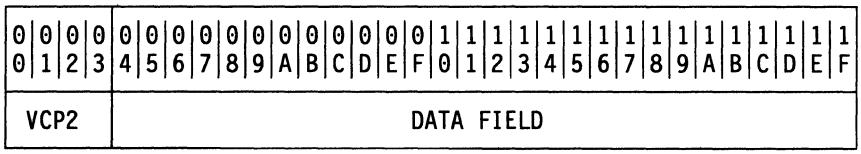
	X'13'	CONNECT_ACCEPT
	X'14'	CONNECT_INDICAT
	X'15'	DISCONNECT_REQUEST
	X'16'	DISCONNECT_INDICAT
	X'17'	DISCONNECT_CONFIRM
	X'18'	ERROR_DISCONNECT_INDICAT
	X'19'	RESTART_INDICAT
	X'1A'	CNS_CLEAR
	X'1B'	ERROR_DISCONNECT_INDICATION (TIMER)
	X'21'	RESET_REQUEST
	X'22'	RESET_CONFIRM
	X'23'	RESET_ACCEPT
	X'24'	RESET_INDICAT
	X'25'	ERROR_RESET_INDICAT
	X'31'	DATA_OUT (LEVEL1)
	X'32'	DATA_IN (LEVEL1)
	X'33'	DATA_OUT (LEVEL2)
	X'34'	DATA_IN (LEVEL2)
	X'35'	INTERRUPT_OUT
	X'36'	INTERRUPT_CONFIRM
	X'37'	INTERRUPT_INDICATION
	X'42'	STATUS_INDICAT_FINAL_TERMINAT
	X'53'	SUSPEND_REQUEST
	X'54'	RESUME_REQUEST
	X'55'	RNR_IN
	X'56'	RR_IN
	X'57'	ROTATE
	X'62'	TIMER_EVENT
	X'63'	RESTART_VC_TRIGGER
	X'71'	RESET_PENDING_TRIGGER
	X'72'	RESET_COMPLETE_TRIGGER
	X'73'	CLEAR_PENDING_TRIGGER
	X'74'	CLEAR_COMPLETE_TRIGGER
	X'75'	CALL_PENDING_TRIGGER
	X'76'	CALL_COMPLETE_TRIGGER
	X'77'	INTERRUPT_PENDING_TRIGGER
	X'78'	ERROR_RESET_TRIGGER
	X'79'	ERROR_DISCONNECT_TRIGGER
	X'82'	PACKET_IN
	X'91'	TERMINATE_PORT
	X'E0'	CNS_NMVT
	X'F0'	READ_REQ
	X'F1'	READ_CONFIRM
	X'F2'	STATUS_PHYSICAL_REQUEST
	X'F3'	STATUS_PHYSICAL_CONFIRM
	X'F4'	STATUS_VIRTUAL_REQ
	X'F5'	STATUS_VIRTUAL_CONFIRM
08	PICB sequence number	
09	PICB CNS flags	
	Bit	Meaning
	1...	Qualified data indicator
	.1..	Delivery confirmation requested
	..1.	Delivery confirmed

	...1 .... More data indicator
	.... 000. PICB format 0
	.... 001. PICB format 1
	.... 010. PICB format 2
	.... 011. PICB format 3
	.... 100. PICB format 4
	....1 No disconnect confirmation allowed to CNS
0A–0B	Zero
0C–0F	For PICB format 1, PICB RIB address
0C–0F	For PICB format 2, PICB connection ID
0C–0F	For PICB formats 3 and 4, PICB major and minor return codes
10–13	PICB address
14–17	PICB CAB address
18–1B	PICB continuation pointer
1C–1D	PICB data length including continuation PICBs
1E–1F	Zero.

VCP2 Entry for PICB Queued (Part 2)

Entry:	**VCP2**
VIT Option:	VCNS
Event:	PICB queued (Part 2)
VIT Processing Module:	ISTRACNS
Control is Returned to:	Modules invoking the INTRACE macro that caused the record to be produced
Applies to:	VM

This trace record is a continuation of the VCP1 trace record. This record is written only when the PICB includes data.



Byte (hex)	Contents
00–03	Record ID: C'VCP2'
04–1F	Twenty-eight bytes of PICB data.

VPST Entry for VPOST Macro

Entry:	**VPST**
VIT Option:	PSS
Event:	VPOST macro
VIT Processing Module:	ISTRACPS
Control is Returned to:	Module invoking the INTRACE macro that caused the record to be produced.
Applies to:	MVS VM VSE

This trace record is written when a VTAM module issues a VPOST macro. The VPOST macro generates a system POST event.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
VPST				ID	0			ASCB ADDRESS OR 0				ECB ADDRESS				0				COMPLTE CODE OR 0				ISSUER ADDRESS				0			

Byte (hex)	Contents
00–03	Record ID: C'VPST'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05–07	Zero
08–0B	Address of the ASCB MVS or 0
0C–0F	Address of the ECB being posted
10–13	Zero
14–17	Completion code or 0
18–1B	Address of the issuer of the VPOST macro
1C–1F	Zero.

VRSM Entry for Resume after VWAIT

Entry: **VRSM**
VIT Option: PSS
Event: VWAIT macro
VIT Processing Module: ISTRACPS
Applies to: MVS VM VSE

This trace record is written when VTAM resumes control following a system WAIT event generated by the VTAM VWAIT macro.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
VRSM				ID	0			0				ECB ADDRESS				0				0				ISSUER ADDRESS				0			

Byte (hex) Contents

00–03 Record ID: C'VRSM'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05–0B Zero

0C–0F Address of the ECB that was posted

10–17 Zero

18–1B Address of the code that is resuming execution

1C–1F Zero.

VTAL Entry for Allocate Storage

Entry:	**VTAL**
VIT Option:	SMS
Event:	Allocate storage
VIT Processing Module:	ISTRACSM
Control is Returned to:	MVS ISTORMVA VM ISTORAVF VSE ISTOREAF
Applies to:	MVS VM VSE

This trace record provides the status of each VTALLOC request issued by VTAM components.

If this entry is associated with an event failure (that is, the return code is non-zero), this entry will be generated whether or not the SMS option is in effect. It is treated as an exception condition and is always traced, regardless of the VIT options specified, if the VIT is active.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
VTAL				ID	FLAG	0		STORAGE ADDRESS				SUBPOOL NUMBER OR POOL ID				RETURN ADDRESS				LENGTH OF STORAGE REQUEST				CALLER OF UTILITY OR 0				RETURN CODE			

Byte (hex)	Contents
00–03	Record ID: C'VTAL'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05	Flags Bit Meaning 0... .... FREED(EXPLICIT) storage request 1... .... FREED(TASK) or FREED(JSTASK) storage request
06–07	Zero
08–0B	Address of area allocated
0C–0F	ID MVS Subpool number of the area allocated VM Buffer pool ID VSE Area where storage is allocated SGA System GETVIS area PGU Private GETVIS area in user partition PGV Private GETVIS area in VTAM partition
10–13	Address of the issuer of the VTALLOC macro
14–17	Length of area allocated

18–1B	If the VTALLOC macro is issued by a utility routine, this location contains the address of the issuer of the utility routine. If the VTALLOC macro is not issued by a utility routine, this location contains zero.
1C–1F	VTALLOC return code.

VTFR Entry for Free Storage

Entry: **VTFR**
VIT Option: SMS
Event: Free storage
VIT Processing Module: ISTRACSM
Control is Returned to: MVS ISTORMVF
VM ISTORAVF
VSE ISTOREAF
Applies to: MVS VM VSE

This trace record provides the status of each VTFREE request issued by VTAM components.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
VTFR				ID	0			STORAGE ADDRESS				SUBPOOL NUMBER OR POOL ID				RETURN ADDRESS				LENGTH OF STORAGE FREED				CALLER OF UTILITY OR 0				0			

Byte (hex) Contents

00–03 Record ID: C'VTFR'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05–07 Zero

08–0B Address of area to be freed

0C–0F Number or ID
MVS Subpool number of the area allocated
VM VSE Buffer pool ID

10–13 Address of the issuer of the VTFREE macro

14–17 Length of area to be freed

18–1B If the VTFREE macro is issued by a utility routine, this location contains the address of the issuer of the utility routine.
If the VTFREE macro is not issued by a utility routine, this location contains zero.

1C–1F Zero.

VWAI Entry for VWAIT Macro

Entry:	**VWAI**
VIT Option:	PSS
Event:	VWAIT macro
VIT Processing Module:	ISTRACPS
Control is Returned to:	Module invoking the INTRACE macro that caused the record to be produced.
Applies to:	MVS VM VSE

This trace record is written when a VTAM module issues a VWAIT macro. The VWAIT macro generates a system WAIT event.

If the "last address in ECB list" field is not zero, then this is a wait on an ECB list. Note that an ECB list is a list of the addresses of ECBs, not a list of ECBs.

0 0 0 0 0 1 2 3	0 4	0 0 0 0 5 6 7	0 0 0 0 8 9 A B	0 0 0 0 C D E F	1 1 1 1 0 1 2 3	1 1 1 1 4 5 6 7	1 1 1 1 8 9 A B	1 1 1 1 C D E F
VWAI	ID	0	0	ECB ADDRESS OR ECB LIST ADDRESS	LAST ADDRESS IN ECB LIST OR 0	0	ISSUER ADDRESS	0

Byte (hex)	Contents
00–03	Record ID: C'VWAI'
04	ID is one of the following: • Primary address space ID MVS This field is zero if the ID is greater than X'FF'. • Machine ID and Task ID VM • Task ID VSE
05–0B	Zero
0C–0F	Address of the ECB that is being waited on, or (if the next field is non-zero) the address of an ECB list.
10–13	Address of the last pointer in an ECB list, or 0. If this field is 0, it implies that the previous field points to an ECB instead of an ECB list.
14–17	Zero
18–1B	Address of the issuer of the VWAIT macro
1C–1F	Zero.

WAIT Entry for TPWAIT Macro

Entry:	**WAIT**
VIT Option:	PSS
Event:	TPWAIT macro
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCTW
Applies to:	MVS VM VSE

This trace record identifies an RPH that has been suspended to wait for some VTAM resource. This entry might be used with the entry immediately preceding it in the table to determine whether the wait is for storage or for a lock. If the preceding entry is for a REQSTORE, a waiting-for-storage condition may exist. If the preceding entry is for a lock request, the wait can be caused by that request. However, VTAM frequently uses a TPWAIT/TPPOST to synchronize intercomponent functions. Therefore, the TPWAIT entry does not necessarily identify an unusual condition.

0 0 0 0 0 1 2 3	0 4	0 5	0 6 7	0 8 9 A B	0 C D E F	1 0 1 2 3	1 4 5 6 7	1 8 9 A B	1 C D E F
WAIT	ID	0	FLAGS	PST ADDRESS	PAB ADDRESS	RETURN ADDRESS	WORK EL Q OR NEXT DSP Q LVL	MODULE NAME OR DVT ADDRESS	RPH ADDRESS

Byte (hex) Contents

00–03 Record ID: C'WAIT'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Zero

06 PAB flag field (PABFLAGS)

Bit	Meaning
1...	PAB is unconditionally scheduled.
.1..	PAB closedown is in progress.
..1.	PAB is synchronous.
...1	PAB extension is present.
.... 1...	Do not dequeue work element.
.... .1..	Do not detach the RPH.
.... ..1.	Indicates a very extended PAB.
....1	Indicates a slightly extended PAB.

07 PAB flag field (PABFLGS1)

Bit	Meaning
1...	Switch the PST address of this PAB's major control block to the new PST address contained in DYPNWPST.
.1..	This PAB has a data space extension.
..1.	This PAB's major control block is an FMCB.
...1	PAB can be referenced in PSW disable mode.
.... 1...	PAB is persistent.

	.... .1.. APSTERM/APSINIT FMCB during PAB dispatch. .... ..xx Reserved.
08–0B	PST address
0C–0F	PAB address
10–13	Address of the issuer of the TPWAIT macro
14–17	PAB work element queue or next dispatchable queue level for a very extended PAB
18–1B	Module name abbreviation (bytes 4, 5, 7, and 8 of the name of the next module to get control), PAB DVT address (high order bit of X'18' = 0), function, or process.
1C–1F	Request parameter header (RPH) Address.

XID Entry for Exchange ID (Part 1)

Entry: **XID**
VIT Option: CIO
Event: Exchange ID (Part 1)
VIT Processing Module: ISTRACCI
Control is Returned to: ISTTSCCU or ISTTSC8X
Applies to: MVS VM VSE

This trace record contains information about inbound or outbound exchange ID processing.

0 0 0 0 0 1 2 3	0 0 0 0 0 4 5 6 7	0 0 0 0 0 8 9 A B	0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 C D E F 0 1 2 3 4 5 6 7 8 9 A B C D E F
XIDI XIDO	NCB ADDRESS	BPB ADDRESS or 0	XID DATA

Byte (hex)	Contents
00–03	Record ID: C'XIDI' for inbound XID C'XIDO' for outbound XID
04–07	NCB address
08–0B	Boundary physical unit block (BPB) address or zero
0C–1F	20 bytes of XID data.

XID2 Entry for Exchange ID (Part 2)

Entry:	**XID2**
VIT Option:	CIO
Event:	Exchange ID (Part 2)
VIT Processing Module:	ISTRACCI
Applies to:	MVS VM VSE

This trace record is a continuation of the XID trace record.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
XID2				ADDITIONAL XID DATA																											

Byte (hex)	Contents
00–03	Record ID: C'XID2'
04–1F	28 bytes of additional XID data.

XPST Entry for TPSWPST Macro

Entry:	**XPST**
VIT Option:	PSS
Event:	TPSWPST macro
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCSP
Applies to:	MVS VM VSE

This trace record identifies a VTAM process (PAB) that is switching PSTs during termination of a task. Switching PSTs causes the PAB to be scheduled for dispatch under the new PST.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
XPST				ID	0		FLAGS	NEW PST ADDRESS				PAB ADDRESS				ISSR				WEQ				MOD NAME OR DVT ADDRESS				RPH ADDRESS OR 0			

Byte (hex) Contents

00–03 Record ID: 'XPST'

04 ID is one of the following:
- Primary address space ID MVS
  This field is zero if the ID is greater than X'FF'.
- Machine ID and Task ID VM
- Task ID VSE

05 Zero

06 PAB flag field (PABFLAGS)

Bit	Meaning
1...	PAB is unconditionally scheduled.
.1..	PAB closedown is in progress.
..1.	PAB is synchronous.
...1	PAB extension is present.
.... 1...	Do not dequeue work element.
.... .1..	Do not detach the RPH.
.... ..1.	Indicates a very extended PAB.
....1	Indicates a slightly extended PAB.

07 PAB flag field (PABFLGS1)

Bit	Meaning
1...	Switch the PST address of this PAB's major control block to the new PST address contained in DYPNWPST.
.1..	This PAB has a data space extension.
..1.	This PAB's major control block is an FMCB.
...1	PAB can be referenced in PSW disable mode.
.... 1...	PAB is persistent.
.... .1..	APSTERM/APSINIT FMCB during PAB dispatch.
.... ..xx	Zero.

08–0B New PST address under which this PAB will now be dispatched

0C–0F PAB address

10–13 Address of the issuer of the TPSWPST macro

14–17	PAB work element queue
18–1B	Module name abbreviation or PAB DVT address
1C–1F	Request parameter header (RPH) Address or zero.

Appendix B. Finding VTAM Buffer Pools (VM, VSE)

This appendix describes general procedures for locating VTAM buffer pools. Included are the following sections:

- "Finding VTAM Buffer Pools (VM)"
- "Finding VTAM Buffer Pools (VSE)" on page 839
- "Relative Locations of VTAM Buffer Pools (VSE)" on page 842.

Some types of problems may require that you collect and supply information for specific buffer pools.

MVS For information on solving storage problems on MVS, see "Storage Problem Procedure (MVS)" on page 105.

Finding VTAM Buffer Pools (VM)

Use the following procedure to find VTAM buffer pools and to determine whether any have used all their buffers.

Dumps processed with PRTDUMP contain all the data necessary to use this procedure if VTAM control block formatting is requested. To determine what book contains more information on processing dumps using PRTDUMP, see Table 80 on page 1165.

1. From the ATCVT, find the address of the buffer pool directory (BPDTY) in field ATCBPDA.
2. From the BPDTY, find the address of the first buffer pool entry (BPENT) in field BPDENTRY. There are 9 BPENTs, one for each buffer pool; each is 10 hex bytes in length and contains the name of a buffer pool at hex offset 00, and the address of the buffer pool control block (BPCB) at hex offset 08.
3. Find each BPCB. Each BPCB represents a buffer pool.

Example: Finding LPBUF (VM)

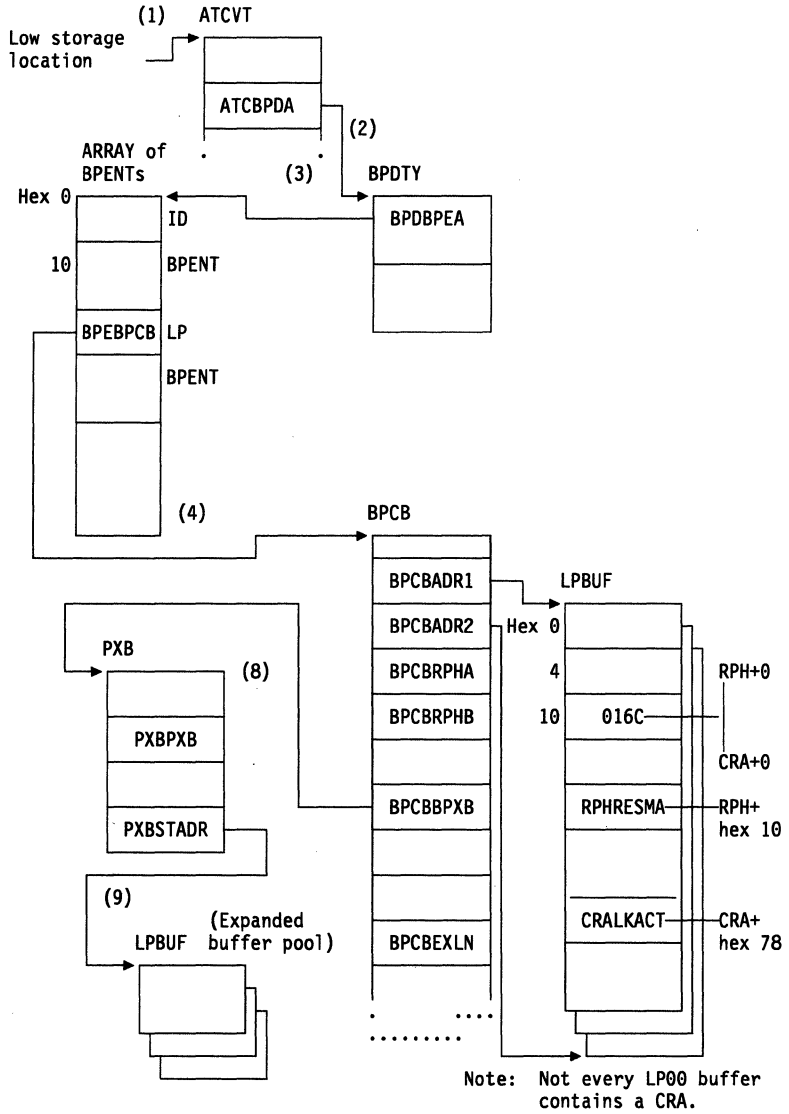


Figure 81. Relationships among Buffer Pool Control Blocks VM

Use the following steps to find the large pageable buffer pool (LPBUF). LPBUF is the buffer pool that contains the CRAs and RPHs. Figure 81 shows the control block relationships involved in these steps.

1. Find the ATCVT. The field ATCLCPTR points to the address of the ATCVT. ATCLCPTR is at low-storage location hex 200.

   The ATCVT is identified by release level at hex offset 00 in the ATCVT. For VTAM V4R2 the identifier is VA42 (X'E5C1F4F240404040').

2. From the ATCVT, find the address of the first buffer pool control block (BPCB) in field ISTBPCBA (BPDTY + X'2C').

3. There are 9 BPCBs, one for each buffer pool. Each BPCB is 248 bytes in length. The large pageable buffer pool (LP00) is the third entry in the array of BPCBs.

4. The beginning address of the static portion of the LP00 buffer pool is at BPCB + X'14'(BPCBADR1); the end of the static portion is pointed to by BPCB + X'18' (BPCBADR2).
5. The first buffer begins at the address found in the BPCB + X'14' (BPCBADR1). To verify the correct buffer pool, hex offset 08 (BFPBCB) in each buffer should have the address of the BPCB found in step 3 on page 838. A buffer is allocated when the X'80' bit (BFPALLOC) at offset X'4' in each buffer is ON.
6. To find the subsequent buffers, add hex 800 to the address of the previous buffer. Two buffers are on each 4K page of storage.

   When you reach the address in BPCB + X'18' (BPCBADR2), you have found the end of the LP00 buffer pool.
7. To find LP00 buffer pool expansion areas, use the address of the pool extension block (PXB) at BPCB + X'94' (BPCBBPXB). If the address is zero, there are no expansions; the buffer pool has not been expanded since VTAM was started. In this case, skip step 8 and 10. If there is an address, the buffer pool has been expanded dynamically and the expansions need to be examined; continue with the next step.
8. Find the PXB, pointed to by BPCB + X'94' (BPCBBPXB). The PXB hex offset 0C (PXBSTADR) is the beginning of the pool expansion area. If PXBSTADR=0, go to step 10. PXBs are not freed by VTAM; only the area containing buffers is freed. The length of the expansion area (BPCBEXLN) is found at BPCB + X'A8'. Add the length (BPCBEXLN) to the starting address (PXBSTADR) for the end of the expansion area.
9. Go through the buffers as directed before in steps 5 and 6, where the first buffer begins at PXBSTADR (instead of BPCBADR1) and the end of the extension area is pointed to by PXBSTADR + BPCBEXLN (instead of BPCBADR2).
10. Find the next PXB, pointed to by PXB + X'04' (PXBPXB). If PXBPXB≠0, proceed as in step 9. If PXBPXB=0, there are no more expansion areas; the search is complete.

Finding VTAM Buffer Pools (VSE)

Use the following steps to find the buffer pools and determine whether any have used all of their available buffers. Figure 82 on page 840 shows the control blocks involved in these steps.

1. Find the ATCVT. Its address is at low-storage location hex 408.

   If this low-storage location is not available in a dump,

   a. Find the SYSCOM pointer at lowcore X'80'.

   b. Find the AVT pointer at SYSCOM + X'60'.

   c. AVT + X'00' points to the ATCVT.

   The ATCVT is identified by release level at hex offset 00 in the ATCVT. For VTAM V4R2 the identifier is VS42 (X'E5E2F4F240404040').
2. The ATCVT is immediately preceded in storage by the buffer pool directory (BPDIR) and the fixed-length pool PCBs, which can be identified by the pool

IDs in EBCDIC. ("Relative Locations of VTAM Buffer Pools (VSE)" shows the relative locations of the buffer pools.)

ATCVT field ATCBPDA + X'414' also points to the BPDIR.

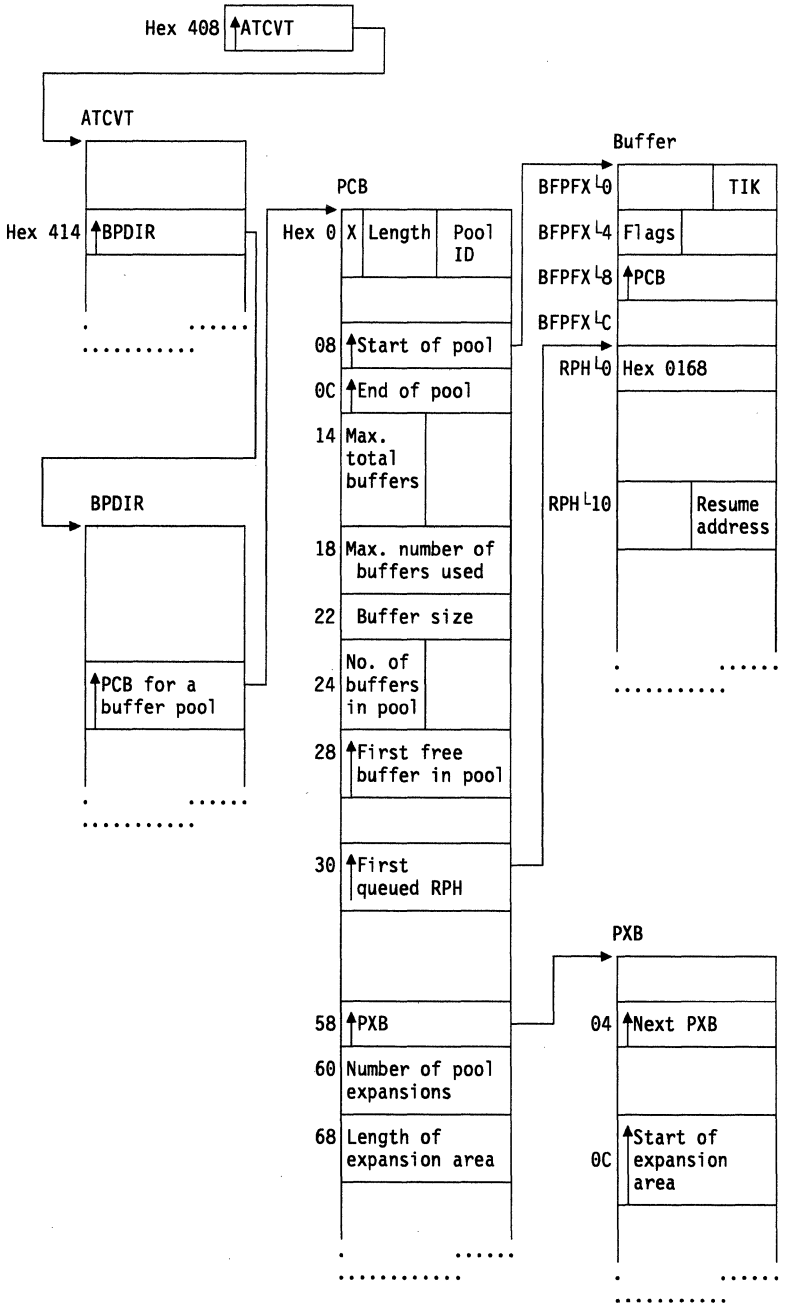


Figure 82. Relationships among Buffer Pool Control Blocks VSE

Example: Finding LPBUF (VSE)

Use the following steps to find the large pageable buffer pool (LPBUF) in VSE. LPBUF is the buffer pool that contains the CRA/RPHs. Figure 82 on page 840 shows the control block relationships involved in these steps. ("Relative Locations of VTAM Buffer Pools (VSE)" on page 842 has more information on the location of VTAM buffer pools in VSE.)

1. Find the ATCVT.
   a. Find the SYSCOM pointer at lowcore X'80'.
   b. Find the AVT pointer at SYSCOM + X'60'.
   c. AVT + X'00' points to the ATCVT.

   The ATCVT is identified by release level at hex offset 00 of the ATCVT. For VTAM V4R2 the identifier is VS42 (X'E5E2F4F240404040').
2. From the ATCVT find the address of the buffer pool directory (BPDIR) in field ATCBPDA (ATCVT + X'414').
3. From the pointer at BPDIR + X'14', find the buffer pool control block (PCB) for the large pageable buffer pool (LP). At hex offset 08 in the PCB is the address of the start of the LP pool.
4. To find the next buffer, add hex 550 to the address of the buffer prefix (BFPFX) of the previous buffer. Continue with this process until the end of the LP buffer pool (at the address in PCB + X'0C').

   Each buffer (including the prefix) is 1360 (hex 550) bytes, and the buffers can cross page boundaries. The buffer size is also given in PCBBUFSZ (PCB + X'22').

   The total number of buffers is given in PCBTOTNO (PCB + X'24').
5. To find the LP buffer pool expansion areas, use the address of the pool extension block (PXB) at PCB + X'58'. If the address is zero, there are no expansions; the buffer pool has not been expanded since VTAM was started. In this case, skip steps 6 and 7. If there is an address, the buffer pool has been expanded dynamically and the expansions need to be examined; continue with the next step.
6. Find the PXB, pointed to by PCB + X'58'. The PXB hex offset 0C (PXBSTADR) is the beginning of the pool expansion area. If PXBSTADR=0, go to step 7. PXBs are not freed by VTAM; only the area containing buffers is freed. The length of the expansion area (PCBEXLN) is found at PCB + X'68'. Add the length (PCBEXLN) to the starting address (PXBSTADR) for the end of the expansion area.
7. Find the next PXB, pointed to by PXB + X'04' (PXBPXB). If PXBPXB≠0, proceed as in step 6. If PXBPXB=0, there are no more expansion areas; the search is complete.

Relative Locations of VTAM Buffer Pools (VSE)

To find a particular VTAM buffer pool in a VSE system, see the partition map in Figure 83 on page 843 and follow these steps:

1. Find the ATCVT at low-storage location hex 408, if available, or from hex offset 00 in the AVT.
2. Find the buffer pool directory (BPDIR) from ATCVT + X'414' (ATCBPDA), or by subtracting hex 30 from the ATCVT address.
3. Find the pool control blocks (PCBs) from the pointers in BPDIR, or by finding the EBCDIC pool ID preceding the BPDIR in the storage dump (for fixed-length pools only).

For all buffer pools, the beginning and ending addresses of the initial pool extent are contained in the PCB at hex offsets 08 (PCBSTGA) and 0C (PCBSTGE), respectively.

For fixed-length pools, the storage area may consist of one or more extents. The initial extent allocated when VTAM is started contains the number of buffers indicated as a start parameter. Additional extents may be allocated if dynamic expansion is allowed for the pool. See "Buffer Pools" in the *VTAM Network Implementation Guide* for more information on buffer pool extents.

To find the buffers in a fixed-length pool, start with the buffer prefix (BFPFX). BFPFX is a prefix on all fixed-length VTAM buffers. It precedes the data portion of the buffer and identifies the buffer pool to which a buffer belongs. The BFPFX is X'3C' in length. The third word of BFPFX points to the PCB. To find other buffers, add to or subtract from the address of the current prefix the length of the buffer (contained in the PCB + X'22'(PCBBUFSZ)) plus 60 (hex 3C) for the prefix. If the new location does not contain a BFPFX pointing to the same PCB, you have reached the end of the pool.

Use the count of buffers at PCB + X'24' (PCBTOTNO) to determine whether there are also expansion extents. Information about them is found in the PXB, which is chained from PCB + X'58' (PCBPXB). If that location is not equal to zero, and if the total buffer count at PCB + X'24' (PCBTOTNO) is greater than the number of buffers found in the initial extent, there are additional buffers pointed to by PXBs. Each PXB may point to a storage area. Examine location hex 0C (PXBSDADR) in the PXB. If that location is not equal to zero, it is the beginning address of the extent. The number of buffers is contained at PXB + X'16'(PXBTOTL). Examine the PXB chain pointer at hex 04 (PXBPXB) to find the next PXB, if any.

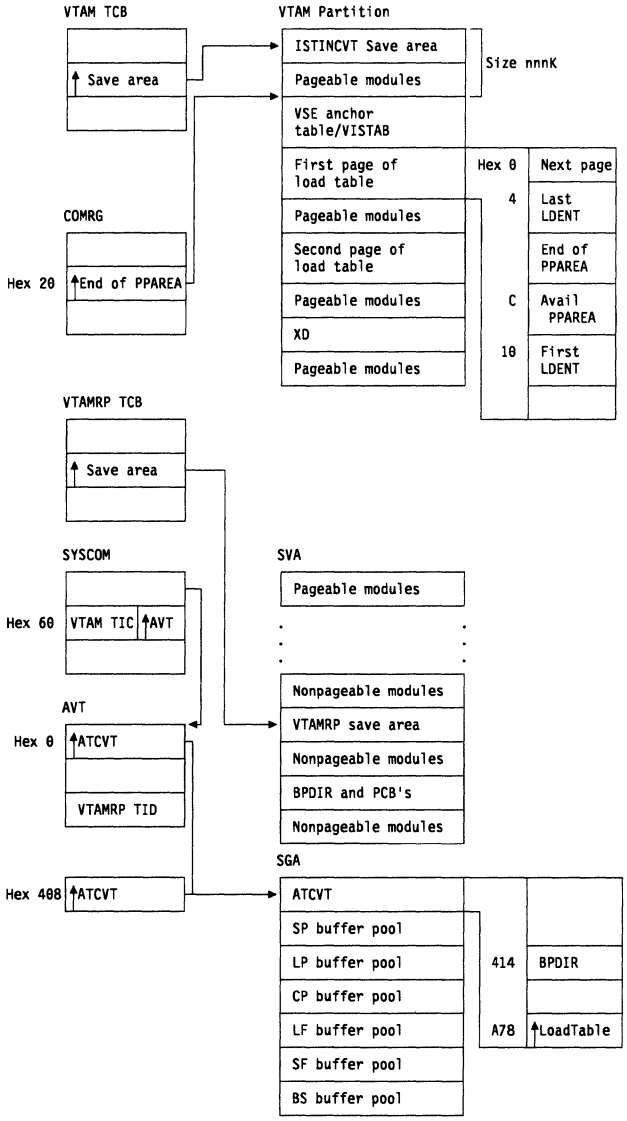


Figure 83. Map of the VTAM Partition VSE

Appendix C. First Failure Support Technology (FFST) Probes (MVS)(VM)

This appendix contains the following sections:

See "First Failure Support Technology (FFST) for VTAM (MVS) (VM)" on page 217 for additional information.

FFST Probe Index

The following table provides an index of FFST probes in alphanumerical order by probe name:

Table 36. FFST Probe Index

Probe Name	Component	See
ISTCOCxx	Control Operator	"Control Operator Probes" on page 848
ISTCSCxx	Configuration Services	"Configuration Services Probes" on page 847
ISTDLCxx	Dependent LU Server	"Dependent LU Server Probes" on page 851
ISTDRCxx	Directory Services	"Directory Services Probes" on page 852
ISTFSCxx MVS	Coupling Facility Services	"Coupling Facility Services Probes (MVS)" on page 849
ISTIPCxx MVS	IP Network Access	"IP Network Access Probes (MVS)" on page 859
ISTMTCxx	Management Services Transport	"Management Services Transport Probes" on page 860
ISTNACxx	Network Resource Management	"Network Resource Management Probes" on page 861
ISTNSCxx	LU Network Services	"LU Network Services Probes" on page 860
ISTORCxx	Storage Management	"Storage Management Probes" on page 869
ISTRACxx	Trace Services	"Trace Services Probes" on page 870
ISTSCCxx	Session Services CP-CP	"Session Services CP-CP Probes" on page 861
ISTSLCxx	Session Services LU-LU	"Session Services LU-LU Probes" on page 863
ISTSSCxx	Session Services	"Session Services Probes" on page 861
ISTTRCxx	Topology and Routing Services	"Topology and Routing Services Probes" on page 869
ISTTSCxx	Transmission Subsystem	"Transmission Subsystem Probes" on page 870

FFST Probe Information

When a VTAM FFST probe is triggered, an unexpected condition has occurred in the network. The process that received the condition might not complete normally. The VTAM program will attempt to recover from the unexpected condition and will continue processing subsequent requests. Recovery might not be possible for some system conditions and subsequent requests might fail, terminals might hang, and other abnormal conditions might occur.

Dump data is collected to assist in identifying the source of the problem. The processing element is freed and processing continues. If the probe triggers multiple times you might need to halt and restart VTAM.

Contact the appropriate IBM Support Center and supply the service representative with the console listing that is written at the time of the error and the dump data produced by the probe.

For information on using FFST dumps, see "First Failure Support Technology (FFST) for VTAM (MVS) (VM)" on page 217.

FFST Probe Naming Convention

The following table describes the naming convention for the FFST probe name. **ISTRAC01** is used as an example.

Characters	Example	Description
1,2,3	IST	These characters represent the product identifier. For VTAM probes, these characters are IST.
4, 5	RA	These characters represent the VTAM component identifier. In this example, RA is the component identifier for the Trace Services component.
6	C	For VTAM probes, this character is usually C.
7, 8	01	These characters represent the probe identification number. This number is not duplicated.

FFST Probe Descriptions

This section includes a table for each component that contains FFST probe instructions. The components are in alphabetical order, and the probes for each component are in alphanumerical order by probe name. "FFST Probe Index" on page 845 provides an index of FFST probes in alphanumerical order by probe name. Each table in this section shows the probe name, the module that issues it, and whether the probe creates a full or minidump when triggered.

"FFST Probe Index" on page 845 provides an index of FFST probes in alphanumerical order by probe name.

For information on using FFST dumps, see "First Failure Support Technology (FFST) for VTAM (MVS) (VM)" on page 217.

Configuration Services Probes

Table 37 (Page 1 of 2). FFST Probes for Configuration Services (ISTCSCxx)

Probe Name	Module	Description	Dump Type
ISTCSC01	ISTACCQ3	This probe detects request contact (REQCONT) errors caused by an unrecognized CPNAME. It produces information about the failing request unit processing element (RUPE), request/response unit (RU), and variable work area (VWA). If available, the resource definition table entry (RDTE) for the LINE and PU is also reported.	Mini
ISTCSC02	ISTCSCRC	Resource registration failure was detected.	Mini
ISTCSC03	ISTCSCRE	Resource registration failure was detected.	Mini
ISTCSC04	ISTCSCRF	Resource registration failure was detected.	Mini
ISTCSC05	ISTCSCRJ	Resource registration failure was detected.	Mini
ISTCSC06	ISTCSCRK	Resource registration failure was detected.	Mini
ISTCSC07	ISTCSCRH	Resource registration failure was detected.	Mini
ISTCSC08	ISTACCDF	Memory is freed inside an NCP causing a pointer that is not valid. Information produced by this probe contains the following: • The address of the resource definition table entry (RDTE) for the NCP • The unrecognized resource definition intermediate node (RIN) pointer	Full
ISTCSC09	ISTDECQE	The resource definition table entry (RDTE) for the dependent LU requester CDRSC was not found.	Full
ISTCSC10	ISTDECQD	The resource definition table entry (RDTE) for the dependent LU requester CDRSC was not found.	Full
ISTCSC11	ISTDECP3	The resource definition table entry (RDTE) for the dependent LU requester CDRSC was not found.	Full
ISTCSC12	ISTDECP4	The resource definition table entry (RDTE) for the dependent LU requester CDRSC was not found.	Full
ISTCSC13	ISTDECQ4	A protocol violation was detected.	Full

Table 37 (Page 2 of 2). FFST Probes for Configuration Services (ISTCSCxx)

Probe Name	Module	Description	Dump Type
ISTCSC14	ISTDECQ4	A topology database update (TDU) error was detected.	Full
ISTCSC15	ISTACCQE	An attempt to add the resource definition table entry (RDTE) for the DLUS-supported LU failed because the network address is already in use or was not properly freed by a previous user. Related pointer values are provided in the secondary symptom string.	Full
ISTCSC16	ISTDECLL	The GETRDTE for the higher-level resource failed.	Full

Control Operator Probes

Table 38 (Page 1 of 2). FFST Probes for Control Operator (ISTCOCxx)

Probe Name	Module	Description	Dump Type
ISTCOC01	ISTCOCDL	A suspended log mode is not found. Check the mode the application program specifies on the APPCCMD CONTROL=OPRCNTRL, QUALIFY=RESTORE macroinstruction. You can use the following VTAM dump functions to display advanced program-to-program communication (APPC)-related information: • APPLCONV • APPLMODE • APPMODAL • PARTNRLU. For information about these functions, see Chapter 7, "Using VTAM Dump Analysis Tools (MVS)" on page 237.	Full
ISTCOC02	ISTCOCSC	A suspended log mode is not found. Check the mode the application program specifies on the APPCCMD CONTROL=OPRCNTRL, QUALIFY=RESTORE macroinstruction. You can use the following VTAM dump functions to display advanced program-to-program communication (APPC)-related information: • APPLCONV • APPLMODE • APPMODAL • PARTNRLU. For information about these functions, see Chapter 7, "Using VTAM Dump Analysis Tools (MVS)" on page 237.	Full

Table 38 (Page 2 of 2). FFST Probes for Control Operator (ISTCOCxx)

Probe Name	Module	Description	Dump Type
ISTCOC03	ISTCOCVR	A suspended log mode is not found. Check the mode the application program specifies on the APPCCMD CONTROL=OPRCNTRL, QUALIFY=RESTORE macroinstruction. You can use the following VTAM dump functions to display advanced program-to-program communication (APPC)-related information: • APPLCONV • APPLMODE • APPMODAL • PARTNRLU. For information about these functions, see Chapter 7, "Using VTAM Dump Analysis Tools (MVS)" on page 237.	Full

Coupling Facility Services Probes (MVS)

Table 39 (Page 1 of 3). FFST Probes for Coupling Facility Services (ISTFSCxx)

Probe Name	Module	Description	Dump Type
ISTFSC00	ISTFSUMT	An unexpected return code *xxxx* and reason code *yyyy* were returned for IXLLIST when VTAM attempted to move an entry to the end of a list.	Mini
ISTFSC01	ISTFSNRT	A work element that was not recognized by coupling facility services (CFS) was queued to the CFS connection services PAB.	Mini
ISTFSC02	ISTFSNRT	A work element was recognized by coupling facility services (CFS), but it contained incorrect information.	Mini
ISTFSC03	ISTFSNCF	An unexpected input was received for the current state of the connection finite state machine.	Mini
ISTFSC04	ISTFSUDC	An unexpected return code *xxxx* and reason code *yyyy* were returned for the IXLDISC macro.	Mini
ISTFSC05	ISTFSURB	An unexpected return code *xxxx* and reason code *yyyy* were returned for the IXLREBLD macro.	Mini
ISTFSC06	ISTFSURS	An unexpected return code *xxxx* and reason code *yyyy* were returned for the IXLEERSP macro.	Mini
ISTFSC07	ISTFSUUS	An unexpected return code *xxxx* and reason code *yyyy* were returned for the IXLUSYNC macro.	Mini
ISTFSC08	ISTFSUFC	An unexpected return code *xxxx* and reason code *yyyy* were returned for the IXLFORCE macro.	Mini

Table 39 (Page 2 of 3). FFST Probes for Coupling Facility Services (ISTFSCxx)

Probe Name	Module	Description	Dump Type
ISTFSC09	ISTFSNCN	An unexpected return code *xxxx* and reason code *yyyy* were returned for the IXLCONN macro.	Mini
ISTFSC0A	ISTFSNCN	VTAM was connected to an unexpected structure type.	Mini
ISTFSC0B	ISTFSNCN	The connection name returned in the IXLCONN answer area was not the name specified on IXLCONN.	Mini
ISTFSC0C	ISTFSGLR	VTAM was not able to find a local generic mapping while rebuilding the coupling facility structure with local data.	Mini
ISTFSC0D	ISTFSGRT	An unexpected input was received for the current state of the generic resource structure finite state machine.	Mini
ISTFSC0E	ISTFSGG1	VTAM made a counting error associated with a generic resource.	Mini
ISTFSC11	ISTFSGRT	VTAM has disconnected from the coupling facility structure ISTGENERIC because a timed process did not complete within the given time.	Full
ISTFSC1A	ISTFSURL	An unexpected return code *xxxx* and reason code *yyyy* were returned for IXLLIST when VTAM attempted to read a list entry.	Mini
ISTFSC1B	ISTFSUUL	An unexpected return code *xxxx* and reason code *yyyy* were returned for IXLLIST when VTAM attempted to update a list entry.	Mini
ISTFSC1C	ISTFSUCL	An unexpected return code *xxxx* and reason code *yyyy* were returned for IXLLIST when VTAM attempted to create a list entry.	Mini
ISTFSC1D	ISTFSUHT	An unexpected return code *xxxx* and reason code *yyyy* were returned for IXLLIST when VTAM attempted to read and move a list entry.	Mini
ISTFSC1E	ISTFSKRT	An unexpected return code *xxxx* and reason code *yyyy* were returned for IXLLIST when VTAM attempted to set a lock entry.	Mini
ISTFSC1F	ISTFSKRT	An unexpected return code *xxxx* and reason code *yyyy* were returned for IXLLIST when VTAM attempted to release a lock entry.	Mini
ISTFSC20	ISTFSKRT	An unexpected return code *xxxx* and reason code *yyyy* were returned for IXLLIST when VTAM attempted to release a lock entry for a failed connector.	Mini

Table 39 (Page 3 of 3). FFST Probes for Coupling Facility Services (ISTFSCxx)

Probe Name	Module	Description	Dump Type
ISTFSC21	ISTFSLFG	VTAM found a name defined as both a generic resource name and a USERVAR.	Mini
ISTFSC22	ISTFSUBF	A bad pointer to an IXLLIST buffer list was detected by this module.	Mini
ISTFSC23	ISTFSUDL	An unexpected return code *xxxx* and reason code *yyyy* were returned for IXLLIST when VTAM attempted to delete a list entry.	Mini
ISTFSC24	ISTFSUEN	An unexpected return code *xxxx* and reason code *yyyy* were returned for IXLLIST when VTAM attempted to read list control information.	Mini
ISTFSC25	ISTFSURH	An unexpected return code *xxxx* and reason code *yyyy* were returned for IXLLIST when VTAM attempted to read a list entry from the head of a list.	Mini
ISTFSC26	ISTFSUWT	An unexpected return code *xxxx* and reason code *yyyy* were returned for IXLLIST when VTAM attempted to write a list entry to the end of a list.	Mini

Dependent LU Server Probes

Table 40. FFST Probes for Dependent LU Server (ISTDLCxx)

Probe Name	Module	Description	Dump Type
ISTDLC01	ISTDLCRD	Element representing the dependent LU requester could not be found	Mini
ISTDLC02	ISTDLCRD	Element representing the PU could not be found.	Mini
ISTDLC03	ISTDLCDP	Element representing the dependent LU requester could not be found.	Mini
ISTDLC04	ISTDLCDR	Element representing the dependent LU requester could not be found.	Mini
ISTDLC33	ISTDLCED	Element representing the dependent LU requester could not be found.	Mini
ISTDLC34	ISTDLCED	Element representing the PU could not be found.	Mini
ISTDLC35	ISTDLCED	Element representing DLUS-supported LU could not be found.	Mini
ISTDLC36	ISTDLCRS	Element representing the dependent LU requester could not be found.	Mini
ISTDLC37	ISTDLCRS	Not enough storage to build a REQDACTPU response to be sent to the dependent LU requester.	Mini

Directory Services Probes

Table 41 (Page 1 of 7). FFST Probes for Directory Services (ISTDRCxx)

Probe Name	Module	Description	Dump Type
ISTDRC00	ISTDRCRT	An unrecognized request element was received.	Full
ISTDRC01	ISTDRCRT	An unrecognized interprocess signal was received.	Full
ISTDRC02	ISTDROBO	An unrecognized function code was received.	Full
ISTDRC04	ISTDROFO	A build of a found CV80 is required but is not present.	Full
ISTDRC05	ISTDROFO	A build of a found CV3C is required but is not present.	Full
ISTDRC06	ISTDROFO	A build of a found CV3C is required but is not present.	Full
ISTDRC07	ISTDROFO	A build of a found CV3D is required but is not present.	Full
ISTDRC08	ISTDROFO	A build of a found CV3E is required but is not present.	Full
ISTDRC09	ISTDROFO	A build of a found CV40 is required but is not present.	Full
ISTDRC0A	ISTDROLM	A build of a locate GDS variable is required but is not present.	Full
ISTDRC0B	ISTDROLM	A build of a locate GDS variable is required but is not present.	Full
ISTDRC0C	ISTDROLM	A build of a found GDS variable is required but is not present.	Full
ISTDRC0D	ISTDROLM	A build of a register GDS variable is required but is not present.	Full
ISTDRC0E	ISTDROLM	A build of a CDINIT GDS variable is required but is not present.	Full
ISTDRC0F	ISTDROLM	A build of an IOCD GDS variable is required but is not present.	Full
ISTDRC10	ISTDRORS	A build of a resource name failed because the requested name was not recognized.	Full
ISTDRC11	ISTDRO0E	A build of a CV0E failed because the requested name was not recognized.	Full
ISTDRC12	ISTDRO60	A build of a CV60 failed because the requested name was not recognized.	Full
ISTDRC13	ISTDROCG	The search type that was requested was not recognized.	Full
ISTDRC14	ISTDROLM	A build of a notify GDS variable is required but is not present.	Full
ISTDRC15	ISTDRDUE	An unknown entry type was received.	Full
ISTDRC16	ISTDRDDD	Either an unknown resource or no resource was found for a resource delete request.	Full
ISTDRC17	ISTDRDDD	The associated network ID information was not found for a resource delete request.	Full
ISTDRC18	ISTDRDDD	A release of storage associated with a resource failed to complete.	Full
ISTDRC19	ISTDRDDD	A release of storage associated with a resource failed to complete.	Full

Table 41 (Page 2 of 7). FFST Probes for Directory Services (ISTDRCxx)

Probe Name	Module	Description	Dump Type
ISTDRC1A	ISTDRDDD	A release of storage associated with a resource failed to complete.	Full
ISTDRC1B	ISTDRDUD	An update directory request was received, but no list elements were found.	Full
ISTDRC1C	ISTDRDCD	A release of storage associated with a resource failed to complete.	Full
ISTDRC1D	ISTDRDCD	A release of storage associated with a CACHE_DATA interprocess signal failed to complete.	Full
ISTDRC1E	ISTDRDCD	A CACHE_DATA interprocess signal failed because the requested name was not recognized.	Full
ISTDRC1F	ISTDRAPC	A CP_STATUS interprocess signal failed because the requested name was not recognized.	Full
ISTDRC20	ISTDRAUA	Cannot remove an adjacent control point from the list.	Full
ISTDRC21	ISTDRAUA	Cannot add an adjacent control point to the list.	Full
ISTDRC22	ISTDRAUA	ISTDRAUA received an action code indicating that an unexpected condition has occurred.	Full
ISTDRC23	ISTDRAUA	ISTDRAUA was unable to process an unrecognized action code.	Full
ISTDRC24	ISTDREDS	An unexpected interprocess signal was received.	Full
ISTDRC25	ISTDRFCH	An unrecognized interprocess signal was received for a node role.	Full
ISTDRC26	ISTDRFLE	The sender of a request cannot be identified.	Full
ISTDRC27	ISTDRFNV	An unknown verify indicator value was received.	Full
ISTDRC28	ISTDRFSI	An unrecognized CV35 vector was received from a VTAM component.	Full
ISTDRC29	ISTDRF35	VTAM detected a software error while processing a CV35 vector.	Full
ISTDRC2A	ISTDRFLE	A protocol error was detected while processing an interprocess signal from a VTAM component.	Full
ISTDRC2B	ISTDRFNV	An unrecognized name was received from a VTAM component.	Full
ISTDRC2C	ISTDRFSI	A CV35 vector was received while the keep indicator was set.	Full
ISTDRC2D	ISTDRFPQ	An unrecognized network-qualified name was received.	Full
ISTDRC2E	ISTDRFPY	An unrecognized network-qualified name was received.	Full
ISTDRC2F	ISTDROLO	A build of a Locate CV80 is required, but none is present.	Full
ISTDRC30	ISTDROLO	A build of a Locate CV60 is required, but none is present.	Full
ISTDRC31	ISTDROLO	A build of a Locate CV81 is required, but none is present.	Full
ISTDRC32	ISTDROLO	A build of a Locate CV2B is required, but none is present.	Full

Table 41 (Page 3 of 7). FFST Probes for Directory Services (ISTDRCxx)

Probe Name	Module	Description	Dump Type
ISTDRC33	ISTDROLO	A build of a Locate CV0E is required, but none is present.	Full
ISTDRC34	ISTDROLO	A build of a Locate CV35 is required, but none is present.	Full
ISTDRC35	ISTDROFI	A build of a Find CV80 is required, but none is present.	Full
ISTDRC36	ISTDROFI	A build of a Find CV3C is required, but none is present.	Full
ISTDRC37	ISTDROFI	A build of a Find CV3C is required, but none is present.	Full
ISTDRC38	ISTDROFI	A build of a Find CV3D is required, but none is present.	Full
ISTDRC39	ISTDROFI	A build of a Find CV3E is required, but none is present.	Full
ISTDRC3A	ISTDROFI	A build of a Find CV81 is required, but none is present.	Full
ISTDRC3B	ISTDROFI	A build of a Find CV81 is required, but none is present.	Full
ISTDRC3C	ISTDROFI	A build of a Find CV82 is required, but none is present.	Full
ISTDRC3D	ISTDRMCH	An unrecognized input was detected.	Full
ISTDRC3E	ISTDRMCS	An unrecognized interprocess signal was received.	Full
ISTDRC3F	ISTDRMCS	An unrecognized state was detected.	Full
ISTDRC40	ISTDRMDO	An unrecognized action was requested.	Full
ISTDRC41	ISTDRMDO	An unexpected interprocess signal was received.	Full
ISTDRC42	ISTDRMDO	Unexpected elements were found in a locate search.	Full
ISTDRC43	ISTDRMDO	An unexpected interprocess signal was received.	Full
ISTDRC44	ISTDRMDO	An unrecognized state was detected.	Full
ISTDRC45	ISTDRMDO	An unexpected return code was received.	Full
ISTDRC46	ISTDRMDO	A list requiring at least one element was empty.	Full
ISTDRC47	ISTDRMFN	An unexpected interprocess signal was received.	Full
ISTDRC48	ISTDRMFN	An unrecognized state was detected.	Full
ISTDRC49	ISTDRMNV	An unexpected return code was received.	Full
ISTDRC4A	ISTDRMNV	An unexpected return code was received.	Full
ISTDRC4B	ISTDRMNV	An unexpected interprocess signal was received.	Full
ISTDRC4C	ISTDRMSA	An unexpected interprocess signal was received.	Full
ISTDRC4D	ISTDRMSA	An unrecognized state was detected.	Full
ISTDRC4E	ISTDRMTD	An unexpected interprocess signal was received.	Full
ISTDRC4F	ISTDRMTD	An unrecognized state was detected.	Full
ISTDRC50	ISTDRRSD	Unrecognized inputs for state were detected.	Full
ISTDRC51	ISTDRRSR	Unrecognized inputs for state were detected.	Full

Table 41 (Page 4 of 7). FFST Probes for Directory Services (ISTDRCxx)

Probe Name	Module	Description	Dump Type
ISTDRC52	ISTDRRTP	A work-element was received when none was expected.	Full
ISTDRC53	ISTDRSBR	ISTDRSBR was unable to add a locate control block to the list.	Full
ISTDRC54	ISTDRSNB	An unrecognized state was detected.	Full
ISTDRC55	ISTDRSCH	An unexpected interprocess signal was received.	Full
ISTDRC56	ISTDRSDB	An unexpected interprocess signal was received.	Full
ISTDRC57	ISTDRSDB	An unrecognized state was detected.	Full
ISTDRC58	ISTDRSDS	An unexpected return code was received.	Full
ISTDRC59	ISTDRSDS	An unexpected return code was received.	Full
ISTDRC5A	ISTDRSDS	An unexpected return code was received.	Full
ISTDRC5B	ISTDRSDS	An unexpected return code was received.	Full
ISTDRC5C	ISTDRSDS	An unexpected return code was received.	Full
ISTDRC5D	ISTDRSDS	An unexpected interprocess signal was received.	Full
ISTDRC5E	ISTDRSDS	An unrecognized state was detected.	Full
ISTDRC5F	ISTDRSFP	An unrecognized combination of node roles was detected.	Full
ISTDRC60	ISTDRSIC	An unrecognized state was received.	Full
ISTDRC61	ISTDRSIC	An unrecognized status was received.	Full
ISTDRC62	ISTDRSIN	An unrecognized search status for a called task was received.	Full
ISTDRC63	ISTDRSMD	An unrecognized function parameter was received.	Full
ISTDRC64	ISTDRSNV	An unrecognized function was specified.	Full
ISTDRC65	ISTDRSNV	An unexpected interprocess signal was received.	Full
ISTDRC66	ISTDRSOH	An unrecognized function code was received.	Full
ISTDRC67	ISTDRSOH	An unrecognized combination of inputs was received.	Full
ISTDRC68	ISTDRSOH	An unrecognized combination of inputs was received.	Full
ISTDRC69	ISTDRSOH	An unexpected interprocess signal was received.	Full
ISTDRC6A	ISTDRSOH	An unrecognized state was detected.	Full
ISTDRC6B	ISTDRSQS	An unrecognized function was detected.	Full
ISTDRC6C	ISTDRSSQ	An unrecognized search task was requested.	Full
ISTDRC6D	ISTDRSYB	VTAM was unable to remove a locate control block from the list.	Full
ISTDRC6E	ISTDRYCD	An unrecognized query function was detected.	Full
ISTDRC6F	ISTDRSCA	An unrecognized entry type was specified.	Full
ISTDRC70	ISTDRDDD	A release of storage associated with a resource failed to complete.	Full
ISTDRC71	ISTDROLO	A build of a Locate CV82 is required, but none is present.	Full

Table 41 (Page 5 of 7). FFST Probes for Directory Services (ISTDRCxx)

Probe Name	Module	Description	Dump Type
ISTDRC72	ISTDROFI	A build of a Find CV40 is required, but none is present.	Full
ISTDRC73	ISTDRMAC	The input received is not recognized for the current state.	Full
ISTDRC74	ISTDRMAC	The input received is not recognized for the current state.	Full
ISTDRC75	ISTDRMAC	An unrecognized state was received.	Full
ISTDRC76	ISTDRMAC	An unexpected return code was received.	Full
ISTDRC77	ISTDRRCF	An unrecognized state was received.	Full
ISTDRC78	ISTDRRCP	A register GDS variable is required, but is not present.	Full
ISTDRC79	ISTDRRCR	An unexpected return code was received.	Full
ISTDRC7A	ISTDRRNE	An unexpected return code was received.	Full
ISTDRC7B	ISTDRRNF	An unrecognized state was received.	Full
ISTDRC7C	ISTDRRNF	The input received is not recognized for the current state.	Full
ISTDRC7D	ISTDRRNF	The input received is not recognized for the current state.	Full
ISTDRC7E	ISTDRRNF	The input received is not recognized for the current state.	Full
ISTDRC7F	ISTDRRNF	The input received is not recognized for the current state.	Full
ISTDRC80	ISTDRRNF	The input received is not recognized for the current state.	Full
ISTDRC81	ISTDRRNF	The input received is not recognized for the current state.	Full
ISTDRC82	ISTDRRNF	The input received is not recognized for the current state.	Full
ISTDRC83	ISTDRRNF	The input received is not recognized for the current state.	Full
ISTDRC84	ISTDRRNP	A locate control block is required but is not present.	Full
ISTDRC85	ISTDRRNP	An unrecognized state was detected.	Full
ISTDRC86	ISTDRRNR	An unexpected return code was received.	Full
ISTDRC87	ISTDRRNR	The available length is not sufficient for a required GDS variable.	Full
ISTDRC88	ISTDRRNR	The available length is not sufficient for a required GDS variable.	Full
ISTDRC89	ISTDRRNR	An unexpected return code was received.	Full
ISTDRC8A	ISTDRRNR	The available length is not sufficient for a required GDS variable.	Full
ISTDRC8B	ISTDRSBC	A parsed locate element is required, but is not present.	Full
ISTDRC8C	ISTDRSNB	An invalid state was detected upon entry to ISTDRSNB.	Full
ISTDRC8D	ISTDRSSD	An unknown task was loaded.	Full

Table 41 (Page 6 of 7). FFST Probes for Directory Services (ISTDRCxx)

Probe Name	Module	Description	Dump Type
ISTDRC8E	ISTDRFDS	A directory server notify was received with an empty list.	Full
ISTDRC8F	ISTDRFDS	An unexpected return code was received.	Full
ISTDRC90	ISTDRSHR	An unexpected input operation was received.	Full
ISTDRC91	ISTDRSHR	An unrecognized max hierarchy was detected.	Full
ISTDRC92	ISTDRSHR	An unexpected resource CV combination was detected.	Full
ISTDRC93	ISTDRSHR	An unexpected max hierarchy combination was detected.	Full
ISTDRC94	ISTDRSNI	An unrecognized locate was detected.	Full
ISTDRC95	ISTDRSNI	An unrecognized node role was identified in an LCB.	Full
ISTDRC96	ISTDRDUP	A DBUPDATE was attempted at a VTAM end node.	Full
ISTDRC97	ISTDRDQP	A DBQUERY was attempted at a VTAM end node.	Full
ISTDRC98	ISTDRDDP	A DBDELETE was attempted at a VTAM end node.	Full
ISTDRC99	ISTDRERT	The DS PAB router received an unrecognized IPS.	Full
ISTDRC9A	ISTDROCG	Unrecognized parameter value was received.	Full
ISTDRCA7	ISTDRMBO	An unrecognized interprocess signal was received.	Full
ISTDRCA8	ISTDRMBO	An unrecognized interprocess signal was received.	Full
ISTDRCA9	ISTDRMBO	An unrecognized interprocess signal was received.	Full
ISTDRCAA	ISTDRMBO	An unrecognized state was detected.	Full
ISTDRCAB	ISTDREBN	An unrecognized state was detected.	Full
ISTDRCAC	ISTDRMBN	An unrecognized state was detected.	Full
ISTDRCAD	ISTDRMGS	An unrecognized interprocess signal was received.	Full
ISTDRCAE	ISTDRMGS	An unrecognized state was detected.	Full
ISTDRCAF	ISTDRFGS	An unrecognized control vector was detected; CV0E was expected but is not present.	Full
ISTDRCB0	ISTDRFGS	An unrecognized control vector was detected; CV35 was expected but is not present.	Full
ISTDRCB1	ISTDRFGU	An unrecognized control vector was detected; CV35 was expected but is not present.	Full
ISTDRCB2	ISTDROIE	A build of a CV0E failed due to an unrecognized name.	Full
ISTDRCB3	ISTDROIE	A build of a Find CV0E is required, but none is present.	Full
ISTDRCB4	ISTDRFMH	Incorrect hierarchy was passed to module.	Full
ISTDRCB5	ISTDRFBN	A unrecognized network-qualified name was detected.	Full
ISTDRCB6	ISTDRFBN	List operation failed.	Full
ISTDRCB7	ISTDRFBN	A unrecognized network-qualified name was detected.	Full
ISTDRCB8	ISTDRFBN	An attempt to free storage failed.	Full
ISTDRCB9	ISTDRSPC	List operation failed.	Full

Table 41 (Page 7 of 7). FFST Probes for Directory Services (ISTDRCxx)

Probe Name	Module	Description	Dump Type
ISTDRCBA	ISTDRSPC	List operation failed.	Full
ISTDRCBB	ISTDRSPC	List operation failed.	Full
ISTDRCBC	ISTDRSPC	List operation failed.	Full
ISTDRCBD	ISTDRSPC	Attempt to free storage failed.	Full
ISTDRCC1	ISTDRSDT	An unrecognized state was detected on entry.	Full

IP Network Access Probes (MVS)

Table 42 (Page 1 of 2). FFST Probes for IP Network Access (ISTIPCxx)

Probe Name	Module	Description	Dump Type
ISTIPC01	ISTIPCDI	Unexpected work element was received.	Mini
ISTIPC02	ISTIPCDN	Network-qualified name was not valid.	Mini
ISTIPC03	ISTIPCDR	Error occurred when VTAM was reading data into its buffers.	Full
ISTIPC04	ISTIPCDR	Error occurred when VTAM was reading data into its buffers.	Full
ISTIPC05	ISTIPCDW	Unexpected MPTN format was received.	Mini
ISTIPC06	ISTIPCDW	Unexpected work element was received.	Mini
ISTIPC07	ISTIPCIC	No outstanding work element was found.	Full
ISTIPC08	ISTIPCIR	Unexpected value was found in work element field.	Mini
ISTIPC09	ISTIPCIS	Unexpected interrupt type was detected. See Table 80 on page 1165 to determine what book describes IUCV external interrupts.	Mini
ISTIPC10	ISTIPCOR	Unexpected work element was received.	Mini
ISTIPC11	ISTIPCRB	Unexpected finite state machine state was detected when VTAM was reading data into its buffers.	Mini
ISTIPC12	ISTIPCRB	Error occurred when VTAM was reading data into its buffers.	Full
ISTIPC13	ISTIPCRB	Unexpected MPTN format was received.	Mini
ISTIPC14	ISTIPCRB	Error occurred when VTAM was reading data into its buffers.	Full
ISTIPC15	ISTIPCRB	Mismatched transmission subsystem control block (TSCB) count was detected when VTAM was reading data into its buffers.	Full
ISTIPC16	ISTIPCRC	Unexpected finite state machine state was detected when VTAM was reading data into its buffers.	Mini
ISTIPC17	ISTIPCSE	Internal control block was not found.	Mini
ISTIPC18	ISTIPCSG	Unexpected work element.	Mini
ISTIPC19	ISTIPCSG	Unexpected value was found in work element field.	Mini
ISTIPC20	ISTIPCSG	Unexpected finite state machine state was detected when VTAM was reading data into its buffers.	Mini
ISTIPC21	ISTIPCWB	Unexpected finite state machine state was detected when VTAM was reading data into its buffers.	Mini
ISTIPC22	ISTIPCWB	MPTN format length mismatch was detected.	Full
ISTIPC23	ISTIPCWR	Unexpected work element was received.	Mini
ISTIPC24	ISTIPCWR	Unexpected work element was received.	Mini
ISTIPC25	ISTIPCWR	Unexpected work element was received.	Mini
ISTIPC26	ISTIPCDW	Unexpected work element was received.	Mini

Table 42 (Page 2 of 2). FFST Probes for IP Network Access (ISTIPCxx)

Probe Name	Module	Description	Dump Type
ISTIPC27	ISTIPCDR	MPTN datagram length is not valid. The FFST probe is tripped when a MPTN datagram received from TCP/IP requires more than twenty data buffers to receive it. A datagram of this size is excessively large and indicates a possible TCP/IP error.	Full

LU Network Services Probes

Table 43. FFST Probes for LU Network Services (ISTNSCxx)

Probe Name	Module	Description	Dump Type
ISTNSC01	ISTNSCRS	A suspended log mode is not found. Check the mode the application program specifies on the APPCCMD CONTROL=OPRCNTRL, QUALIFY=RESTORE macroinstruction. You can use the following VTAM dump functions to display advanced program-to-program communication (APPC)-related information: • APPLCONV • APPLMODE • APPMODAL • PARTNRLU.	Full

Management Services Transport Probes

Table 44 (Page 1 of 2). FFST Probes for Management Services Transport (ISTMTCxx)

Probe Name	Module	Description	Dump Type
ISTMTC01	ISTMTCFF	The main router received an unrecognized request-unit processing element.	Mini
ISTMTC02	ISTMTCFF	The main router received an unrecognized work element.	Full
ISTMTC03	ISTMTCFF	The START_TP_REPLY signal that is received contains a return code that is incorrect or unrecognized for this leg.	Mini
ISTMTC04	ISTMTCFF	An unexpected condition is detected when a query request is sent to the NetView application program's programmable peripheral interface (PPI) queue.	Mini
ISTMTC05	ISTMTCFF	An unexpected condition is detected while defining the VTAM queue to the NetView application program's programmable peripheral interface (PPI).	Mini
ISTMTC06	ISTMTCFF	An unexpected condition is detected while sending data from VTAM to the NetView processor across the programmable peripheral interface (PPI).	Mini
ISTMTC07	ISTMTCFF	The main router received an unrecognized work element.	Mini
ISTMTC08	ISTMTCFF	Unrecognized data is received from the NetView application program across the programmable peripheral interface (PPI).	Mini

Table 44 (Page 2 of 2). FFST Probes for Management Services Transport (ISTMTCxx)

Probe Name	Module	Description	Dump Type
ISTMTC09	ISTMTCFF	Unrecognized data is received from the NetView application program.	Mini
ISTMTC10	ISTMTCFF	The PPI queue received from the NetView application program contains unrecognized data.	Mini
ISTMTC11	ISTMTCFF	VTAM could not create a process scheduling table for MI.	Mini
ISTMTC12	ISTMTCFF	The MDS-MU received from the NetView application program is unrecognized.	Mini
ISTMTC13	ISTMTCFF	An expected outstanding request unit is not found on the outstanding request unit list.	Mini
ISTMTC14	ISTMTCFF	An expected outstanding request unit is not found on the outstanding request unit list.	Mini
ISTMTC15	ISTMTCFF	An expected session list entry is not found, but an ATL entry is found.	Mini
ISTMTC16	ISTMTCFF	An expected session-list entry is not found, but an ATL entry is found.	Mini

Network Resource Management Probes

Table 45. FFST Probes for Network Resource Management (ISTNACxx)

Probe Name	Module	Description	Dump Type
ISTNAC01	ISTNACTT	VTAM cannot complete a normal CLOSE ACB and must force the CLOSE ACB.	Full

Session Services Probes

Table 46. FFST Probes for Session Services (ISTSSCxx)

Probe Name	Module	Description	Dump Type
ISTSSC01	ISTSSCTM	A request/response unit (RU) is rejected by one of the session services parsing modules. This probe provides information about a condition that will cause hung terminals and logon problems.	Mini
ISTSSC02	ISTSSCXM	Address conflict was detected during merge.	Mini
ISTSSC03	ISTSSCXV	An SRT was missing for an autologon session. As a result, session setup failed.	Full
ISTSSCZ1	ISTSSCYC	An attempt was made to dequeue a SIB from a PU's SIB queue, but the SIB was not on the queue.	Full

Session Services CP-CP Probes

Table 47 (Page 1 of 2). FFST Probes for Session Services CP-CP (ISTSCCxx)

Probe Name	Module	Description	Dump Type
ISTSCC01	ISTSCCRT	A request processing element is received which contained an unrecognized request unit processing element (RUPE).	Mini

Table 47 (Page 2 of 2). FFST Probes for Session Services CP-CP (ISTSCCxx)

Probe Name	Module	Description	Dump Type
ISTSCC02	ISTSCRCA	SSC attempted to add an adjacent node control block to the adjacent node control block list, but the add failed to complete normally.	Mini
ISTSCC03	ISTSCCFM	The SSC coordinator finite state machine processor received null input in an unexpected state when one or more of the CP-CP sessions is in active or pending active status.	Mini
ISTSCC04	ISTSCUAR	An internal SSC module passed an unrecognized session status to ISTSCUAR.	Mini
ISTSCC05	ISTSCUAR	An internal SSC module passed an unrecognized session type to ISTSCUAR.	Mini
ISTSCC06	ISTSCUBS	An internal SSC module passed an unrecognized session type to ISTSCUBS.	Mini
ISTSCC07	ISTSCUBS	An internal SSC module passed an unrecognized session status to ISTSCUBS.	Mini
ISTSCC08	ISTSCUDS	An internal SSC module passed an unrecognized session type to ISTSCUDS.	Mini
ISTSCC09	ISTSCUNO	SSC entered an unexpected state on a contention-winner session.	Mini
ISTSCC10	ISTSCUNO	SSC entered an unexpected state on a contention-loser session.	Mini
ISTSCC11	ISTSCUNO	An internal SSC module passed an unrecognized session type to ISTSCUNO.	Mini
ISTSCC15	ISTSCCFM	The SSC coordinator finite state machine processor received non-null input in an unexpected state when one or more of the CP-CP sessions is in active or pending active status.	Mini
ISTSCC16	ISTSCTRQ	The REQUEST_CP_CAPABILITIES_TP received as input a signal that is neither an ACT_CP_CP_SESSION request nor a CONTINUE_CW response.	Mini
ISTSCC17	ISTSCRAY	An adjacent node has changed either from network node to end node or from end node to network node.	Full
ISTSCC19	ISTSCTRC	The SEND_REJECT_CONVGRP_TP received an input signal other than a DEACTIVATE_CP_CP_SESSION request.	Mini
ISTSCC20	ISTSCCFM	The SSC coordinator finite state machine processor encountered a should-not-occur condition and no CP-CP session cleanup is required.	Mini
ISTSCC21	ISTSCCRT	An adjacent node control block is not found for the received signal.	Mini
ISTSCC23	ISTSCCRT	A processing element is queued to session services CP-CP (SSC) but is not recognized to be either a request unit processing element (RUPE) or a timer queue element.	Mini

Session Services LU-LU Probes

The following table contains the LU-LU session services (SSL) software probes. When an SSL software probe is triggered, an error has occurred while processing network session services for an LU-LU connection. The search, session initiation, or request causing the error is rejected. VTAM will attempt to recover from the error and continue processing subsequent requests. Recovery might not be possible for some types of errors and subsequent requests might fail, terminals might hang, and other types of errors might occur.

A system dump is taken to assist in identifying the source of the problem. A VABEND macroinstruction can also be processed if specified by the probe instruction. The processing element is freed and processing continues.

Table 48 (Page 1 of 7). FFST Probes for Session Services LU-LU (ISTSLCxx)

Probe Name	Module	Description	Dump Type
ISTSLC01	ISTSLCRT	An unknown interprocess signal was dispatched to the SSL component process anchor block (PAB).	Mini
ISTSLC02	ISTSLRDP	Finite state machine information is missing when a SEARCH_RPY signal is processed.	Mini
ISTSLC03	ISTSLRDP	Finite state machine information is missing when a CDINIT SEARCH_RPY signal is processed.	Mini
ISTSLC04	ISTSLRDP	Finite state machine information is missing when an IOCD SEARCH_RPY signal is processed.	Mini
ISTSLC05	ISTSLRDP	Finite state machine information is missing when a NOTIFY SEARCH_RPY signal is processed.	Mini
ISTSLC06	ISTSLRDP	Finite state machine information is missing when a SEARCH_RPY signal is processed and a storage failure occurs.	Mini
ISTSLC07	ISTSLRXS	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC08	ISTSLTPI	An unexpected combination of conditions was encountered during third-party initiated finite state machine processing.	Mini
ISTSLC09	ISTSLBXN	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC10	ISTSLURR	Route Selection control vector (RSCV) information is missing from SSL's session initiation control block when performing a build request for a RECOMPUTE_ROUTE signal.	Mini
ISTSLC11	ISTSLBXJ	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC12	ISTSLRXJ	Finite state machine information is missing when an INIT_OTHER_COMP signal is processed, and a storage failure occurred.	Mini
ISTSLC13	ISTSLRXJ	Finite state machine information is missing when an INIT_OTHER_COMP signal is processed.	Mini

Table 48 (Page 2 of 7). FFST Probes for Session Services LU-LU (ISTSLCxx)

Probe Name	Module	Description	Dump Type
ISTSLC14	ISTSLBXI	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC15	ISTSLUDS	An error occurred while the SSL component was removing a session initiation control block key from the list.	Full
ISTSLC16	ISTSLUDS	An error occurred while the SSL component was removing a session initiation control block from the list.	Mini
ISTSLC17	ISTSLUGT	The ACMDT's node type information for this node had an unknown value because it did not indicate whether this is a VTAM end node or a VTAM network node.	Mini
ISTSLC18	ISTSLUBX	A build request was received for an unrecognized interprocess signal.	Mini
ISTSLC19	ISTSLUBX	A build request was received for an unrecognized interprocess signal.	Mini
ISTSLC20	ISTSLUBX	The SSL component could not remove a fully qualified procedure correlation identifier (FQPCID) from the list.	Mini
ISTSLC21	ISTSLUBX	The SSL component could not remove a FQPCID_MODIFIER from the list.	Mini
ISTSLC22	ISTSLRXI	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC23	ISTSLVPE	An unrecognized signal was received from directory services.	Mini
ISTSLC24	ISTSLUSC	The SSL component was unable to determine the type of signal to send to session services.	Mini
ISTSLC25	ISTSLURX	An unrecognized interprocess signal was received from session services.	Mini
ISTSLC26	ISTSLURD	An unrecognized interprocess signal was received from directory services.	Mini
ISTSLC27	ISTSLSNO	The Route Selection control vector (RSCV) information is missing from the control vector list of the session initiation control block.	Mini
ISTSLC28	ISTSLSNO	The Route Selection control vector (RSCV) information is missing from the control vector list of the session initiation control block.	Mini
ISTSLC29	ISTSLSNO	An unrecognized session initiation status was received for the current system state.	Mini
ISTSLC30	ISTSLSNO	An unexpected combination of conditions was encountered during finite state machine processing initiated by the secondary logical unit (SLU) at a network node (NN).	Mini
ISTSLC31	ISTSLUBD	A build request was received for an unrecognized interprocess signal.	Mini

Table 48 (Page 3 of 7). FFST Probes for Session Services LU-LU (ISTSLCxx)

Probe Name	Module	Description	Dump Type
ISTSLC32	ISTSLSED	An unexpected combination of conditions was encountered during finite state machine processing initiated by the secondary logical unit (SLU) at an end node (EN) destination logical unit (DLU).	Mini
ISTSLC33	ISTSLRXO	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC34	ISTSLRXQ	Finite state machine information is missing when a QUEUED signal is processed.	Mini
ISTSLC35	ISTSLRXC	Finite state machine information is missing when a SESS_REQ_COMP signal is processed.	Mini
ISTSLC36	ISTSLRXC	Finite state machine information is missing when a SESS_REQ_COMP signal is processed and storage problems occur.	Mini
ISTSLC37	ISTSLBXV	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC38	ISTSLBXS	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC39	ISTSLRXD	Finite state machine information is missing when a DEQUEUE signal is processed.	Mini
ISTSLC40	ISTSLPND	An unexpected combination of conditions was encountered during finite state machine processing initiated by the primary logical unit (PLU) at a network node.	Mini
ISTSLC41	ISTSLRXV	Finite state machine information is missing when a PROV_SC signal is processed.	Mini
ISTSLC42	ISTSLPNO	The Route Selection control vector (RSCV) information is missing from the control vector list of the session initiation control block.	Mini
ISTSLC43	ISTSLPNO	The RSCV information is missing from the control vector list of the session initiation control block.	Mini
ISTSLC44	ISTSLPNO	An unexpected combination of conditions was encountered during finite state machine processing initiated by the primary logical unit (PLU) at a network node.	Mini
ISTSLC45	ISTSLPNO	The RSCV information is missing from the control vector list of the session initiation control block.	Mini
ISTSLC46	ISTSLBXQ	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC47	ISTSLBXC	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC48	ISTSLRTT	Finite state machine information is missing when a RECOMPUTE_ROUTE_RPY signal is processed.	Mini
ISTSLC49	ISTSLRTR	Finite state machine information is missing when a REQ_ROUTE_RPY signal is processed.	Mini

Table 48 (Page 4 of 7). FFST Probes for Session Services LU-LU (ISTSLCxx)

Probe Name	Module	Description	Dump Type
ISTSLC50	ISTSLRDR	Finite state machine information is missing when a SEARCH_REQ signal is processed.	Mini
ISTSLC51	ISTSLRTC	Class of service (COS) and transmission priority field (TPF) control vector information is missing from the control vector list of the session initiation control block.	Mini
ISTSLC52	ISTSLRTC	Finite state machine information is missing when a REQ_COS_TPF_RPY signal is processed.	Mini
ISTSLC53	ISTSLRTV	Finite state machine information is missing when a REQ_TG_VECTORS_RPY signal is processed.	Mini
ISTSLC54	ISTSLRNT	The NOTIFY generalized data stream (GDS) variable being processed contained a NOTIFY_TYPE that was unknown or not valid.	Mini
ISTSLC55	ISTSLRNT	Notify control vector 80 (CV80) information is missing from the control vector list of the session initiation control block.	Mini
ISTSLC56	ISTSLRNT	Notify CV81 information is missing from the control vector list of the session initiation control block.	Mini
ISTSLC57	ISTSLRDN	The CDINIT GDS variable information is missing from a NON_VERIFY_REQ signal received from directory services.	Mini
ISTSLC58	ISTSLRDC	Finite state machine information is missing when a CHAIN_FLOW signal is processed.	Mini
ISTSLC59	ISTSLPXR	An unexpected combination of conditions was encountered during processing of the PLU-initiate exit border node pre-processor finite state machine (FSM).	Mini
ISTSLC60	ISTSLPXT	An unexpected combination of conditions was encountered during processing of the PLU-initiate exit border node post-processor finite state machine.	Mini
ISTSLC61	ISTSLPYR	An unexpected combination of conditions was encountered during processing of the PLU-initiate entry border node pre-processor finite state machine.	Mini
ISTSLC62	ISTSLPNS	RSCV information is missing from the control vector list of the session initiation control block.	Mini
ISTSLC63	ISTSLPNS	An unexpected combination of conditions was encountered during finite state machine processing initiated by the primary logical unit (PLU) at a network node.	Mini
ISTSLC64	ISTSLPED	An unexpected combination of conditions was encountered during finite state machine processing initiated by the primary logical unit (PLU) at an end node destination logical unit.	Mini
ISTSLC65	ISTSLPEO	An unexpected combination of conditions was encountered during finite state machine processing initiated by the primary logical unit (PLU) at an end node origination logical unit.	Mini
ISTSLC66	ISTSLBXT	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini

Table 48 (Page 5 of 7). FFST Probes for Session Services LU-LU (ISTSLCxx)

Probe Name	Module	Description	Dump Type
ISTSLC67	ISTSLBXR	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC68	ISTSLBXP	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC69	ISTSLBXO	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC70	ISTSLBXE	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC71	ISTSLBXD	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC72	ISTSLBDL	A failure occurred while performing post-processor information processing for a signal being sent to session services.	Mini
ISTSLC73	ISTSLSND	An unrecognized session initiation status was received for the current system state in SLU-initiate finite state machine processing.	Mini
ISTSLC74	ISTSLSND	An unexpected combination of conditions occurred during SLU-initiate finite state machine processing at a network node.	Mini
ISTSLC75	ISTSLSEO	An unexpected combination of conditions occurred during SLU-initiate finite state machine processing at an end node origination logical unit.	Mini
ISTSLC76	ISTSLUCS	The SSL component was unable to remove the key entry for the session initiation control block from the list.	Full
ISTSLC77	ISTSLRXN	Finite state machine information is missing when a PEND_SC signal is processed.	Mini
ISTSLC78	ISTSLUSL	An unrecognized LU role was provided for a session initiation control block find request.	Mini
ISTSLC79	ISTSLCRT	An unrecognized work element was dispatched to the SSL process anchor block (PAB).	Mini
ISTSLC80	ISTSLR63	The SSL component was unable to remove a CV63 from the control vector list of the session initiation control block.	Mini
ISTSLC81	ISTSLBLC	A CV82 control vector contains a network-qualified sender name that is not valid.	Mini
ISTSLC82	ISTSLBLC	A CV82 control vector contains a network-qualified host receiver name that is not valid.	Mini
ISTSLC83	ISTSLPNS	The precomputed RSCV that was to be destroyed was missing from the session initiation control block.	Mini

Table 48 (Page 6 of 7). FFST Probes for Session Services LU-LU (ISTSLCxx)

Probe Name	Module	Description	Dump Type
ISTSLC84	ISTSLUNV	The original class of service (COS) associated with the SLU-initiate request was not found in the parent session initiation control block during a non-verify search sent in response to a SLU-initiate request.	Mini
ISTSLC85	ISTSLRDI	The SSL component received a SESS_INIT_INFO_REQ signal which contained inconsistent data.	Full
ISTSLC86	ISTSLBXB	Failure occurred when performing front-end processing for a BN_SESS_RPY signal being sent to session services.	Mini
ISTSLC87	ISTSLBXH	Failure occurred when performing front-end processing for a CACHE_BN_INFO signal being sent to session services.	Mini
ISTSLC88	ISTSLBTM	The network-qualified adjacent non-native CP name is not valid.	Mini
ISTSLC89	ISTSLRXB	The network-qualified adjacent LU name is not valid.	Mini
ISTSLC90	ISTSLRXB	Failure occurred when performing front-end processing for a BN_SESS_RPY signal being sent to session services.	Mini
ISTSLC91	ISTSLRTM	Finite state machine information was missing when a PROCESS_MAP_COS signal from topology routing services (TRS) was being processed.	Mini
ISTSLC92	ISTSLUBX	The RSCV that should have been in the session initiation control block when building a CACHE_BN_INFO to session services was missing.	Mini
ISTSLC93	ISTSLRXB	The class of service (COS) was received on a BN_SESS_REQ, but the primary logical unit (PLU) COS from LOCATE time processing could not be found.	Mini
ISTSLC94	ISTSLPYT	An unexpected combination of conditions was encountered during processing of the PLU-initiate entry border node post-processor finite state machine.	Mini
ISTSLC95	ISTSLSXR	An unexpected combination of conditions was encountered during processing of the SLU-initiate exit border node pre-processor finite state machine.	Mini
ISTSLC96	ISTSLSXT	An unexpected combination of conditions was encountered during processing of the SLU-initiate exit border node post-processor finite state machine.	Mini
ISTSLC97	ISTSLSYR	An unexpected combination of conditions was encountered during processing of the SLU-initiate entry border node pre-processor finite state machine.	Mini
ISTSLC98	ISTSLSYT	An unexpected combination of conditions was encountered during processing of the SLU-initiate entry border node post-processor finite state machine.	Mini
ISTSLC99	ISTSLUBR	An unexpected combination of conditions was encountered during border node request finite state machine processing.	Mini

Table 48 (Page 7 of 7). FFST Probes for Session Services LU-LU (ISTSLCxx)

Probe Name	Module	Description	Dump Type
ISTSLCA0	ISTSLUFR	The finite state machine router was called for a situation where a FSM is not used.	Mini
ISTSLCA1	ISTSLSNO	Route Selection Control Vector (RSCV) missing from the session initiation control block's control vector list.	Mini
ISTSLCA2	ISTSLUAS	An unexpected combination of conditions was encountered during dependent LU requester search finite state machine processing.	Full
ISTSLCA3	ISTSLR2B	Missing subvector on CV2B (RSCV).	Full
ISTSLCA4	ISTSLRDI	The DLU is non-native, but the previously stored DLU information and the DLU information on the SESS_INIT_INFO_REQ IPS just received do not match.	Mini
ISTSLCA5	ISTSLRDP	The DLU is non-native, but the previously stored DLU information and the DLU information on the SEARCH_RPY IPS just received do not match.	Mini

Storage Management Probes

Table 49. FFST Probes for Storage Management (ISTORCxx)

Probe Name	Module	Description	Dump Type
ISTORC01	ISTORCHB	Unexpected I/O condition was detected. The output contains the buffer pool control block (BPCB) and one page of storage from each buffer pool expansion block (PXB).	Mini

Topology and Routing Services Probes

Table 50 (Page 1 of 2). FFST Probes for Topology and Routing Services (ISTTRCxx)

Probe Name	Module	Description	Dump Type
ISTTRC01	ISTTRCRT	An unknown work element is received.	Mini
ISTTRC02	ISTTRCRT	An unknown signal is contained with the work element.	Mini
ISTTRC03	ISTTRQDN	Issued when attempting to clean up storage for a node information block that is not found.	Full
ISTTRC04	ISTTRIFP	While processing a topology database update (TDU), an unrecognized finite state machine state is encountered.	Mini
ISTTRC05	ISTTRINP	An internal parsing error has occurred. An Input_TDU is encountered with a mismatch between the number of node vectors and the number of associated transmission group (TG) vectors.	Mini
ISTTRC06	ISTTRPSH	An unrecognized transmission group (TG) type is received on a RQ_SINGLE_HOP_ROUTE signal.	Mini
ISTTRC07	ISTTRQDT	Unable to delete a node from a tree because the node cannot be found on the tree.	Full
ISTTRC08	ISTTRTLT	An unrecognized partner node type is received on a TG_UPDATE signal.	Mini

Table 50 (Page 2 of 2). FFST Probes for Topology and Routing Services (ISTTRCxx)

Probe Name	Module	Description	Dump Type
ISTTRC09	ISTTRTLT	The transmission group (TG) direction specified on a TG_UPDATE signal is not recognized.	Mini
ISTTRC10	ISTTRRPP	A problem is encountered while attempting to uncache the session trees from the topology and routing services (TRS) database.	Full
ISTTRC11	ISTTRRPP	While scanning the topology and routing services (TRS) database an unidentified resource is found.	Mini
ISTTRC12	ISTTRDGM	An unrecognized secondary LU (SLU) node identification is received on a REQ_MULTIPLE_ROUTES signal.	Mini
ISTTRC13	ISTTRRRR	A route to a gateway node is received on a REQUEST_ROUTE; however, the signal contained unrecognized information.	Mini

Trace Services Probes

Table 51. FFST Probes for Trace Services (ISTRACxx)

Probe Name	Module	Description	Dump Type
ISTRAC01	ISTRACZT	FFST dump from the VTAM trap module.	Full
ISTRAC02	ISTRACZT	FFST dump from the VTAM trap module consisting of the ATCVT and the VTAM internal trace (VIT).	Mini
ISTRACZ3 VM	ISTITCAS	FFST full dump for a predetermined APPC sense code	Full
ISTRACZ4 VM	ISTITCAS	FFST full dump for a predetermined RPL6 return code	Full

Transmission Subsystem Probes

Table 52 (Page 1 of 2). FFST Probes for Transmission Subsystem (ISTTSCxx)

Probe Name	Module	Description	Dump Type
ISTTSC01	ISTTSCRI	Inbound path information unit (PIU) with an out-of-order sequence number was detected. The output includes the transmission subsystem control block (TSCB) containing the PIU with the unexpected sequence number and the virtual route control block (VRBLK) containing the expected sequence number.	Mini
ISTTSC02	ISTTSC8S	Unrecognized multipath channel sweep work element was detected.	Mini
ISTTSC03	ISTTSC8E	Unexpected multipath channel sweep finite state machine state was detected.	Mini
ISTTSC04	ISTTSC8I	Zero active read subchannel count was detected.	Mini
ISTTSC05	ISTTSC8E	Zero active write subchannel count was detected.	Mini
ISTTSC06	ISTTSC8E	Zero read sweep count decremented.	Mini
ISTTSC07	ISTTSC8E	Zero write sweep count decremented.	Mini
ISTTSC08	ISTTSC9L	Unexpected input in current state was received.	Mini
ISTTSC09	ISTTSC9C	Unexpected input in current state was received.	Mini

Table 52 (Page 2 of 2). FFST Probes for Transmission Subsystem (ISTTSCxx)

Probe Name	Module	Description	Dump Type
ISTTSC10	ISTTSC8E	Multipath channel incorrect path information unit (PIU) was built.	Mini
ISTTSC11	ISTTSC8D	Unexpected input was received.	Mini
ISTTSC12	ISTTSC8E	Multipath channel group lost the last read device.	Mini
ISTTSC13	ISTTSC8E	An attempt to send a data element response failed.	Mini
ISTTSC14	ISTTSC8E	Multipath channel device busy was detected.	Mini
ISTTSC15	ISTTSCM8	Multipath channel PU services process anchor block (PUPAB) finite state machine error was detected.	Mini
ISTTSC16	ISTTSCM8	Unable to obtain storage for a request unit processing element (RUPE).	Mini
ISTTSC17	ISTTSC8X	Multipath channel PUPAB finite state machine error was detected.	Mini
ISTTSC18	ISTTSC8X	Unable to obtain storage for a request unit processing element (RUPE).	Mini
ISTTSC19	ISTTSC8B	Empty node control block (NCB) pending queue was detected.	Mini
ISTTSC20	ISTTSC87	Unexpected value for request/response unit (RU) was received.	Mini
ISTTSC21	ISTTSC9X	Unexpected input in current state was received.	Mini
ISTTSC22	ISTTSC9P	Unexpected input was received.	Mini
ISTTSC23	ISTTSC9D	Unexpected input was received.	Mini
ISTTSC24	ISTTSC8I	Unexpected multipath channel sweep finite state machine state was detected.	Mini
ISTTSC25	ISTTSC8I	Zero active write subchannel count was detected.	Mini
ISTTSC26	ISTTSC8I	Zero read sweep count decremented.	Mini
ISTTSC27	ISTTSC8I	Zero write sweep count decremented.	Mini
ISTTSC28	ISTTSCM8	Unexpected input was received.	Mini
ISTTSC29	ISTTSC8X	Unexpected input was received.	Mini
ISTTSC30	ISTTSC6F	ISTTSC6F fails to dequeue the TRGCB from ATCNCBQ during deactivation of a major node. This can result in an abend in ISTTSCWU.	Mini

Appendix D. Channel Programs

This appendix describes the normal sequence of I/O channel control words (CCWs) within channel programs for the communication controller and channel-attached devices. If you determine that the problem is in an I/O sequence, you need to know the normal sequence of I/O CCWs within these channel programs. With a dump, the VIT trace with the CIO option, or a CCWTRACE (if available), you can compare the sequence that happened with the expected sequence. When there is a deviation, you can then look at status and sense bytes returned from the communication controller or the cluster controller for information that can help you determine the location of the error condition.

This appendix includes the following sections:

Channel Programs for Channel-Attached Type 2 and 4 Physical Units

The ICNCB represents type 2 and 4 physical units, and contains addresses and CCWs needed for channel programs. Figure 84 shows the following:

- ICNCB
- Location in storage of various CCWs
- Write buffers required for writing three PIUs, each of which is contained in a single buffer. Write or Write-Break CCWs alternate with transfer-in-channel (TIC) CCWs.

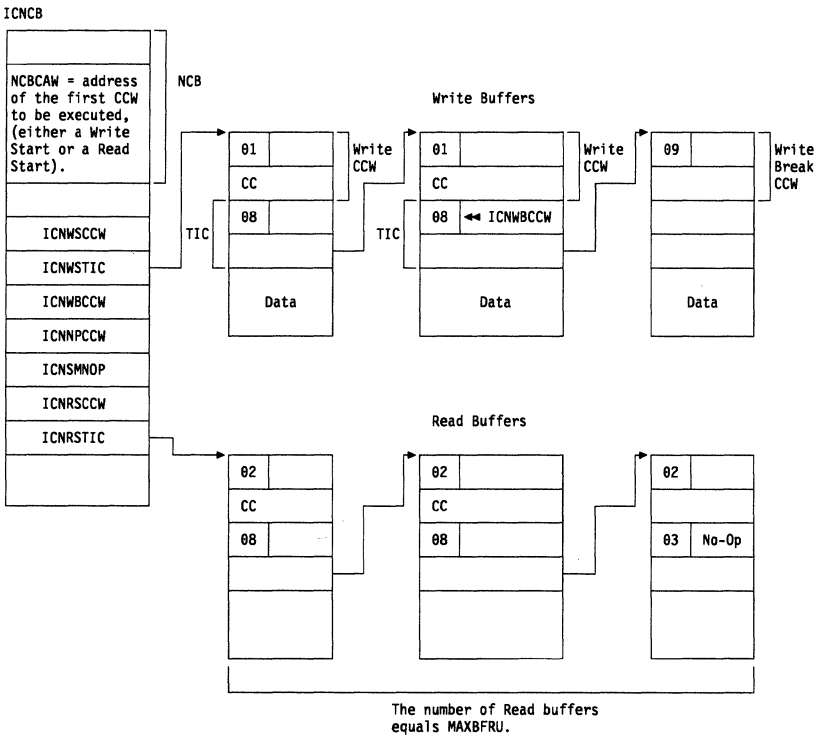


Figure 84. Data Areas Used by Channel Programs for PU Types 2 and 4

Channel Commands for Channel-Attached Type 2 and 4 Physical Units

Table 53 describes the channel commands used by the VTAM program to write data to and read data from channel-attached type 2 and 4 physical units.

Table 53 (Page 1 of 2). VTAM Channel Commands for Type 2 and 4 Physical Units

Command Code (hex)	Command	Description	Contents of Address Field
01	Write	Transfers data from storage in the host processor to the I/O device.	Output area
02	Read	Transfers data from the I/O device to storage in the host processor.	Input area
03	No-op	Causes the channel to respond with a channel end and device end. It is the last CCW in a read or write chain.	Zero
04	Sense	Transfers 1 or 2 bytes of sense data to storage in the host processor.	Address of sense data area
08	TIC (Transfer in Channel)	Causes the channel to fetch an instruction that is not the next sequential instruction within the channel program sequence.	Address of next CCW to be executed
09	Write Break	Transfers data from storage in the host processor to the I/O device and indicates that it is the last or only Write command in a chain of Write CCWs.	Output area
31	Write Start 0	Begins a Write sequence. Alternates with Write Start 1.	Zero
32	Read Start 0	Begins a Read sequence. Alternates with Read Start 1.	Zero
51	Write Start 1	Begins a Write sequence. Alternates with Write Start 0.	Zero
52	Read Start 1	Begins a Read sequence. Alternates with Read Start 0.	Zero
Note: Data transfer does not occur on Read-Start or Write-Start commands.			
61	Write XID	The host sends the Write XID command to signal the NCP that a channel contact sequence is beginning and to prepare to receive the host's XID.	Zero
62	Read XID	The host sends the Read XID command to signal the NCP that the host expects to read the NCP's XID.	Zero

Table 53 (Page 2 of 2). VTAM Channel Commands for Type 2 and 4 Physical Units

Command Code (hex)	Command	Description	Contents of Address Field
A3	Discontact	Indicates that the channel is no longer contacted and the attachment to the transmission group should be broken. Releases the PIUs on the channel hold and intermediate queues.	Zero
C3	Contact	Establishes contact between the host and the NCP. Tells the NCP to use XID information for operations with the host.	Zero
93	Restart	Causes the controller to reset its switches to indicate that the last Write-Start and Read-Start commands were Write-Start-1 and Read-Start-1 commands.	Zero

Format of Transfer-in-Channel (TIC) CCWs

VM VSE The Format 0 TIC CCW is formatted as follows:

Byte (hex)	Contents
00	X'08' (TIC identifier)
01–03	Real address
04–07	Reserved.

MVS VM The Format 1 TIC CCW is formatted as follows:

Byte (hex)	Contents
00	X'08' (TIC identifier)
01–03	Zero
04–07	Real address.

A doubleword TIC extension immediately follows both the Format 0 TIC and the Format 1 TIC. VTAM uses the last 4 bytes of the TIC extension to contain the virtual address of the next buffer in the chain.

The TIC extension is formatted as follows:

Byte (hex)	Contents
08–0B	Reserved
0C–0F	Virtual address.

For write buffers, the next to the last physical buffer is handled specially. The real address of the TIC points to the last Write-Break CCW (ICNWBCCW), but the virtual address points to the last write buffer that contains data (see Figure 85 on page 877.) This last buffer is formatted with a Write-Break CCW which is not used but is copied into ICNWBCCW. If only one buffer exists in the channel program, the Write-Start TIC is formatted so that the real address points to ICNWBCCW, but the virtual address points to the only write buffer.

Note that the last write buffer looks unusual because the Write-Break command is chained, but the next CCW is zero. This Write-Break CCW is never physically executed by the channel, but the copied version of the CCW (in the ICNCB) is executed.

Figure 85 shows the write buffers required for writing two PIUs when each spans three buffers.

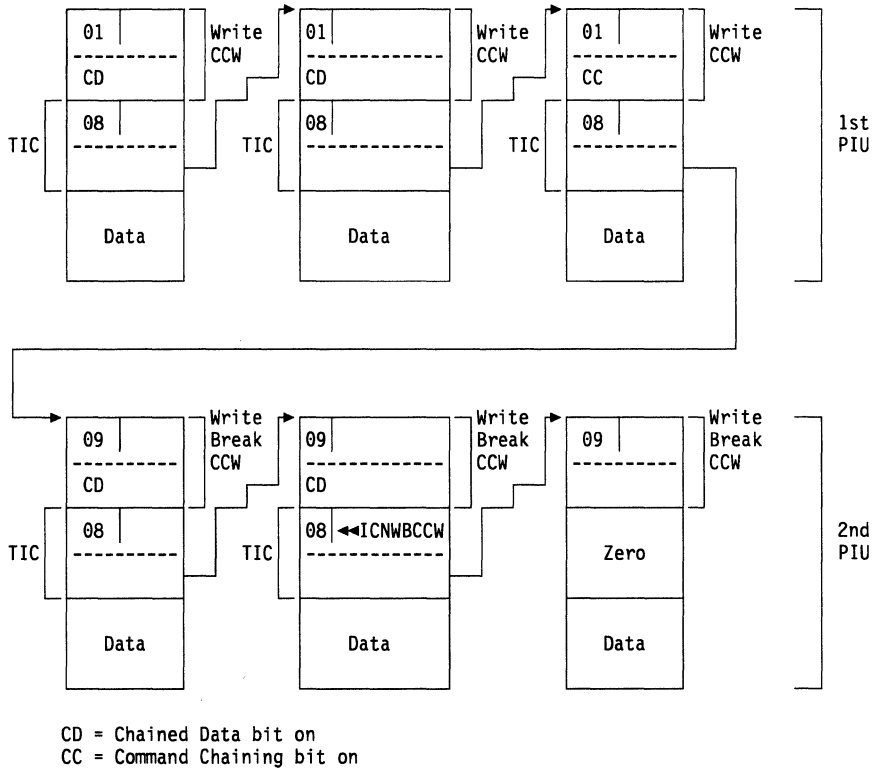


Figure 85. Format of Write CCWs with Chained Data

Channel Program Sequences

The following describes the CCWs, in the order that they are executed, in a typical WRITE channel program.

1. *ICNWSCCW - Write Start:* Each time a Write sequence (the Write start and all associated write buffers) has completed successfully, the Write-Start CCW is alternated: The first Write-Start operation code is X'31' (Write Start 0), the second Write-Start code is X'51' (Write Start 1), the third Write-Start code is X'31', and so on. There is no data transfer associated with this CCW; it is used to inform the controller that the previous Write has successfully completed.
2. *ICNWSTIC - Write Start TIC:* This CCW is used to chain to the first Write CCW. When the WRITE channel program consists of a single write buffer, the real address points to ICNWBCCW (the last Write Break). The virtual address of this CCW always points to the first write buffer on a chain or is zero when no WRITE channel program is pending or active for the device.
3. *Write or Write-Break CCWs Alternating with TICs:* Figure 84 on page 874 shows the write buffers required for writing threeWrite PIUs, each of which is

contained in a single buffer. Figure 85 shows the write buffers required for writing two PIUs, when each spans three buffers.

4. *ICNWBCCW CCW(3) - Last Write Break:* In a channel program this is the last Write CCW that is physically executed by the device. The data address points to the last write buffer that contains data. If a Read is requested, when this operation completes, the controller will signal a status modifier (in addition to channel end and device end). This causes ICNNPCCW to be skipped by the channel.

5. *ICNNPCCW - No-Op:* If a Write completes and no Read is requested, No-Op is the ending CCW in a channel program. If a Read was requested, this CCW is skipped by the channel. The command chain flag will be on in this CCW if a Read was previously requested but was not previously initiated, and read buffers are available.

6. *ICNSMNOP - Status Modifier No-Op:* This CCW receives control after the last Write Break if a Read is signaled. This CCW normally command chains to the Read Start so that writing and reading occur without interruption. If no buffers are available, however, the channel program ends here, and VTAM recognizes that a Read is required.

7. *ICNRSCCW - Read Start:* Each time a Read sequence_ (the Read Start and at least one Read) has completed successfully, the Read-Start CCW is alternated.

   The first Read-Start generation code is X'32' (Read Start 0), the second Read-Start generation code is X'52' (Read Start 1), the third Read-Start generation code is X'32', and so on.

   As with the Write-Start CCW, there is no data transfer. The CCW alternation is used to inform the controller that the previous Read has successfully completed.

   When a Write is not required, but a Read has been requested by an attention status, Read Start is the first CCW in the channel program.

8. *ICNRSTIC - Read Start TIC:* This CCW is used to chain to the first Read CCW. The virtual address of this CCW points to the first read buffer on the chain, except:

   - When deblocking PIUs, in which case the virtual address is changed by the channel end appendage.
   - When there are not enough read buffers available, in which case it is zero.

9. *Read CCWs Alternating with TICs:* Figure 84 on page 874 shows the buffers required for reading three PIUs.

10. *Read No-Op:* This CCW should never be executed. If it is, it indicates that the controller and VTAM do not agree on how many read buffers are required.

    The normal ending status for a Read is channel end, device end, attention, or unit exception. The unit exception indication is presented on the Read CCW that has completed data transfer and terminates the command chaining. The attention indication is the same as the unit exception indication, but it also means that a Read is requested.

    Unit exception may also be presented to the Read-Start CCW. It is used by the controller to release input buffers. If the same Read Start (as opposed to

the alternate Read Start) is given to the controller, the data buffers must be re-sent.

PUNS-Related Channel Programs

During activation of an NCP in a channel-attached communication controller, the SSCP sends a Contact RU to PUNS. PUNS responds by giving control to ISTTSCP4 to schedule one of five channel programs (A–E in Table 54). When the SSCP sends a Discontact RU to PUNS, one of two channel programs (F and G in Table 54) is executed.

Table 54. PUNS-related Channel Programs

	CCW	Code (hex)	Flags	Notes
A.	Sense	04	SLI	Determines if the device needs to be loaded.
B.	Write XID No-Op	61 03	SLI,CC SLI	This channel program follows A if the device does not need to be loaded. It is followed by either C or D.
C.	Write Break Read XID Read No-Op	09 62 02 03	SLI,CC SLI,CC SLI,CC SLI	This channel program follows B if there is no command reject.
D.	Restart/Reset No-Op	93 03	SLI,CC SLI	Executed only if the Write XID in B caused a command reject (implies the NCP is NCP Release 2 or earlier).
E.	Contact Restart/Reset No-Op	C3 93 03	SLI,CC SLI,CC SLI	Restart/Reset is executed only if a Contacted (error) response is sent to PUNS.
F.	Discontact No-Op	A3 03	SLI,CC SLI	This channel program is executed only if B did not cause a command reject.
G.	No-Op	03	SLI	This channel program is executed if B caused a command reject.

These CCWs are contained in I/O buffers that are allocated from the IOBUF buffer pool when doing PUNS I/O.

The data area pointed to by the address portion of a Read XID or Write XID CCW is described in ISTXID in *VTAM Data Areas for MVS/ESA* and *VTAM Data Areas for VM/ESA*.

Channel Programs for Channel-to-Channel Adapters (CTCA), Multipath Channels (MPC), and APPN Host-to-Host Channels

VTAM's CTC function supports two protocols: blocking and non-blocking. If VTAM is communicating with another VTAM through the CTC adapter and both VTAMs support the blocking protocol, blocking is the chosen protocol. If one VTAM does not support blocking protocols, the non-blocking protocol is used.

You cannot specify the protocol choice during system definition. Because the blocking protocol is the preferred mode, it is used if both VTAMs support it.

Channel Programs for Activating the CTC Connection

Each side of a channel-to-channel adapter (CTCA) is represented by a cross-channel node control block (XCNCB). In addition, each side has a physical unit service I/O (PIO) control block and a station control block (SCB). The PIO is used for exchange ID (XID) channel programs (the physical unit services I/O that occurs before the link is active). The PIO is mapped by ISTPIO. The SCB is a station work area where CCWs for normal data transfer are built. It is not mapped, and is *not* described in *VTAM Data Areas*.

A series of channel programs are issued when the operator activates the CTC connection. These I/O exchanges are used by the hosts to communicate various capabilities to the other host. The capabilities are transferred via the XID channel program.

The three primary pieces of information gained through the XID exchange are:

- Choice of protocol – blocking or non-blocking
- Determination of who is X-side and who is Y-side
- I/O buffering information to use once the connection is active.

Protocol Choice

As mentioned earlier, VTAM always chooses the blocking protocol if the partner VTAM is capable of supporting it. If the XID indicates it is not capable, the non-blocking protocol is used.

X-Side / Y-Side

Figure 86 on page 882 shows how VTAM determines which side will be the X-side and which side will be the Y-side. In this example, the operator in subarea 4 is the first to activate the link. Subarea 4 begins as the X-side and then switches to the Y-side.

I/O Buffering

During the XID exchange, each host informs the other about its read buffer capability.

In the blocking protocol, the total size of the single read buffer is communicated. The write buffer in the other host is allocated based on the size of the Read buffer.

In the non-blocking protocol, the total number of read buffers available, as well as the size of each buffer, is communicated. Each host must then allocate the write portion of its channel program to match the read portion of the other host.

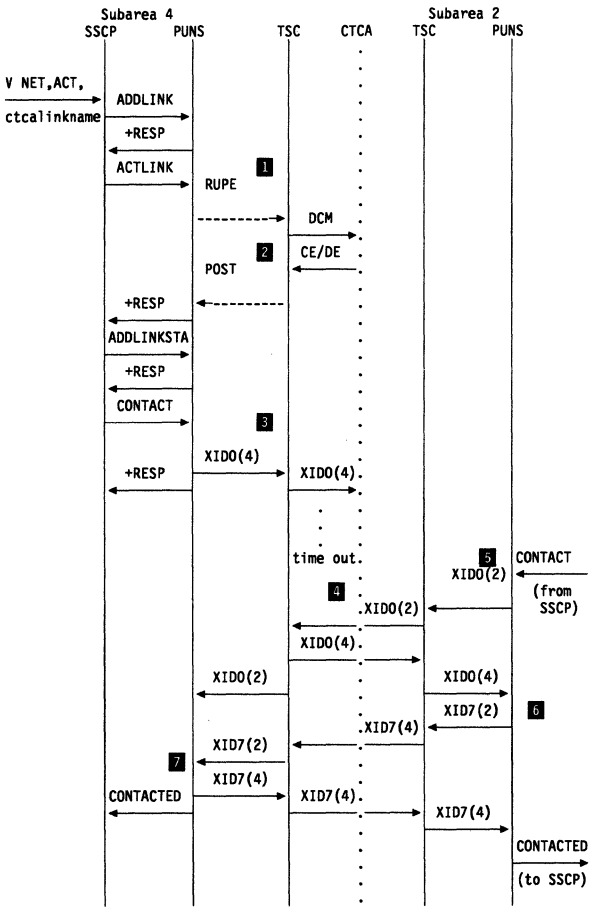


Figure 86. Example of an XID Exchange

CTCA channel-to-channel adapter
PUNS Physical unit services
SSCP System services control point
TSC Transmission subsystem component

Note: (2) and (4) refer to Subarea 2 and Subarea 4.

1 PUNS queues a RUPE to NCBPUPAB, causing TSC to get control.

2 TSC posts PUNS.

3 Subarea 4 assumes X-side protocols because it is initiating the XID exchange.

4 The XID exchange "times out" because the link in subarea 2 has not been activated. The CONTACT is issued in subarea 2, and upon attention, subarea 4 assumes Y-side protocols.

5 Assume the operator activates the link in subarea 2. Subarea 2 assumes X-side protocols.

6 The subarea with the lower subarea number (subarea 2) responds first to the XID0 exchange by sending an XID7. (XID7 is an XID Format 2 with the contact option field set to 7.)

7 Subarea 4 checks the XID7 from subarea 2 and responds with its XID7. It also sends a CONTACTED RU to the SSCP.

Channel Commands for Channel-to-Channel (CTC) Adapters

Table 55 contains the channel commands used for channel programs between two hosts connected by a channel-to-channel adapter.

Note: The Multipath channel (MPC) connection only uses these commands during activation and deactivation. For the channel commands used for data flow, when the MPC is in the CONTACTED-ACTIVE phase, see "Channel Programs for Multi-path Channel (MPC)" on page 895.

Table 55 (Page 1 of 2). Channel Commands for Channel-to-Channel Adapters

Command Code (hex)	Command	Description
01	Write	Transfers data from storage in this host processor to the CTC adapter (CTCA). For CTCAs this CCW is used to transfer status information from XCNOCTL, XID information from PIOPOTXT, validity checking information from XCNOVTXT, and data from buffers. Status information, mapped by XCNOCTL, is transferred in the first 8 bytes of the write buffer, and only one write buffer is used. Figure 89 on page 888 illustrates the buffer usage. For MPC connections this CCW, preceded by a Prepare command, is used to transfer data from the CPNCB transmit buffer. The transmit buffer is defined by YCNOBUF of the YCNCB contained within the CPNCB. The first 8 bytes of the transmit buffer is mapped by ISTBKHDR and contains control information about the current data transfer. Figure 95 on page 898 illustrates the MPC buffers used for normal data transfer.
02	Read	Transfers data from the CTCA to this host processor. For CTCAs, this CCW is used to read status information into XCNICTL, XID information into PIOPITXT, validity checking information into XCNIVTXT, and data into buffers. Status information, mapped by XCNICTL, is transferred in the first 8 bytes of the read buffer, and only one read buffer is used. Figure 89 on page 888 illustrates the buffer usage. For MPC, this CCW is used to read normal data into the CPNCB transmit buffer. The transmit buffer is defined by YCNIBUF of the YCNCB contained in the CPNCB. Figure 95 on page 898 illustrates the MPC buffers used for normal data transfer.

Table 55 (Page 2 of 2). Channel Commands for Channel-to-Channel Adapters

Command Code (hex)	Command	Description
03	No-Op	Causes the channel to respond with a channel end or device end. It is the last CCW in a read or write chain. The No-Op command does not apply to the MPC connection.
08	TIC	Causes the channel program to execute an instruction that is not the next sequential instruction within the channel program sequence. The TIC command does not apply to the MPC connection.
14	Sense Command Byte (SCB)	The SCB is normally issued in response to an attention generated when the adapter processes a WCTL from the other side. The SCB clears WCTL from the adapter, allowing the WCTL CCW to complete.
17	Write Control (WCTL)	Causes an attention interruption on the other side of the channel-to-channel adapter. The WCTL is issued to alert the other side that a channel program is active at the adapter.
43	Enable Compatibility Mode (ECM)	Prepares the adapter to operate in System/360* (compatibility) mode. The ECM is issued when VTAM gives up control of the adapter.
C3	Disable Compatibility Mode (DCM)	Prepares the adapter to operate in System/370* (Extended) mode. The DCM is issued when VTAM acquires control of the adapter.
E3	Prepare	Primes the CTC adapter for the next CCW. This CCW does not cause an attention on the other side of the adapter. The Prepare command applies only to the MPC connection.

XID Channel Program (X-Side)

Figure 87 on page 886 shows the data areas associated with the following XID channel program.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
1	Write Control	17	CC,SLI	Zero	1
2	TIC	08	—	WRITE CCW	—

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
3	Write	01	CC,SLI	XCNOCTL	8
4	Write	01	CC,SLI	PIOPOTXT	Length of XID
5	TIC	08	—	READ CCW	—
6	Read	02	CC,SLI	XCNICTL	8
7	Read	02	CC,SLI	PIOPITXT	Length of XID
8	TIC	08	—	READ CCW	—
9	Read	02	CC,SLI	XCNOVTXT	4

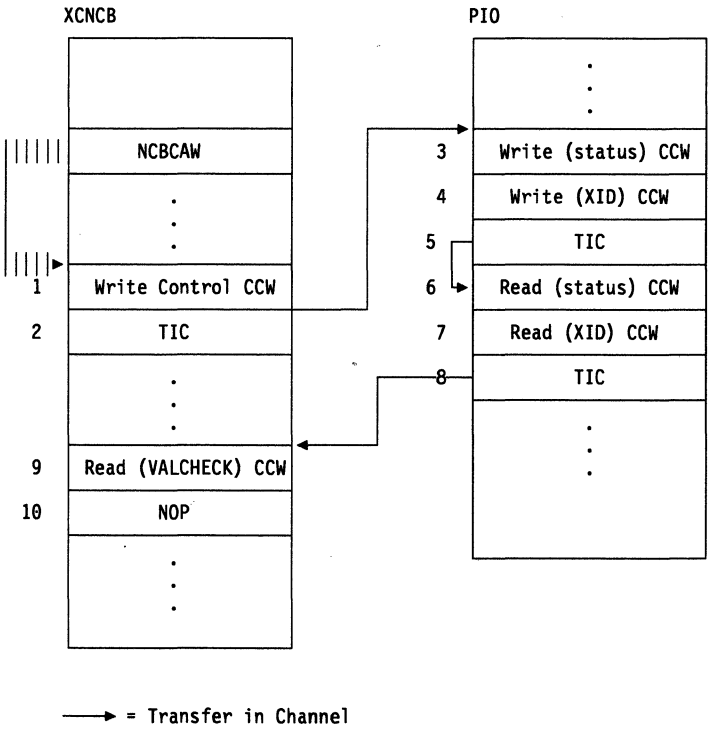


Figure 87. Data Areas Used for XID Channel Programs (X-Side)

XID Channel Program (Y-Side)

Figure 88 on page 887 shows the data areas associated with the following XID channel program.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
1	Sense Command Byte	14	CC,SLI	Zero	1

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
2	TIC	08	—	READ CCW	—
3	Read	02	CC,SLI	XCNICTL	8
4	Read	02	CC,SLI	PIOPITXT	Length of XID
5	TIC	08	—	WRITE CCW	—
6	Write	01	CC,SLI	XCNOCTL	8
7	Write	01	CC,SLI	PIOPOTXT	Length of XID
8	TIC	08	—	WRITE CCW	—
9	Write	01	CC,SLI	SCNOVTXT	4

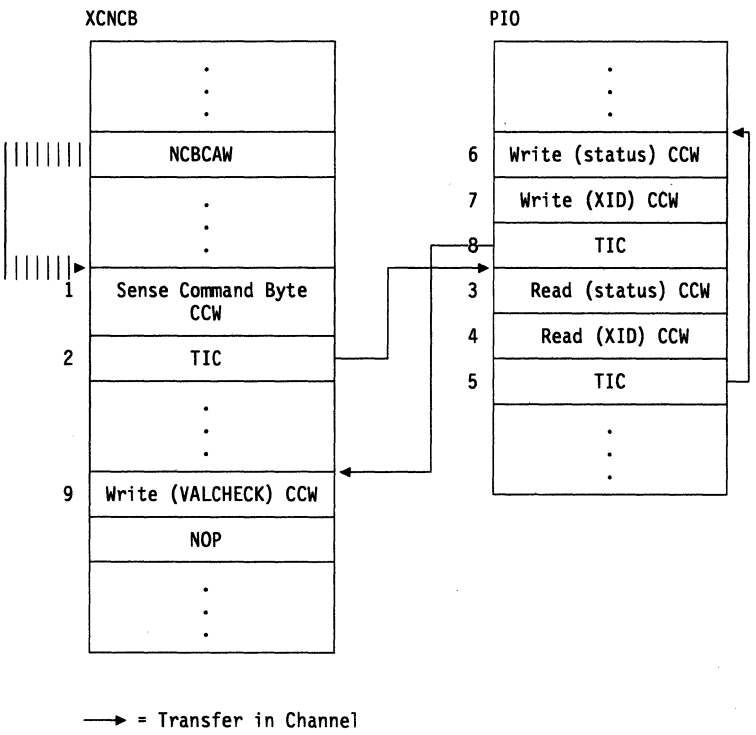


Figure 88. Data Areas Used for XID Channel Programs (Y-Side)

Channel Programs for CTC Data Transfer: (Blocking Protocol)

MAXBFRU defines a single read buffer. The write buffer in the other host is allocated to be the same size. PIUs are blocked for transfer, and are written by a single write, and read by a single read.

VTAM only uses three CCWs in its Data Transfer channel program: a Write CCW, a Read CCW, and another that is either a WCTL CCW or an SCB CCW. Output control information is included within the write buffer and input control information is received in the first bytes of the read buffer.

Normal data transfer for channel programs is a Write CCW followed by a Read CCW (on the X-side) or a Read CCW followed by a Write CCW (on the Y-side). VTAM uses the procedure described in "Channel Programs for Activating the CTC Connection" on page 881 to determine which side will be X and which will be Y.

During the XID exchange, each host informs the other host of the size (in pages) of the buffer that will be used in the read portion of its channel programs. Each host then allocates a write buffer to match exactly the read buffer of the other host.

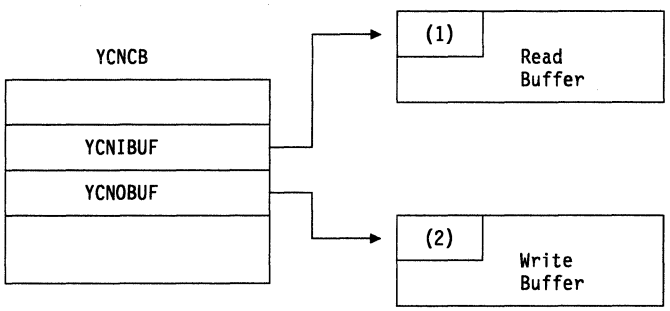


Figure 89. Buffers Used for Normal Data Transfer

The numbers 1 and 2 in Figure 89 represent:

(1) The first 8 bytes of the read buffer is the control information mapped by XCNICTL.

(2) The first 8 bytes of the write buffer is the control information mapped by XCNOCTL.

Normal Data Transfer (X-Side)

Figure 90 on page 889 shows the data areas associated with the X-side of a normal data transfer channel program.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
1	WCTL[2]	17/ 14	CC,SLI	Zero	1
Note: [2] WCTL is used when this host is initiating a write operation. SCB is used when this host is responding to an attention (because the other host has data that it wants this host to read).					
2	Write	01	CC,SLI	YCNWRIDA	Number of bytes to transfer
3	Read	02	SLI	YCNRDIDA	Total length of Read buffer

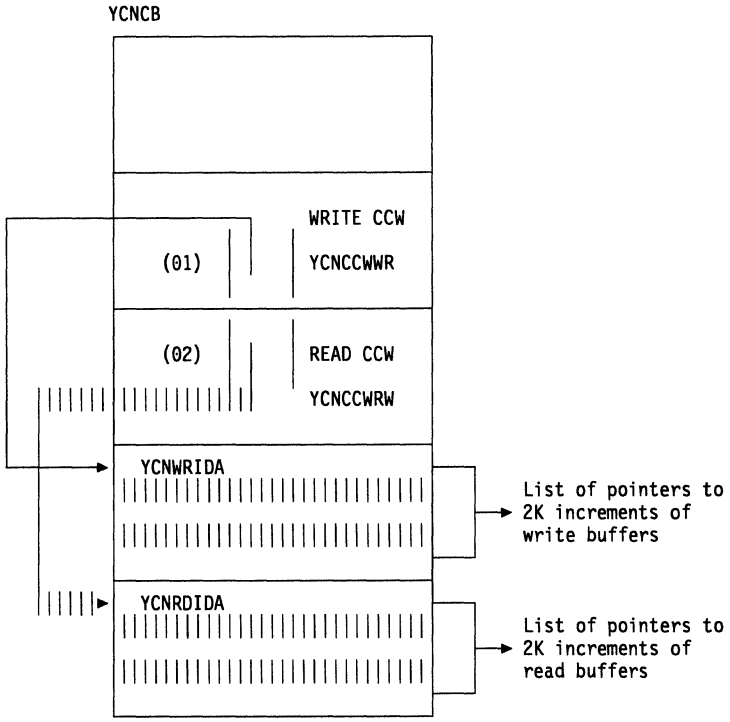


Figure 90. Data Areas Used for Normal Data Transfer (X-Side)

Normal Data Transfer (Y-Side)

Figure 91 on page 890 shows the data areas associated with the following normal data transfer channel program.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
1	WCTL[3]	17/ 14	CC,SLI	Zero	1
Note: [3] WCTL is used when this host is initiating a write operation. SCB is used when this host is responding to an attention (because the other host has data that it wants this host to read).					
2	Read	02	CC,SLI	YCNRDIDA	Total length of Read buffer
3	Write	01	SLI	YCNWRIDA	Number of bytes to transfer

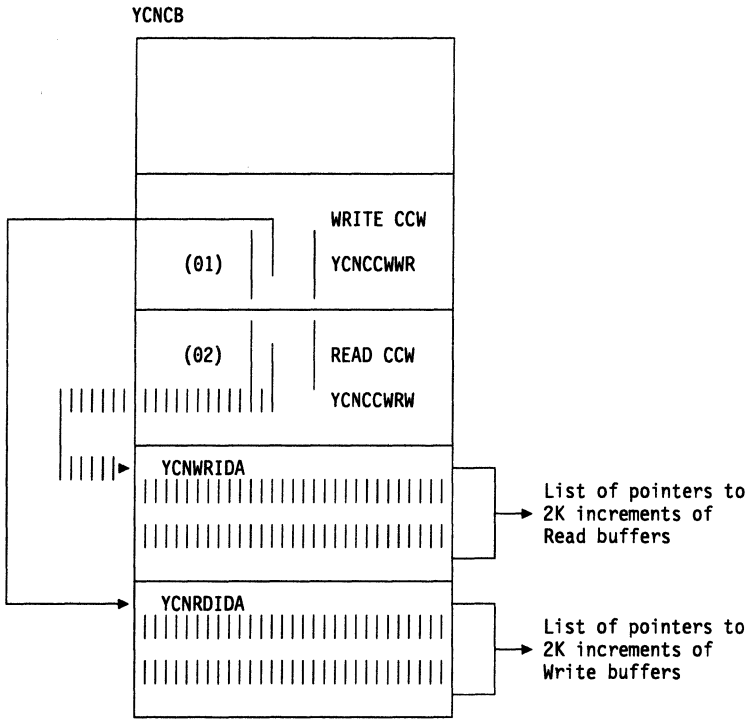


Figure 91. Data Areas Used for Normal Data Transfer (Y-Side)

Channel Programs for Channel-to-Channel Adapters: Non-Blocking Protocol

MAXBFRU defines the total number of IOBUF sized read buffers to use to receive data from the CTC. Consequently, the channel program is a series of writes followed by a series of reads for the X-side. For the Y-side, it is a series of reads followed by a series of writes.

Normal Data Transfer (X-Side) for Non-Blocking Protocols

Figure 92 on page 892 shows the data areas associated with the X-side of a normal data transfer channel program for non-blocking protocols.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
1	WCTL[4]	17/ 14	CC,SLI	Zero	1
Note: [4] WCTL is used when this host is initiating a write operation. SCB is used when this host is responding to an attention (because the other host has data that it wants this host to read).					
2	TIC	08	—	WRITE CCW	—
3	Write	01	CC,SLI	XCNOCTL	8
4	TIC	08	—	WRITE CCW	—
	Write	01	CC,SLI	Buffer	Length of data in this buffer
	:		(CD)		
	:				

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
5	Write : : Write	01 01	CC,SLI, (CD) CC,SLI	Buffer Buffer	Length of data in this buffer Length of data in this buffer
6	TIC	08	—	READ CCW	—
7	Read	02	CC,SLI	XCNICTL	8
8	TIC	08	—	READ CCW	—
	Read	02	CC,SLI	Address of Data	Length of data in this buffer
9	TIC : : Read	08 02	— CC,SLI	READ CCW Address of Data	— Length of data in this buffer
	TIC	08	—	WRITE CCW	—
10	Write	01	CC,SLI	XCNOVTXT	4

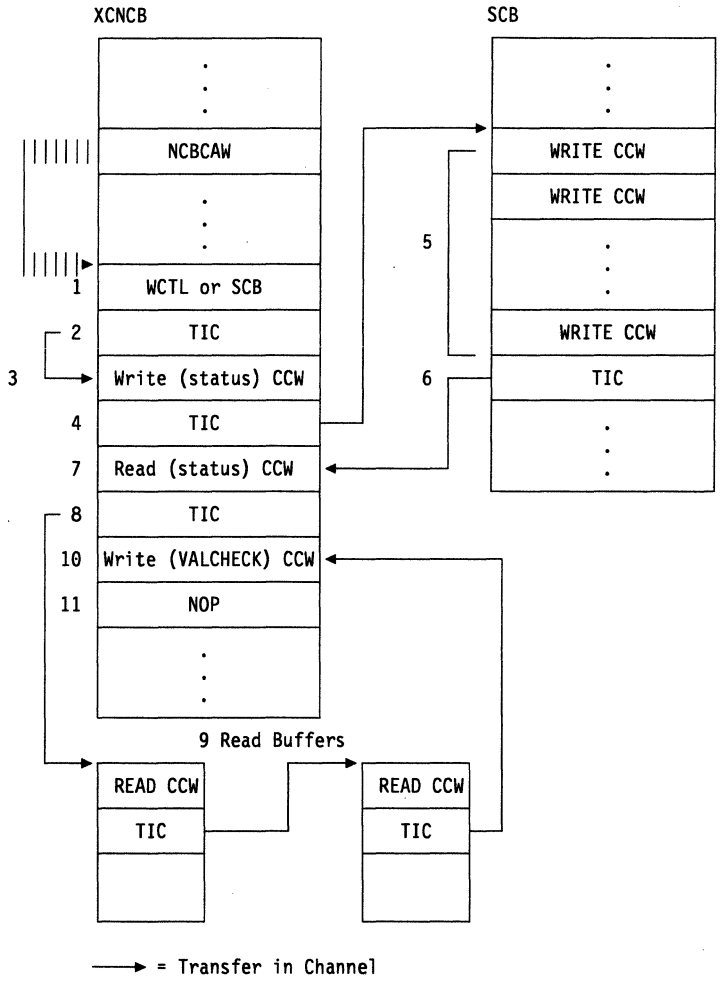


Figure 92. Data Areas Used for Normal Data Transfer (X-Side) Non-Blocking

Normal Data Transfer (Y-Side) for Non-Blocking Protocols

Figure 93 on page 894 shows the data areas associated with the Y-side of the following normal data transfer channel program for non-blocking protocols.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
1	WCTL[5]	14/ 17	CC,SLI	Zero	1
Note: [5] WCTL is used when this host is initiating a write operation. SCB is used when this host is responding to an attention (because the other host has data that it wants this host to read).					
2	TIC	08	—	READ CCW	—
3	Read	02	CC,SLI	XCNICTL	8
4	TIC	08	—	READ CCW	—
	Read	02	CC,SLI	Address of Data	Length of data in this buffer

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
	TIC	08	—	READ CCW	—
5	: :				
	Read	02	CC,SLI	Address of Data	Length of data in this buffer
	TIC	08	—	WRITE CCW	—
6	Write	01	CC,SLI	XCNOCTL	8
7	TIC	08	—	WRITE CCW	—
	Write : :	01	CC,SLI, (CD)	Buffer	Length of data in this buffer
8	Write : :	01	CC,SLI, (CD)	Buffer	Length of data in this buffer
	Write	01	CC,SLI	Buffer	Length of data in this buffer
9	TIC	08	—	READ CCW	—
10	Read	02	CC,SLI	XCNIVTXT	4

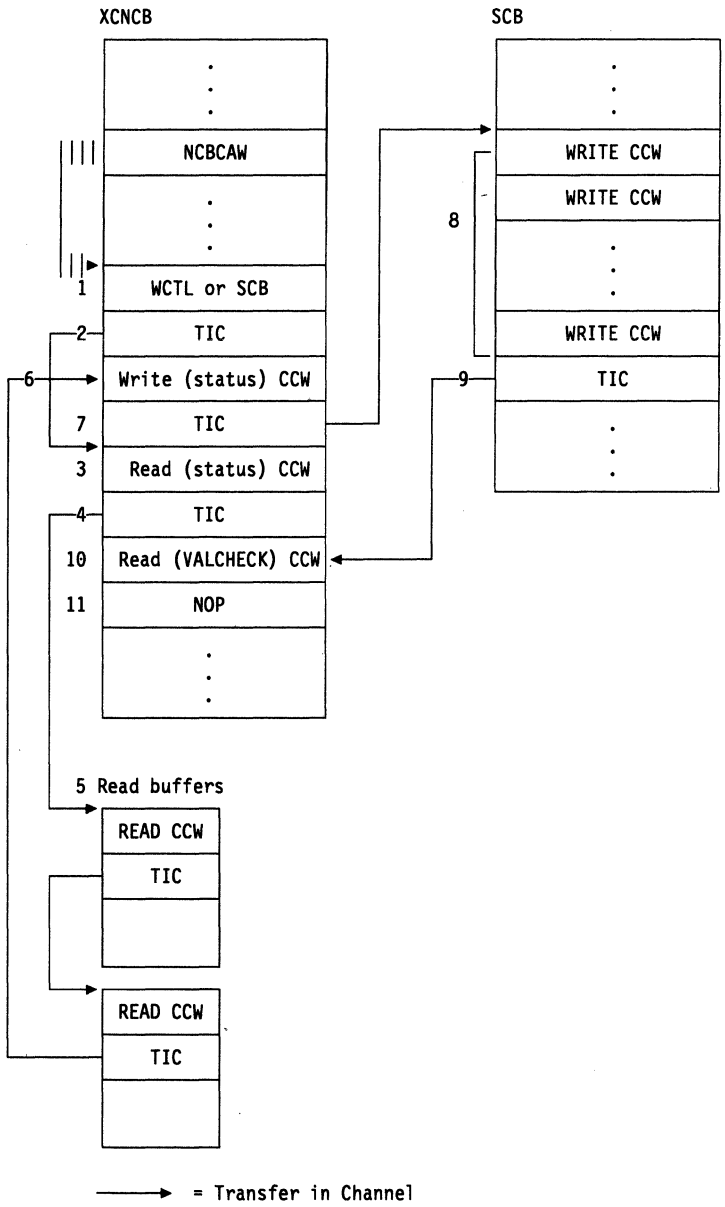


Figure 93. Data Areas Used for Normal Data Transfer (Y-Side) Non-Blocking

Channel Programs for Multipath Channel (MPC)

VTAM's MPC function is derived from the VTAM channel-to-channel (CTC) function. Both MPC and CTC communicate with the CTC Adapter, but MPC uses its own set of channel programs.

During normal data transfer, MPC uses the 'never-ending' Read channel program. This means that a subchannel defined as a Read device can have an outstanding Read channel program or can be processing the last channel program to complete. On subchannels defined as Write devices, a Write channel program is generated when data is available to be written to the CTC adapter.

The following sections describe the multipath channel programs for activating or deactivating an MPC connection as well as for normal data transfer.

Channel Programs for Activating the MPC Connection

A series of channel programs is issued when the operator activates the MPC connection. These I/O exchanges are used by the hosts to communicate various capabilities to the other host. The capabilities are transferred using the XID channel program.

The primary pieces of new information gained through the MPC XID exchange are:

- MPC to CTC connection
- Polarity of the device (Read or Write)
- I/O buffering information for an active connection.

Channel Program (X-Side or Y-Side)

Unlike CTCA, X-Side or Y-Side has meaning only during XID exchange for MPC.

Figure 94 on page 896 shows how VTAM determines which side is the X-side and which side is the Y-side.

In this example, the operator in subarea 1 is the first to activate the link. Because subarea 2 is not active, the XID exchange does not complete. Later, when the operator in subarea 2 activates the link, the XID exchange is completed.

I/O Buffering

During the XID exchange, each host informs the other about its read buffer size.

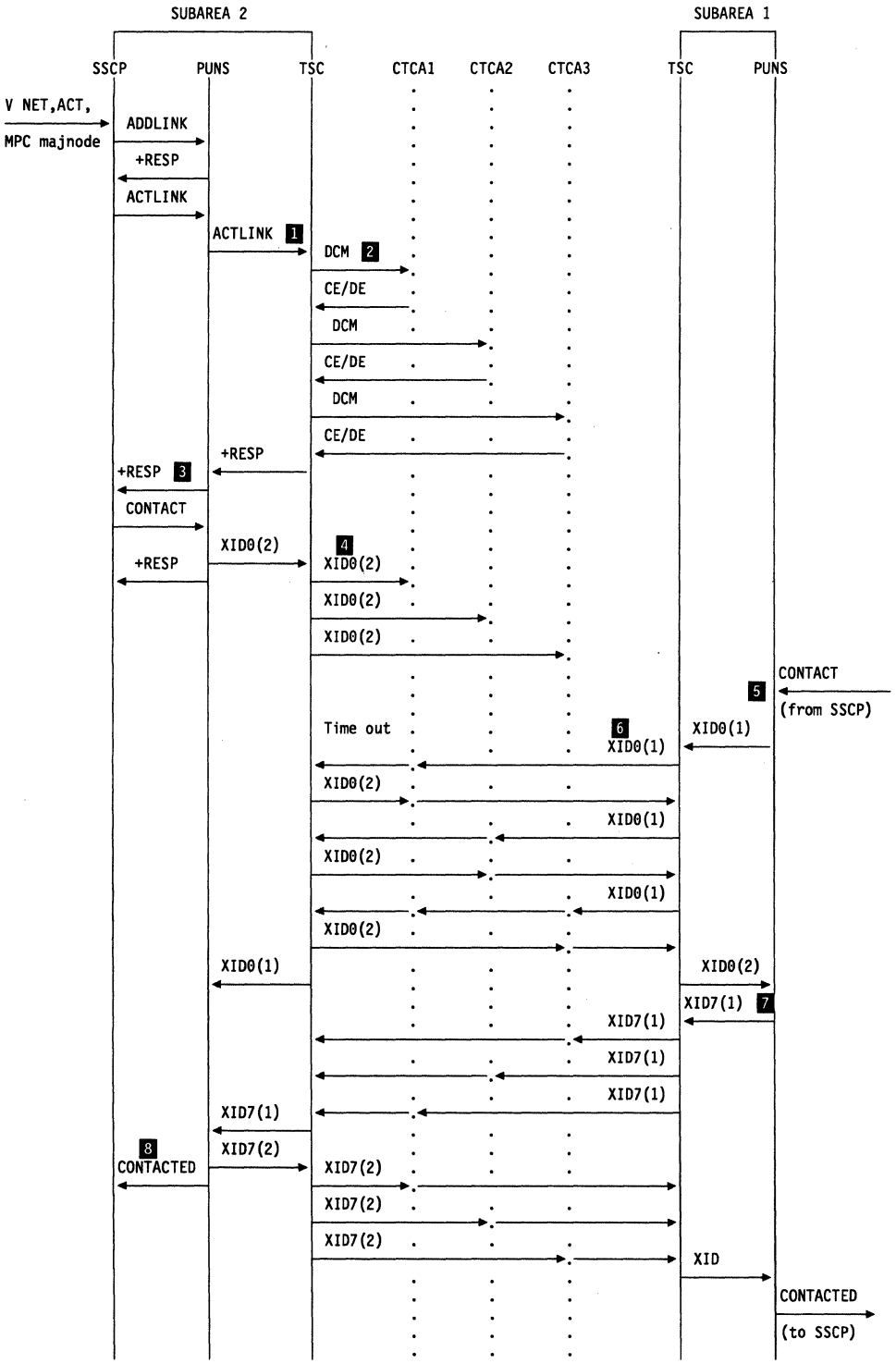


Figure 94. MPC Activation Flow

CTCA channel-to-channel adapter
PUNS Physical unit services
SSCP System services control point
TSC Transmission subsystem component

Note: When MPC is used for an APPN host-to-host channel connection, the flows between the SSCP, PUNS, AND TSC components differ somewhat from those in this figure. The XID flows are the same.

1 PUNS queues an ACTLINK RUPE to the MPNCB PUPAB, causing TSC to get control.

2 TSC issues a DCM channel program on every device in the MPC group.

3 A single ACTLINK response is passed to PUNS and SSCP indicating that the MPC path ACTLINK initialization is completed.

4 PUNS XID0 is copied to all MPC subchannels so that they are being written to all MPC subchannels. In this case, subarea 2's XID0s are timed out and the other side is not ready to read.

5 Assume the operator activates the link in subarea 2. Subarea 2 assumes X-side protocols.

6 Subarea 1 initiates writing XID0s at all MPC subchannels and then completes.

7 The subarea with the lower number (subarea 1) responds first to the XID0 exchange by sending an XID7.

8 Subarea 2 checks the XID7 from subarea 1 and responds with its XID7. It also sends a CONTACTED RU to the SSCP.

Channel Commands for Activating the MPC Connection

See "Channel Commands for Channel-to-Channel (CTC) Adapters" on page 884 for the commands used for activation and deactivation of the MPC connection.

The MPC XID channel programs are identical to the CTC packed format XID channel programs. See "XID Channel Program (X-Side)" on page 885 and "XID Channel Program (Y-Side)" on page 886.

Channel Programs for MPC Data Transfer

MAXBFRU defines a single read buffer. The write buffer in the other host has the same size allocation. PIUs are blocked for transfer, and are written by a single write, and read by a single read.

VTAM uses three CCWs in its data transfer channel program:

- Write CCW
- Read CCW
- Prepare CCW

MPC unique output control information is in the write buffer and input control information is received in the first 8 bytes of the read buffer.

Normal data transfer for channel programs is a PREP CCW followed by a Write CCW (Write device) or a Prepare CCW followed by a Read CCW (Read device).

During the XID exchange, each host passes buffer size information for the read portion of its channel programs to other hosts in the network. Each host allocates a write buffer to match the other host's read buffer.

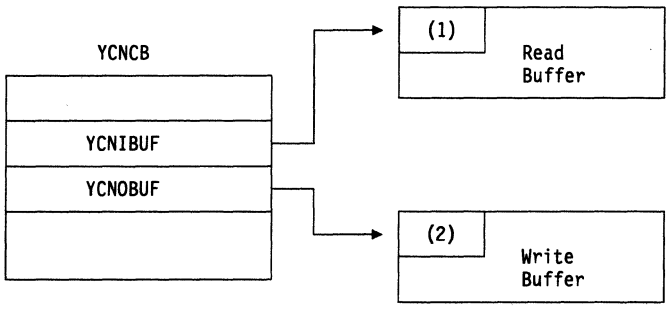


Figure 95. MPC Transmit Buffers Used for Normal Data Transfer

1. The first 8 bytes of the channel buffer is the control information mapped by ISTBKHDR.

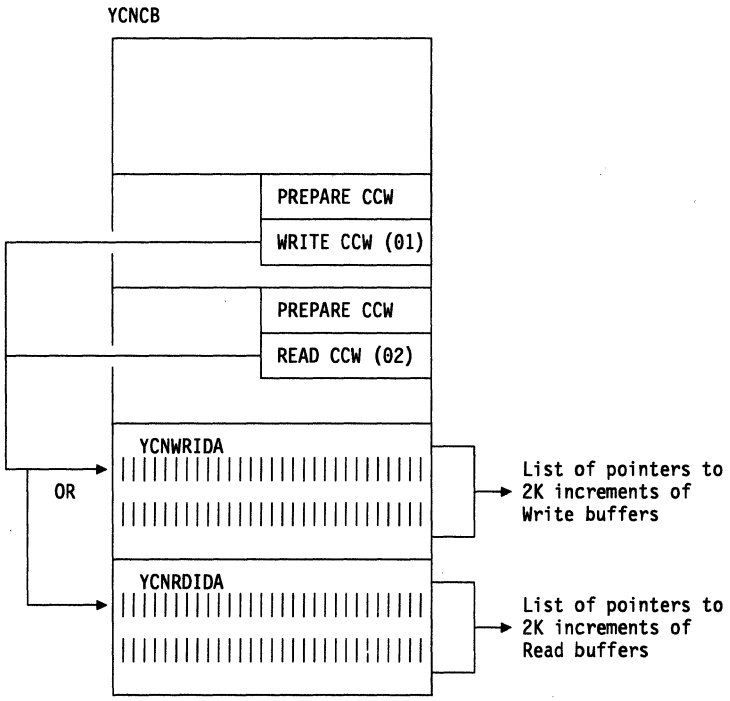


Figure 96. Indirect Address Word Structure of Multipath Channel Programs for Normal Data Flow

Channel Programs for Channel-Attached Non-SNA 3270 Devices

The publications for the non-SNA 3270 devices contain diagnostic procedures. For more information, see the documentation for your display type, or for your control unit. See Table 80 on page 1165 to determine what book contains information on the 3174 controller.

The LDNCB represents local devices and contains addresses and CCWs needed for channel programs. Figure 97 shows the LDNCB and the location in storage of various CCWs.

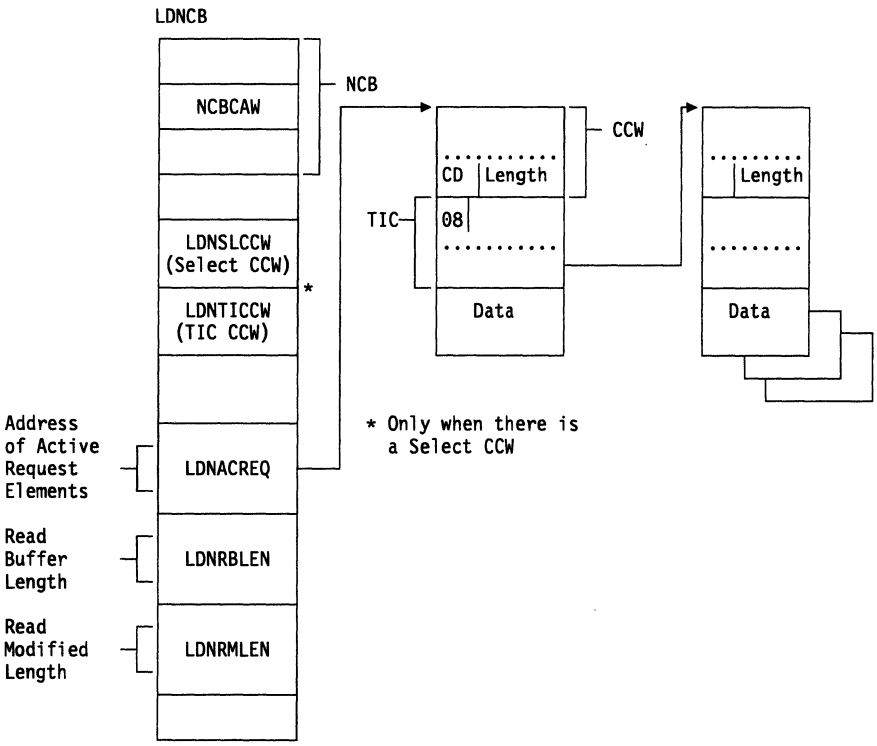


Figure 97. Data Areas Used by a Channel Program for Channel-Attached Non-SNA Devices

Channel Command Words

Table 56 contains the channel commands used by VTAM to send data to and receive data from channel-attached non-SNA 3270 terminals.

Table 56. Channel Command Words for Channel-Attached Non-SNA 3270 Devices

Command Code (hex)	Command	Description
01	Write	Transfers data from storage in the host processor to the I/O device. Modifies existing buffer data.
02	Read or Read buffer	Transfers the entire buffer contents from the I/O device to storage in the host processor.
05	Erase/Write	Clears the device buffer before starting the write.
06	Read Modified	Transfers data (that was modified since the last read) from the I/O device to storage in the host processor.
0B	Select	Transfers data from the I/O device buffer to the controller buffer. Separates the device-to-controller unit buffer transfer from the execution command.
0D	erase/write Alternate	Same as Erase/Write, except that it allows for a larger buffer size required for some devices.
1B	Select Read Buffer	Same as Select, except that it is only used for Read buffer channel programs.
4B	Select Write	Same as Select, except that it is only used for Write channel programs.
0F	Erase All Unprotected	Clears all unprotected buffers
11	Write Structured Field	Writes a structured field

Channel Programs

Table 57 through Table 62 on page 903 show the CCWs used in various channel programs. The order of execution is the same as the order in which they appear.

Write Data Channel Program

Table 57. Write Data Channel Program

CCW	Command Code (hex)	Flags	Address	Byte Count
Select	0B or 4B[9]	CC,SLI	Zero	1
TIC	08	—	Write CCW	—
Write[10] (As many as needed)	01	CD,SLI	Output Area	Length of data

Read Modified Channel Program

Table 58. Read Modified Channel Program

CCW	Command Code (hex)	Flags	Address	Byte Count
Select	0B	CC	Zero	1
TIC	08	—	Read Modified CCW	—
Read Modified[11] (As many as needed)	06	CD	Input Area	Length of data
TIC	08	—	Read Modified CCW	—
Read Modified[12]	06	CD	Input Area	Length of data
Read Modified (Skip)	06	SKIP	Zero	X'7FFF'

9 If the UCB indicates that the device will accept the Select Write, code X'4B', it is used until a command reject is received. Then Select, code X'0B', is used.

10 The first Write CCW is pointed to by LDNACREQ.

11 The first Read Modified CCW is pointed to by LDNACREQ.

12 The number of bytes transmitted in the previous Read Modified CCW is stored in the LDNRMLEN field in the LDNCB. For this Read Modified, enough buffers are allocated to hold LDNRMLEN bytes of data. If this is not enough buffers for the device to send all of the pending data, the channel program will end on the Read Skip CCW. The Read Skip CCW reads all of the pending data but does not transmit any of it. The new length is stored in LDNRMLEN. (The length is computed by subtracting the residual byte count in the CSW from X'7FFF'.) Then the channel program is executed again with one more buffer than necessary allocated to read and transmit all of the pending data.

Read Buffer Channel Program

Table 59. Read Buffer Channel Program

CCW	Command Code (hex)	Flags	Address	Byte Count
Select	0B or 1B[13]	CC,SLI	Zero	1
TIC	08	—	Read buffer CCW	—
Read Buffer[14] (As many as needed)	02	CD	Input Area	Length of data
TIC	08	—	Read buffer CCW	—
Read Buffer[15]	02	CD	Input Area	Length of data
Read Buffer (Skip)	02	SKIP	Zero	X'7FFF'

Erase/Write Channel Program

Table 60. Erase/Write Channel Program

CCW	Command Code (hex)	Flags	Address	Byte Count
Erase/ Write[16] (As many as needed)	05	CD,SLI	Output Area	Length of data
TIC	08	—	Erase/Write CCW	—
Erase/ Write	05	SLI	Output Area	Length of data

[13] If the UCB indicates that the device will accept the Select Read, code X'1B', it is used until a command reject is received. Then Select, code X'0B', is used.

[14] The first Read CCW is pointed to by LDNACREQ.

[15] The number of bytes transmitted in the previous Read buffer CCW is stored in the LDNRBLEN field in the LDNCB. For this Read buffer, enough buffers are allocated to hold LDNRBLEN bytes of data. If this is not enough buffers for the device to send all of the pending data, the channel program will end on the Read Skip CCW. The Read Skip CCW reads all of the pending data but does not transmit any of it. The new length is stored in LDNRBLEN. (The length is computed by subtracting the residual byte count in the CSW from X'7FFF'.) Then the channel program is executed again with enough buffers allocated to read and transmit all of the pending data.

[16] LDNACREQ has the address of the first Erase/Write CCW.

Erase/Write Alternate Channel Program

Table 61. Erase/Write Alternate Channel Program

CCW	Command Code (hex)	Flags	Address	Byte Count
Erase/ Write[17] Alternate (As many as needed)	0D	CD,SLI	Output Area	Length of data
TIC	08	—	Erase/Write Alternate CCW	—
Erase/ Write Alternate	0D	SLI	Output Area	Length of data

Erase All Unprotected Channel Program

Table 62. Erase All Unprotected Channel Program

CCW	Command Code (hex)	Flags	Address	Byte Count
Erase All[18] Unprotected	0F	SLI	Zero	1

[17] LDNACREQ contains the address of the first Erase/Write Alternate CCW.

[18] LDNACREQ contains the address of the Erase All Unprotected CCW.

Channel Programs for a BSC Communication Adapter Line (VM) (VSE)

The BSC line control block (BSCLB) represents a BSC line attached through a BSC communication adapter and contains addresses and CCWs needed for channel programs. See Table 80 on page 1165 to determine what books contain detailed descriptions of channel programs, the interface between VTAM and the telecommunications subsystem controller or communication adapter, and the sense codes returned from a telecommunications subsystem controller or communication adapter.

Channel Commands

The following channel commands are used by VTAM to write data to or read data from a BSC terminal attached through a communication adapter line. Of the channel commands supported by the BSC communication adapter, VTAM only uses those required for master-station multipoint operations.

Command Code (hex)	Command	Description
01	Write	Transfers data from storage in the host processor to the BSC device.
02	Read	Transfers data from the control unit to storage in the host processor.
03	No-Op	Used to stop channel program execution (for example, General Poll). It also causes the channel to respond with a channel end and device end.
04	Sense	Transfers two bytes of sense data to VTAM.
08	TIC (Transfer in Channel)	Requests a transfer in the channel program. It causes the channel program to execute a command other than the next sequential command in the channel program.
09	Poll	Starts the autopoll of the poll list at the entry specified.
23	Set Mode	Used to condition a line in an Enable WRITE channel program. It sets the desired operating environment.
27	Enable	Sets the BSC line to the enabled state. It is the last command in an Enable WRITE channel program.
2F	Disable	Sets the line to the disabled state. It is a part of the Enable Write and Disable channel programs.

I/O Scheduling Overview of BSC Lines

During activation of a BSC line, a polling list is constructed containing entries for all of the BSC cluster controllers that are defined for the line. Initially, all the poll list entries are inactive and no I/O is started.

At the time a Contact RU is processed for the cluster controller, activation processing activates the appropriate polling list entry and, if it is the first cluster on the line to be activated, causes the General Poll channel program to be started. The General Poll channel program is designed to run continuously, without interruption, as long as negative responses to poll (EOTs) are being received. When an inter-

ruption occurs, VTAM analyzes the non-negative response to determine the next action.

Normally, it would be a Continue READ channel program to acknowledge the previous response and to solicit the next transmission. The Continue READ channel program is issued repeatedly until an EOT is received that indicates that the cluster controller has completed its data transmission.

Upon receipt of an EOT, the station pending traffic queues are checked for pending output to any active terminal on the line. If any is found, that transmission is scheduled prior to restarting the General Poll. Any pending activation or deactivation of polling list entries is done at this time as well.

If output I/O, activation, or deactivation occurs while the General Poll is active, but prior to an interruption, the General Poll is stopped to allow this processing to proceed. This is accomplished in a controlled manner by changing the TIC (the sixth command) in the General Poll channel programs to a No-op. Changing the TIC to No-op causes the channel program to end when it has polled all entries in the polling list.

Following completion of any scheduled operations, the General Poll is restarted at the first entry in the polling list. If General Poll is restarted after the completion of a General Poll, it is restarted at the next entry from the one that caused the General Poll to be completed. This ensures an even distribution of poll operations among the cluster controllers on a line.

Channel Programs

The following channel programs are used in the I/O sequence that sends data to or receives data from a BSC terminal.

Note: Except for the Enable Write and Disable WRITE channel programs, all of the channel programs (or portions of them) are in the BSC line control block (BSCLB).

Enable WRITE Channel Program

This channel program is started for ACTLINK processing in order to condition and enable the line.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
(1)	Disable	2F	CC,SLI	—	1
(2)	Set Mode	23	CC,SLI	Control Information	1
(3)	Enable	27	SLI	—	1

Disable WRITE Channel Program

This program is started for DACTLINK processing in order to disable a line.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
(1)	Disable	2F	SLI	—	1

General Poll (Initial Read) Channel Program

To solicit data from BSC terminals, this program is constructed to allow polling to begin with any entry in the poll list and to continue wrapping through the list until:

- A non-negative response is received from a terminal.
- The channel program is stopped for the scheduling of output.
- The operations terminate.

The program is also updated dynamically to reflect a changed poll list entry, poll list length, or read data area. When this program is initiated, the BSCOCFSM, BSCPCFSM, and BSCCCFSM fields of the BSCLB are set to X'01'.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
(1)	Write	01	CC,SLI	EOT	1
(2)	Poll	09	CC,SLI	Poll list entry	n[19] + 1
(3)	TIC	08	—	Command (5)	1
(4)	TIC	08	—	Command (7)	1
(5)	Poll	09	CC,SLI	First entry poll list	m[20] + 1
(6)	TIC	08	—	Command (5)	1
(7)	Read	02	SLI	I/O buffer	Length of data

Note: The numbers in the Address column refer to the numbers in the Sequence column.

[19] n is the number of stations left after the last one to respond, times the length of the BSC polling list entries.

[20] m is the number of stations on the polling list, times the length of the BSC polling list entries.

Continue READ Channel Program

This program is used after either an Initial READ channel program or another Continue READ channel program. It sends a positive acknowledgment to previously received data and reads in additional data. When this program is executing, the BSCOCFSM, BSCCPFSM, and BSCCCFSM fields of the BSCLB are set to X'02'.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
(1)	Write	01	CC,SLI	ACK0 or ACK1	2
(2)	Read	02	SLI	I/O buffer	Length of data

Repeat READ Channel Program

During error recovery procedure (ERP) for BSC lines, this channel program is executed to request that the terminal retransmit the previous text block in which a "block check" error was detected. At this point, the BSCOCFSM, BSCPCFSM, and BSCCCFSM fields of this BSCLB are set to X'0B'.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
(1)	Write	01	CC,SLI	NAK	1
(2)	Read	02	SLI	I/O Buffer	Length of data

Interrupt WRITE Channel Program

When the inbound data flow must be stopped immediately, the Interrupt WRITE channel program is used to stop the General Poll (Initial Read) channel program. In order to prevent data loss, the channel program is executed following a text block that ends with ETX. The BSCCCFSM and BSCPCFSM fields of the BSCLB are set to X'0C'.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
(1)	Write	01	CC,SLI	RVI	2
(2)	Read	02	SLI	Response area	2

Reset WRITE Channel Program

The Reset WRITE channel program is used to write an EOT to the terminal and to return the line to control mode. When this program is executing, the BSCCCFSM and BSCPCFSM fields of the BSCLB are set to X'07'.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
(1)	Write	01	SLI	EOT	1

Status READ Channel Program

The Status READ channel program is executed when a terminal indicates that it has pending BSC sense/status information. The program specifically polls the terminal and reads the BSC sense/status information only. The BSCCCFSM and BSCPCFSM fields of the related BSCLB are set to X'0A'.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
(1)	Write	01	CC,SLI	EOT	1
(2)	Write	01	CC,SLI	Specific poll characters	5
(3)	Read	02	SLI	I/O buffer	Length of data

Inquiry READ Channel Program

The Inquiry READ channel program clears a communication adapter line buffer. Any data read is discarded because of the skip flag in the program's Read CCW. The BSCCCFSM and BSCPCFSM fields of the BSCLB are set to X'0D'.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
(1)	Read	02	SLI,SKIP	Response area	4095

Sense Channel Program

The Sense channel program is issued when VTAM receives unit-check status from the communication adapter.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
(1)	Sense	04	SLI	Sense area	2

Initial Write (Conversational) Channel Program

A variation of the Initial WRITE channel program, the Initial Write (Conversational) channel program performs a required conversational function. This function allows a terminal to respond to a text block with another text block, rather than with a response. The channel program must have an appropriate length buffer ready to receive the text block. The BSCOCFSM, BSCCCFSM, and BSCPCFSM fields of the BSCLB are set to X'08' when an Initial Write (Conversational) channel program is executing. They are set to X'04' when an Initial WRITE channel program is executing.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
(1)	Write	01	CC,SLI	EOT	1
(2)	Write	01	CC,SLI	Selecting characters	5
(3)	Read	02	SLI	Response area	2
(4a)	Write	01	CD,SLI	STX ESC characters	2
(4b)	TIC	08	—	Command (4c) in I/O buffers	1
(4c)	Write	01	CD,SLI	Data in I/O buffer	Length of data
(4d)	TIC	08	—	Command (4e)	1
(4e)	Write	01	CC,SLI	ETX character	1
(5a)	Read	02	SLI	Response area	2
(5b)	Read	02	SLI	I/O buffers	Length of data

Inquiry Write (Conversational) Channel Program

The Inquiry Write (Conversational) channel program is issued during ERP (error recovery procedure) in order to request the terminal to retransmit its last response. The BSCCCFSM and BSCPCFSM fields of the BSCLB are set to X'06'.

Sequence	CCW	Command Code (hex)	Flags	Address	Byte Count
(1)	Write	01	CC,SLI	ENQ	1
(2a)	Read	02	SLI	Response area	2
(2b)	Read	02	SLI	I/O buffer	Length of data

Channel Program Finite State Machines for BSC Lines

The finite state machine (FSM) represents what is happening on the BSC line. Table 63 shows the finite state machines and what they mean.

Table 63. Channel Program Finite State Machines for BSC Lines

FSM	State	Explanation
(1) GP	General Poll	The General Poll channel program is executing.
(2) CR	Continue Read	The Continue READ channel program is executing.
(3) CM	Control Mode	The BSC line is in control mode. No channel program is executing for the line.
(4) IW1	Initial Write 1	The first part of an Initial WRITE channel program is executing.
(5) IW2	Initial Write 2	The second part of an Initial WRITE channel program is executing.
(6) INQ	Inquiry Write	An Inquiry WRITE channel program is executing.
(7) EOT	End of Transmission	An EOT has been sent, or an EOT is expected.
(8) IC1	Initial Write Conversational 1	The first part of an Initial Write (Conversational) channel program is executing.
(9) IC2	Initial Write Conversational 2	The second part of an Initial Write (Conversational) channel program is executing.
(10) STR	Status Read	A Status READ channel program is executing.
(11) RR	Repeat Read	A Repeat READ channel program is executing.
(12) IWT	Interrupt Write	An Interrupt WRITE channel program is executing.
(13) IQR	Inquiry Read	An Inquiry READ channel program is executing.
(14) IQW	Inquiry Write Conversational	An Inquiry Write (Conversational) channel program is executing.

Channel Programs for an SDLC Communication Adapter Line (VM) (VSE)

The HALCB represents an SDLC line attached through an SDLC communication adapter and contains addresses and CCWs needed for channel programs. See Table 80 on page 1165 to determine what books contain detailed descriptions of channel programs, the interface between VTAM and the telecommunications subsystem controller or communication adapter, and the sense codes returned from a telecommunications subsystem controller or communication adapter.

Channel Commands

The following channel commands are used in the channel programs to communicate with SNA devices on SDLC lines or with the SDLC lines themselves.

Command Code (hex)	Command	Description
01	Write	Writes an SDLC command located in the TSCODATA field of the TSCB.
02	Read	Reads an SDLC command or control data located at TSCIDATA.
03	No-op	Used to stop channel program execution.
05	Write PIU	Points to TSCILNG, a two-byte prefix to TSCPIU. TSCILNG contains the total length of the data in the PIU. This length includes the TH, RH, RU, and TSCILNG. Data in any continuation buffers for the Write PIU channel program will start at TSCPIU as usual.
06	Read PIU	Points to TSCILNG, a two-byte prefix of TSCPIU. The total length of the data is returned in TSCILNG. Data in any continuation buffers of the Read PIU channel program begins at TSCILNG.
08	TIC (Transfer in Channel)	Requests a transfer in the channel program. It causes the channel program to execute a command other than the next sequential command in the channel program.
09	Autopoll	Points to a polling list of SCBs. The length is the length of the polling list. This command solicits input from active stations, contact polls those stations being activated, and ignores the inactive stations.
0B	Control SCB (Control SPL)	Transfers 12 bytes of SCB information from VTAM to the communication adapter.
0F	Poll	Transfers a final RR or RNR after a chain of Write P commands. When the Poll command is executed, the line is turned to inbound.
14	Sense SCB (Sense SPL)	Retrieves the current SCB from the communication adapter after a line operation completes. The current SCB indicates which station responded.

Command Code (hex)	Command	Description
23	Set Mode	Transfers up to 8 bytes of setmode data to the communication adapter to change the status of a line.
27	Enable	Makes the line operational.
29	Dial	Transfers dialing data to the automatic calling unit (ACU).
2F	Disable	Makes the line non-operational.
E4	Sense I/O	Transfers up to 7 bytes of line configuration information from the communication adapter.

I/O Scheduling Overview of SDLC Lines

During activation of an SDLC line, a station polling list (SPL) is constructed containing entries for all the physical units that are defined for the line. Initially, all the station entries are inactive. When any station has to be contacted and it is the first station on the line to be activated, the Autopoll channel program is started to do a Contact Poll. When a station sends in a nonsequenced acknowledgment (NSA), data polling begins.

Depending on the coding of the XID operand in the definition statement for the station, VTAM will contact poll with a null XID or a SNRM. If the contact poll is a null XID, XID exchanges will occur to provide VTAM with additional information about the station. When the XID exchange is complete, VTAM will contact poll with a SNRM.

The Autopoll channel program is designed to run continuously as long as Receive Ready (RR) or Receive Not Ready (RNR) responses flow. The Autopoll channel program is interrupted when SDLC supervisor frames or information frames are received, when data is to be transmitted, or when certain PU services I/O has to be processed. In order to send the Write data, the Autopoll channel program is interrupted by changing the third CCW, which is a No-op, to a TIC CCW pointing to the Write PIU channel program.

The Write buffers to be transmitted are pointed to by a pending queue (PUTPENDQ), or an error hold queue (PUTEHOLD) in the PUT entry. The SDLC station scheduler scans every PUT entry, starting from the current entry indicated in HALCUSCB, and passes the Write buffers to TSC's SDLC channel program build routine.

If there is no Write data to be sent to any station, the Autopoll is restarted. The communication adapter begins polling from the next station in the station polling list unless the polling index is changed by the Set Mode CCW. One pass of the Autopoll channel program is restarted if a Read Error CCW occurs on one of the stations. This is accomplished by changing the TIC CCW after the Autopoll CCW to a No-op.

I/O Scheduling for PU Services

All PU services requests are processed by the SDLC connection-manager outbound on the PU services PAB in the HALCB (NCBPUPAB). Before initiating I/O for PU services, a switch is set to wait for current I/O (including the Autopoll channel program) to finish. This is done by setting the state of PU services I/O (HALPUSIO) to pending, which signals the channel end not to restart application program I/O or Autopoll after the completion of the current I/O. The continuous Autopoll channel program is broken by changing the third CCW, which is a No-op, to a TIC CCW pointing to another No-op.

When the current I/O has been completed, the state of PU services I/O is set to "halted." When the connection manager processes the request and schedules I/O, the state of PU services I/O changes to "active." The state changes to "inactive" after the completion of I/O.

The connection manager builds its channel program in a buffer (at TSCPCCW).

I/O Scheduling for the Link Level 2 Test

A TESTMODE request is initiated and completed by the SDLC connection-manager outbound routine on the PU services PAB in the HALCB (NCBPUPAB). The link level 2 test is executed by the channel end appendage and the link scheduler.

If the TESTMODE request is to start a link level 2 test, the SDLC connection-manager outbound routine sets the station to "test mode," initializes fields in the PUT entry, obtains storage for the write and Read buffers, moves the specified data into the Write buffer, and starts the Write Test channel program. The Write Test channel program is repeated the specified number of times, or until the test is canceled by the operator.

If the TESTMODE request is to cancel a current test, a RECTR request is built and sent to network operator services so that the operator message with the test results can be built.

Channel Programs

The following channel programs are used in I/O sequences that send data to or receive data from devices on nonswitched and switched SDLC lines.

Note: The Autopoll channel program (a part of the Write PIU channel program) and the Sense SCB channel program are in the HALCB (HALCHPGM). The other channel programs are formatted in I/O buffers.

Autopoll Channel Program

The Autopoll channel program is used to solicit data from active stations (data poll) and to check whether inactive stations are ready to be activated (contact poll). When the Autopoll CCW is executed, the communication adapter steps through the SDLC polling list table (SPL table; SCB table to the communication adapter) and sends either a data poll or a contact poll until one of the following occurs:

- An active station responds with data.
- An inactive station responds with a nonsequenced acknowledgment (NSA).
- An error occurs.

- VTAM changes the TIC address to point to a Write PIU channel program or to a No-op CCW to stop the Autopoll channel program.

If all responses to the polls are negative, the final TIC chains to the second CCW (the No-op) to start another polling pass through the SPL table.

CCW	Command Code (hex)	Flags	Description	Remarks
Set Mode	23	CC,SLI	Sets the desired operating mode.	Optional
No-Op	03	CC,SLI	Required as the last in a chain.	
TIC	08	—	When polling, transfers control to the Autopoll CCW immediately following the TIC. For a Write channel program, either transfers to a string of Write PIUs or to a No-Op CCW to stop Autopoll. (See the Write PIU channel program.)	
Autopoll	09	CC,SLI	Starts polling, points to the SPL table, sends Receive Ready (RR) for data poll or Set Normal Response Mode (SNRM) for contact poll depending upon the flag settings in the SPL.	
TIC	08	—	Transfers back to the second command (No-Op) to continue polling, is changed to No-Op if continuous Autopoll is not required (for example, when an error occurs on a Read CCW), and transfers to a Write Test channel program when LL2 test is required.	
TIC	08	—	Is executed when status modified is given to the channel by the communication adapter and transfers to Read PIU or Sense SCB channel program.	

Read PIU Channel Program

The Read PIU channel program reads data sent to VTAM by another SDLC station. Normally, the channel program is started when an Autopoll TIC points to the first Read PIU buffer. The Read PIU CCWs are built in the buffers that will receive the data and are chained together by TICs. The number of buffers in the chain is determined by the values for the INBFRS operand of the line's definition statement.

When VTAM is a secondary link station or when the line is in an inbound direction, the communication adapter starts the Read PIU channel program with the Control SCB CCW that is located in the HALCB.

When the transmission is complete or when the communication adapter detects error information, the communication adapter skips over any remaining Read PIU CCWs to a Sense CCW by which it can report error information.

The Read PIU channel program is pointed to by the Autopoll channel program unless it starts with a Control SCB CCW.

CCW	Command Code (hex)	Flags	Description	Remarks
Control SCB	0B	CC,SLI	Used only when VTAM is a secondary link station, the line is in an inbound direction, and the Autopoll channel program is not used to point to the first Read CCW in the HALCB.	Optional
TIC	08	—	Points to the first Read PIU in the HALCB.	Optional
Read PIU	06	CC,SLI	Points to the input area.	
TIC	08	—	Points to the next Read PIU CCW.	
Read PIU	06	CC,SLI	Points to the input area.	
TIC	08	—	Points to the Sense SCB CCW that is in the HALCB.	

Write PIU Channel Program

The Write PIU channel program sends data to another SDLC station. Normally, the channel program is started when an Autopoll TIC points to the first Write PIU buffer. The Write PIU CCWs are built in the buffers that contain the outbound data. After the data in the last buffer is sent to the station, the program executes a Poll CCW that sends RR (Receive Ready) so that the station may acknowledge receipt of the data and, optionally, may send any waiting data. The final TIC points to the Read PIU channel program in order to receive possible inbound data and to verify that the outbound data was received.

The Write PIU channel program is pointed to by the Autopoll channel program in the HALCB.

CCW	Command Code (hex)	Flags	Description
Control SCB	0B	CC,SLI	Points to an SPL entry.
TIC	08	—	Points to the first Write PIU in the HALCB.
Write PIU	05	CC	Points to the buffer containing the PIU to be sent. Even when the data spans more than one buffer, the CCWs are command chained.
TIC	08	—	Points to a Write PIU CCW if there are more PIUs, or points to a Poll CCW if the last PIU has been sent.
Poll	0F	CC,SLI	Starts polling at the next entry in the SPL.
TIC	08	—	Points to a Read PIU or Sense SCB CCW.

Write Test Channel Program

The Write Test channel program sends and receives test frames. It is used when a link level 2 test is requested. The Write Test channel program is pointed to by the Autopoll channel program or by NCBCAW in the HALCB.

CCW	Command Code (hex)	Flags	Description
Write	01	CC,SLI	Points to data in the Write buffer.
Read	02	SLI	Points to the Read buffer.

Sense SCB Channel Program

The Sense SCB channel program is used to retrieve completion status from the communication adapter. The Sense SCB CCW is the last CCW in the chain for channel programs that perform normal data transfer for the Autopoll channel program, the Read PIU channel program, or the Write SDLC Control channel program.

CCW	Command Code (hex)	Flags	Description
Sense SCB	14	SLI	Points to an input area (HALCUSCB).

Sense I/O Channel Program

The Sense I/O channel program requests and receives line configuration information from the communication adapter. This information is used to verify information in the line's definition statement. The verified information is included in the outbound record for the line. For this channel program, the HALLFSM field of the HALCB is set to X'01'.

CCW	Command Code (hex)	Flags	Description
Sense I/O	E4	—	Points to an input area used to obtain the line configuration.

Enable Line Channel Program

The Enable Line channel program is used when a nonswitched line is activated. When this channel program is being executed, the HALLFSM field in the HALCB is set to hex 02.

CCW	Command Code (hex)	Flags	Description
Disable	2F	CC,SLI	Sets the line to disable state and clears certain pending status conditions.
Set Mode	23	CC,SLI	Tells the communication adapter how VTAM expects to use the line and sets operating values such as timeout values and control indexes. Desired operating mode values are contained in the HALSMODE field in the HALCB.
Enable	27	SLI	Causes the communication adapter to make the DCE interface ready. For CCITT V.24 or EIA RS-232C, Enable means activating the Data Terminal Ready (DTR) line and verifying that the response is Data Set Ready (DSR).

Answer Channel Program

The Answer channel program is used when a switched line is activated. When a call is received, VTAM checks information in the XID response against values in the corresponding switched line's definition statement. Based on that information, VTAM either accepts or rejects the call. For this channel program, the HALLFSM field in the HALCB is set to X'01'.

CCW	Command Code (hex)	Flags	Description
Disable	2F	CC,SLI	Sets the line to disable state, clears certain pending status conditions, and breaks any outstanding switched connections.
Set Mode	23	CC,SLI	Tells the communication adapter how VTAM expects to use the line and sets operating values such as control indices. The desired operating mode is contained in the HALSMODE field in the HALCB.
Enable	27	CC,SLI	Causes the communication adapter to make the DCE interface ready. This CCW is not completed until an incoming call is accepted, so it may be active for a long time before it completes with the Data Terminal Ready (DTR) sequence.
Write	01	CC,SLI	Points to the area containing all station identifiers and XID commands (TSCODATA).
Read	02	SLI	Points to an input area (TSCIDATA).

Dial Channel Program

The Dial channel program is identical to the Answer channel program except that a Dial CCW is used to initiate an outbound call (an Enable CCW is used to receive an inbound call). Execution of the Dial CCW transfers the desired telephone number to the communication adapter which forwards the number, one digit at a time, to an autocall adapter. The HALLFSM field in the HALCB is set to hex 02 when this program is active.

VSE For X.21 switched lines, the Listen CCW follows the Dial CCW and is used by the communication adapter to pass call progress signals (CPS) or other DCE-provided information to VTAM. VTAM supports only CPS and ignores any other DCE-provided information received with the Listen CCW.

CCW	Command Code (hex)	Flags	Description
Disable	2F	CC,SLI	Sets the line to disabled state.
Set Mode	23	CC,SLI	Sets the desired operating mode as specified in HALSMODE.
Dial	29	CC,SLI	Points to an area (TSCPDATA) containing the digits to be dialed.
Listen (X.21 only) VSE	0A	CC,SLI	Points to an area to receive CPS or DCE-provided information.
Write	01	CC,SLI	Points to the area containing all station identifiers and XID commands (TSCODATA).
Read	02	SLI	Points to an input area (TSCIDATA).

Write SDLC Control Channel Program

The Write SDLC Control channel program is used when VTAM is the secondary station (for example, in a VTAM-NCP connection). This channel program sends a nonsequenced acknowledgment (NSA) in response to a Set Normal Response Mode (SNRM) from a Contact Poll or in response to a Discontact (DISC) sent during shutdown processing.

CCW	Command Code (hex)	Flags	Description
Set Mode	23	CC,SLI	Sets the desired operating mode.
Write	01	CC,SLI	Points to an output area (TSCODATA) and will be changed to a No-Op CCW when closedown recording is required for the physical unit.
TIC	08	—	Transfers to the Sense SCB CCW that is in the HALCB, causing the communication adapter to wait for some transmission from the primary station.

Read SDLC Control Channel Program

Like the Write SDLC Control channel program, the Read SDLC Control channel program is used when VTAM is the secondary station. It waits for the Set Normal Response Mode (SNRM) sent by the primary station on a Contact Poll.

The Read SDLC control program can be pointed to by the Enable Line channel program.

CCW	Command Code (hex)	Flags	Description
Set Mode	23	CC,SLI	Sets the desired operating mode in the HALSMODE field in the HALCB. It can be a Write CCW or No-Op CCW, if required.
Read	02	—	Reads an SDLC command.

Disable Line Channel Program

The Disable Line channel program sends a Disable CCW to shutdown the modem interface of the communication adapter.

CCW	Command Code (hex)	Flags	Description
Disable	2F	SLI	Sets the line to disabled state.

Primary XID Channel Program

The Primary XID channel program is for a communication-adapter-attached type 5 physical unit (XID exchange).

CCW	Command Code (hex)	Flags	Description	Contents of Address Field
Set Mode	23	CC,SLI	Changes the timeout value.	HALSMODE
Write	01	CC,SLI	Transfers the XID(00) from this host to the communication adapter attached host.	Output Area
Read	02	SLI	Reads an SDLC command.	Input Area

Secondary XID Channel Program

The Secondary XID channel program is for a communication-adapter-attached type 4 physical unit (XID exchange).

CCW	Command Code (hex)	Flags	Description	Contents of Address Field
Disable for non-switched lines only	2F	CC,SLI	Makes sure that the line direction is outbound.	—
Set Mode	23	CC,SLI	Changes the timeout value.	HALSMODE
Enable	27	CC,SLI	Causes the communication adapter to make the DCE interface ready. For CCITT v.24 or EIA RS-232C, Enable means activating the Data Terminal Ready (DTR) line and verifying that the response is Data Set Ready (DSR).	
Read	02	SLI	Reads an SDLC command.	Input Area

Link, Connection, and Physical Unit Finite State Machines (MVS) (VM)

If an error occurs during the activation or deactivation of the SDLC link or of a physical unit, the finite state machines can be used to determine the state of the link or station.

Link State Machine Values

00 Reset
01 Sense I/O pending
02 Enable pending
08 Active
10 Halt I/O pending
20 DACTLINK disable pending
30 INOP disable pending
40 ACTLINK disable pending

Connection State Machine Values

00 Reset
01 Answer pending
02 Auto dial pending
03 X.21 answer pending VSE
04 Manual dial pending
05 X.21 autocall pending VSE
06 X.21 SHM HALT I/O pending 1 VSE
07 X.21 SHM HALT I/O pending 2 VSE
08 Connected
10 Abandon connection 1
11 Abandon connection 2
12 X.21 SHM autocall pending 1 VSE
13 X.21 SHM autocall pending 2 VSE
14 Pre-negotiation XID pending
15 Send CV22 pending
21 DACTCONNIN pending

41 Abandon connect out pending
51 X.21 OBR pending VSE
60 Answer pending for sub-area dial start
61 Answer pending for sub-area dial finish
65 Autodial pending for sub-area dial links
69 Disconnect/re-enable answer state

Physical Unit State Machine Values

00 Reset
01 Contact pending
03 Pending Discontact pending
04 XID pending
08 XID(00) pending
09 Negotiation preceding as primary
10 Active
11 Test mode
12 Link role negotiation proceeding
13 Primary non-activation XID pending
14 Primary non-activation Contact pending
15 Primary non-activation CONTACTED (0B) pending
21 Request Discontact (RQD) received
22 Discontact pending
23 Pending reset
30 Negotiation proceeding as secondary
31 Secondary Contact pending
32 Secondary Set Normal Response Mode (SNRM) received
39 Primary/secondary contention
40 Secondary active
41 Secondary non-activation contact pending
42 Secondary non-activation CONTACTED (0B) pending
51 Secondary Discontact pending
52 XID error proceeding
56 OBR pending INOP
58 OBR pending Contacted
60 OBR pending Discontact

Hints for Common Errors: Finite States (VM) (VSE)

If an error occurs, the finite state in the PUT (PUTFSM) and in the HALCB (HALFSM) should be checked to find the state at which the error occurred. The finite state machine diagrams below show the actions occurring as a result of various inputs for the finite state machine values.

The SCX in HALCB (HALCUSCB) contains the status for the last physical unit for which I/O was active.

The sense field in HALCB (NCBSENSE) contains the sense data as a result of a unit check on the line.

The state of PU services I/O (HALPUSIO) indicates whether or not the connection manager was doing I/O.

The NCBPOLL flag, if on, indicates that the line is polling. If off, it indicates that either Write I/O or connection manager I/O is in progress.

The NCBREADY flag, if on, indicates that the link is active.

NCBLINOP indicates a permanent link error.

NCBSINOP indicates a permanent station error.

Appendix E. Network Flows

This appendix describes flows of the VTAM program RUs and AMRUs between network addressable units in single and multiple VTAM networks. Use these flows as guidelines to help analyze and isolate network problems caused by unexpected network events, such as protocol violations. The flow diagrams are divided into the following categories:

- "Generic BIND (GBIND) AMRUs" on page 930
- "Resource Activation Flows" on page 933
- "Session Establishment Flows" on page 949
- "Deactivation and Session Termination Flows" on page 976
- "Error Detection and Recovery and SSCP Management Services" on page 1000.
- "Dial and Answer for X.21 Switched Lines (VSE)" on page 1011.

For certain session establishment RUs (ACTCDRM, ACTPU, ACTLU, and BIND), additional RUs can flow if the explicit route (ER) or virtual route (VR) selected for a session is not yet active. Because this flow is essentially the same for all four RUs, these RUs are referred to as generic bind (GBIND) AMRUs. To avoid repetition, the flows for these AMRUs are shown once at the beginning of this appendix.

Table 64 lists all the network flows illustrated here.

Table 64 (Page 1 of 4). Index of Network Flows

GBIND AMRU Flow	Page
ACTLU: Sending an ACTLU request for a logical unit (LU)	Figure 98 on page 931
ACTPU: Sending an ACTPU request for a comunication controller or physical unit (PU)	Figure 99 on page 931
BIND: Sending a BIND request to a secondary logical unit (SLU)	Figure 100 on page 932
Virtual and explicit route: Activating a virtual route (VR) and the associated explicit route (ER)	Figure 101 on page 932
Resource Activation Flow	**Page**
Activating a CDRM	
CDRM with COLD response, activating	Figure 116 on page 944
CDRM with ERP response, activating	Figure 115 on page 944
CDRM with a virtual route-based transmission group, activating	Figure 117 on page 945
Activating a cross-network SSCP-SSCP session	
Back-to-back gateway NCPs request sessions	Figure 118 on page 946
Gateway VTAM requests session	Figure 119 on page 947
Non-gateway VTAM requests session	Figure 120 on page 948
Activating an NCP major node	
Channel-attached communication controller, activating	Figure 102 on page 934
Link-attached communication controller, activating	Figure 103 on page 935
Activating resources controlled by a host or NCP major node	
Link: Activating a link	Figure 104 on page 935
Link station: Activating a cross-subarea link station	Figure 106 on page 936
Logical unit (LU): Activating a logical unit	Figure 113 on page 942
Application program: Activating an application program and processing an OPEN ACB request	Figure 114 on page 943
Physical unit (PU): Activating a physical unit type 2.0	Figure 108 on page 938
Physical unit (PU): Activating a physical unit type 2.0 with load required	Figure 109 on page 939
Physical unit (PU): Moving a dynamically added physical unit	Figure 111 on page 941
Physical unit (PU): Moving a SYSGENed physical unit	Figure 110 on page 940
SSCP takeover of peripheral node logical units (LUs)	Figure 112 on page 942
Switched connection, establishing	Figure 107 on page 937
Switched link with takeover, activating	Figure 105 on page 936

Table 64 (Page 2 of 4). Index of Network Flows

Session Establishment Flow	Page
Failed session establishment	
Failure (CDINIT rejection) of session initiation by a secondary logical unit (SLU) for single gateway VTAM and single gateway NCP	Figure 145 on page 973
Failure (CINIT rejection) of setup procedure initiated by a secondary logical unit (SLU) for single gateway VTAM and single gateway NCP	Figure 147 on page 975
Failure (SETCV failure) of session initiation by a secondary logical unit (SLU) for single gateway VTAM and single gateway NCP	Figure 146 on page 974
Gateway VTAM	
Default partitioning of gateway VTAM responsibility spanning three networks	Figure 139 on page 966
Multiple gateway VTAMs and back-to-back gateway NCPs	Figure 122 on page 952
PLU availability for autologon, notification of	Figure 144 on page 972
Requests initiated by primary logical units (PLUs)	
Dependent PLU initiating cross-domain session with independent SLU	Figure 128 on page 957
Independent PLU initiating cross-domain session with independent SLU	Figure 127 on page 956
Independent PLU requesting session with independent SLU through a single gateway VTAM and single gateway NCP	Figure 130 on page 959
OPNDST ACQUIRE	Figure 123 on page 953
PLU initiating request for single gateway VTAM and single gateway NCP	Figure 129 on page 958
PLU initiating request setup queued for single gateway VTAM and single gateway NCP	Figure 131 on page 960
SIMLOGON	Figure 124 on page 953
SIMLOGON(RELREQ)	Figure 125 on page 954
SIMLOGON(RELREQ): Session is pending active or already in progress	Figure 126 on page 955
Requests initiated by secondary logical units (SLUs)	
Dependent SLU initiating cross-domain session with application logical unit (LU)	Figure 136 on page 963
INIT SELF	Figure 134 on page 962
LOGON	Figure 132 on page 961
Predesignated control of gateway NCP by middle host	Figure 139 on page 966
REQSESS	Figure 133 on page 961
Sending an unformatted request to the SSCP	Figure 135 on page 963
Single gateway connecting three or more networks	Figure 138 on page 965
Single gateway VTAM and single gateway NCP	Figure 137 on page 964
Requests initiated by third parties	
CLSDST PASS	Figure 140 on page 968
CLSDST PASS with NOTIFY	Figure 141 on page 969
Request spanning three networks	Figure 142 on page 970
VARY LOGON or LOGAPPL processing	Figure 143 on page 971

Table 64 (Page 3 of 4). Index of Network Flows

Deactivation or Session Termination Flow	Page
CLOSE ACB processing	Figure 163 on page 990
Deactivating an application program	Figure 164 on page 991
Deactivating a CDRM	
Forced	Figure 167 on page 993
Forced, without affecting active sessions	Figure 169 on page 995
Forced or immediate, VTAM releases prior to V3R4.1	Figure 170 on page 996
Immediate	Figure 166 on page 993
Immediate, without affecting active sessions	Figure 168 on page 994
Normal	Figure 165 on page 992
Deactivating a logical unit (LU), single network	
Forced	Figure 149 on page 978
Immediate	Figure 148 on page 978
VARY NET,TERM Cleanup	Figure 161 on page 988
VARY NET,TERM Unconditional	Figure 160 on page 987
With Giveback	Figure 150 on page 979
Deactivating a logical unit (LU), multiple networks	
Independent PLU sends BFCLEANUP for independent SLU	Figure 151 on page 980
Independent PLU sends UNBIND for independent SLU	Figure 152 on page 980
PLU sends UNBIND for multiple gateway VTAMs and single gateway NCP	Figure 153 on page 981
PLU sends UNBIND for single gateway VTAM and single gateway NCP	Figure 154 on page 981
SLU requests TERMINATE SELF (CLEANUP) for single gateway VTAM and single gateway NCP	Figure 156 on page 983
SLU requests TERMINATE SELF for multiple gateway VTAMs and back-to-back gateway NCPs	Figure 155 on page 982
SLU requests TERMINATE SELF for single gateway VTAM and single gateway NCP	Figure 157 on page 984
Type 2.1 nodes, active termination	Figure 158 on page 985
Deactivating a physical unit (PU) acting as an adjacent link station for independent logical unit (LU) sessions	Figure 159 on page 986
Queued session, terminating	Figure 162 on page 989
Route failure	
Route failure in intermediate network causes termination of LU-LU sessions	Figure 172 on page 998
Route failure in intermediate network causes termination of SSCP-SSCP sessions	Figure 173 on page 999
SSCP-SSCP session termination causes LU-LU sessions to be broken	Figure 171 on page 997

Table 64 (Page 4 of 4). Index of Network Flows

Error and SSCP Management Services Flow	Page
Error recovery processing (ERP)	
Hard INOP	Figure 175 on page 1002
Soft INOP	Figure 174 on page 1001
LPDA-2 processing	
Unsolicited LPDA-2 test on permanent link error with two link segments	Figure 179 on page 1005
Unsolicited LPDA-2 test on thresholds reached for an LPDA-2 physical unit (PU) with one link segment	Figure 177 on page 1003
Unsolicited LPDA-2 test on thresholds reached for an LPDA-2 physical unit (PU) with two link segments	Figure 178 on page 1004
SSCP management services processing	
FORWARD and DELIVER Routing	Figure 176 on page 1003
XRF processing	
Secondary logical unit (LU) initiate with USERVAR (LOGON)	Figure 182 on page 1009
Third-party initiate (CLSDST PASS)	Figure 183 on page 1010
XRF primary and backup sessions, establishment of	Figure 180 on page 1006
XRF session switch (takeover)	Figure 181 on page 1008
VSE X.21 Switched Lines Flow	**Page**
Dial and answer for X.21 switched lines	Figure 184 on page 1011

Many abbreviations are shown at the top of the network flows. The following list gives the meaning of some of those abbreviations:

APPL Application
BF Boundary function
BFSS Boundary function session services
BNN Boundary network node
CS Configuration services
EU End user
LU Logical unit
NCP Network Control Program
NOS Network operator services
PLU Primary logical unit
PN Peripheral node
PU Physical unit
PUNS Physical unit services
SLU Secondary logical unit
SS Session services
SSCP System services control point
TSC Transmission subsystem component
XRF Extended recovery facility

Generic BIND (GBIND) AMRUs

Access method RUs (AMRUs) are internal requests that might appear in the PIU trace and are a function of Physical Unit Services (PUNS), Configuration Services, or Session Services.

Figure 98 on page 931 through Figure 101 on page 932 show the flow of these requests and responses between the SSCP and logical and physical units when a virtual route (VR) or explicit route (ER) selected for a session is not yet active.

Index of Generic BIND (GBIND) AMRU Flows

Table 65 lists the GBIND AMRU flows illustrated here.

Table 65. Index of Generic BIND (GBIND) AMRU Flows

Flow	Page
ACTLU: Sending an ACTLU request for a logical unit (LU)	Figure 98 on page 931
ACTPU: Sending an ACTPU request for a communication controller or physical unit (PU)	Figure 99 on page 931
BIND: Sending a BIND request to a secondary logical unit (SLU)	Figure 100 on page 932
Virtual and explicit route: Activating a virtual route (VR) and the associated explicit route (ER)	Figure 101 on page 932

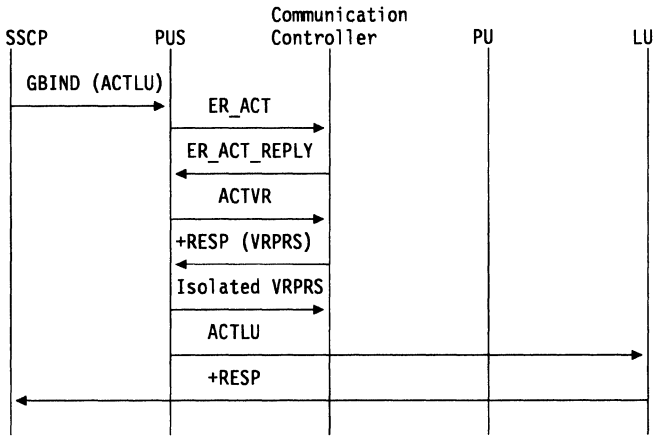


Figure 98. Sending an ACTLU Request for a Logical Unit (LU)

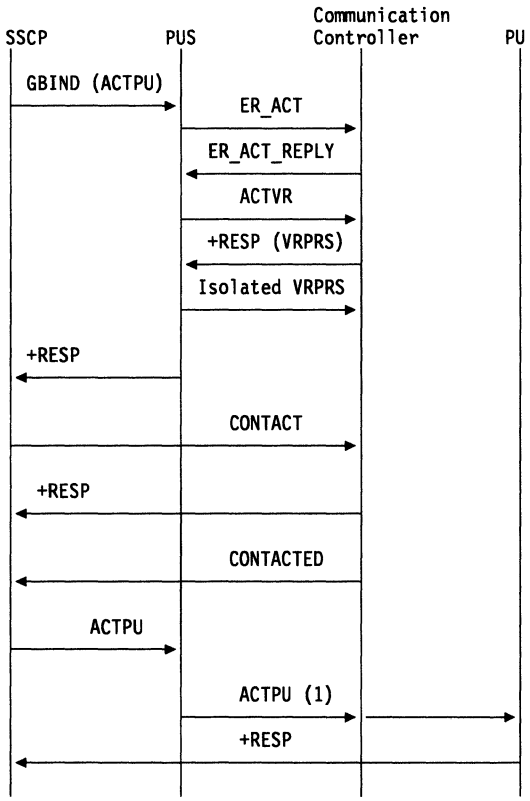


Figure 99. Sending an ACTPU Request for a Communication Controller or Physical Unit (PU)

1. The ACTPU can flow either to the communication controller or to a physical unit.

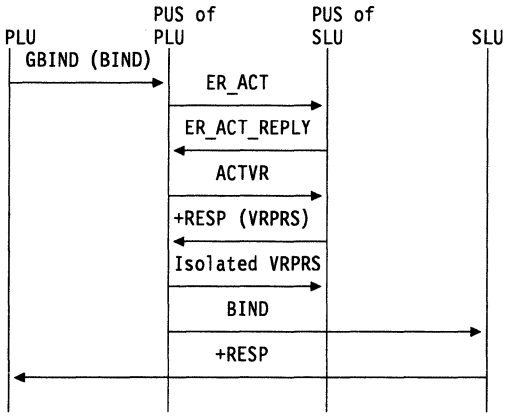


Figure 100. Sending a BIND Request to a Secondary Logical Unit (SLU)

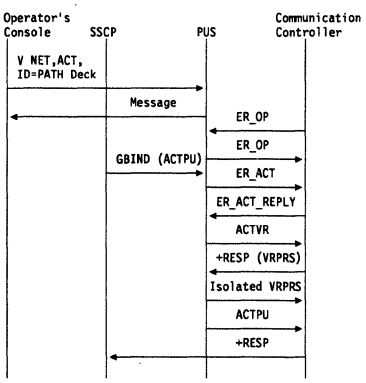


Figure 101. Activating a Virtual Route (VR) and the Associated Explicit Route (ER)

Resource Activation Flows

Figure 102 on page 934 through Figure 120 on page 948 show the flow of requests and responses between the SSCP and logical and physical units to activate resources.

For channel activation flows see Appendix D, "Channel Programs" on page 873.

Index of Resource Activation Flows

Table 66 lists the resource activation flows that are illustrated here.

Table 66. Index of Resource Activation Flows

Flow	Page
Activating a CDRM	
CDRM with COLD response, activating	Figure 116 on page 944
CDRM with ERP response, activating	Figure 115 on page 944
CDRM with a virtual route-based transmission group, activating	Figure 117 on page 945
Activating a cross-network SSCP-SSCP session	
Back-to-back gateway NCPs request sessions	Figure 118 on page 946
Gateway VTAM requests session	Figure 119 on page 947
Non-gateway VTAM requests session	Figure 120 on page 948
Activating an NCP major node	
Channel-attached communication controller, activating	Figure 102 on page 934
Link-attached communication controller, activating	Figure 103 on page 935
Activating resources controlled by a host or NCP major node	
Link: Activating a link	Figure 104 on page 935
Link station: Activating a cross-subarea link station	Figure 106 on page 936
Logical unit (LU): Activating a logical unit	Figure 113 on page 942
Application program: Activating an application program and processing an OPEN ACB request	Figure 114 on page 943
Physical unit (PU): Activating a physical unit type 2.0	Figure 108 on page 938
Physical unit (PU): Activating a physical unit type 2.0 with load required	Figure 109 on page 939
Physical unit (PU): Moving a dynamically added physical unit	Figure 111 on page 941
Physical unit (PU): Moving a SYSGENed physical unit	Figure 110 on page 940
SSCP takeover of peripheral node logical units (LUs)	Figure 112 on page 942
Switched connection, establishing	Figure 107 on page 937
Switched link with takeover, activating	Figure 105 on page 936

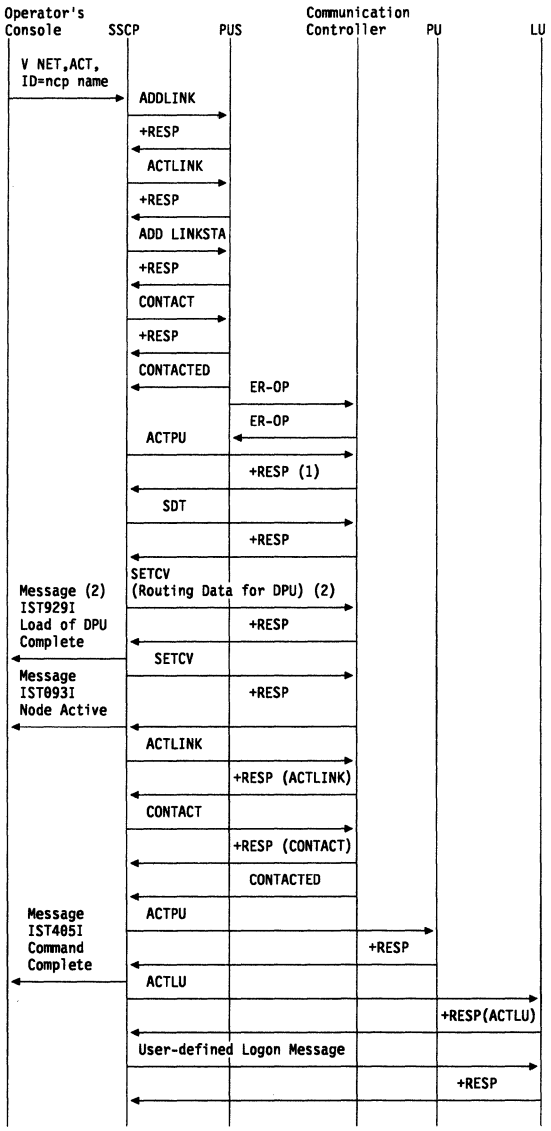


Figure 102. Activating a Channel-Attached Communication Controller

1. Includes NCP dynamic path definition capability indicator.
2. Flows only for dynamic path definition. SETCV and IST929I flow for each dynamic path definition member specified.

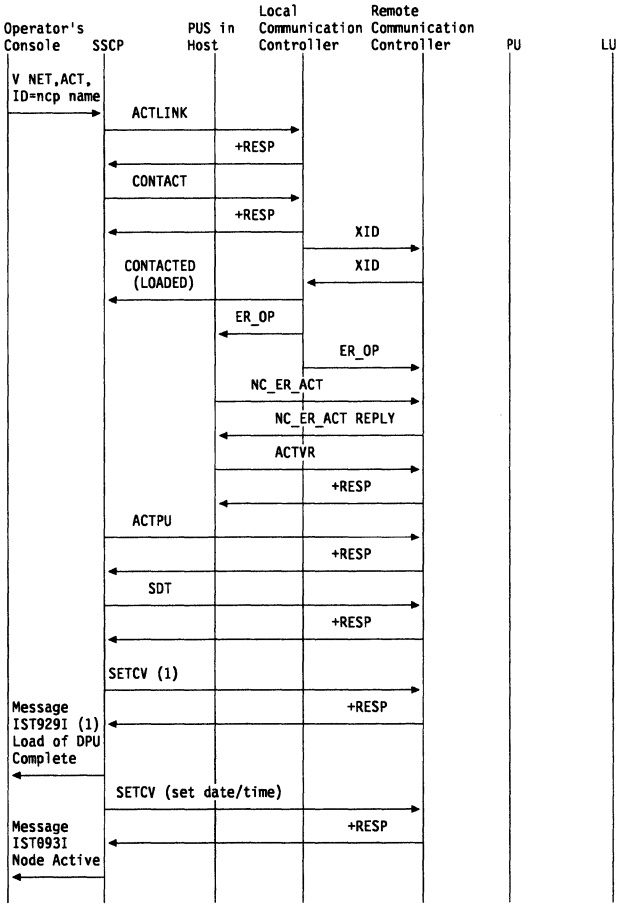


Figure 103. Activating a Link-Attached Communication Controller

1. Flows only for dynamic path definition. SETCV and IST929I flow for each dynamic path definition member specified.

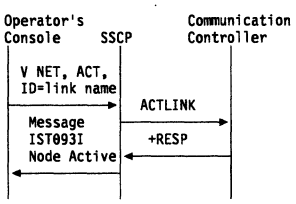


Figure 104. Activating a Link (ACTLINK)

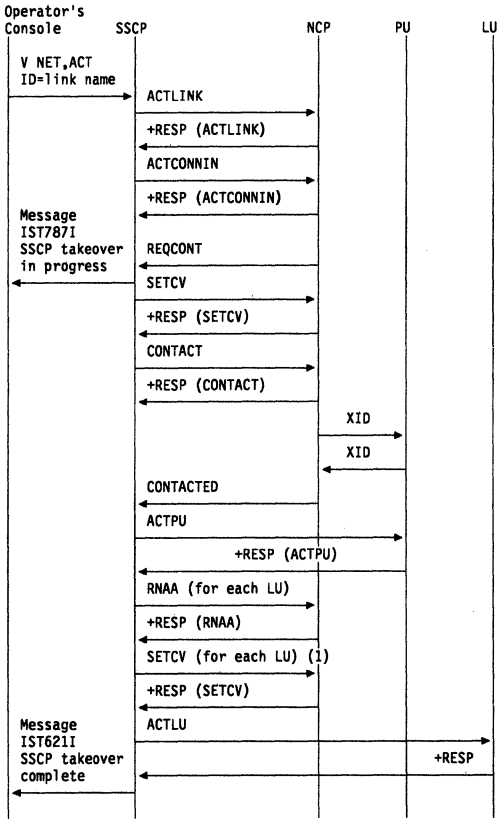


Figure 105. Activating a Switched Link with Takeover

1. SETCV does not flow for NCPs that support peripheral nodes.

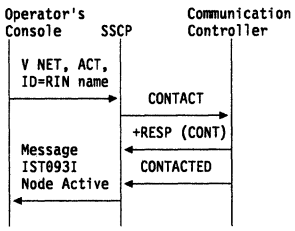


Figure 106. Activating a Cross-Subarea Link Station

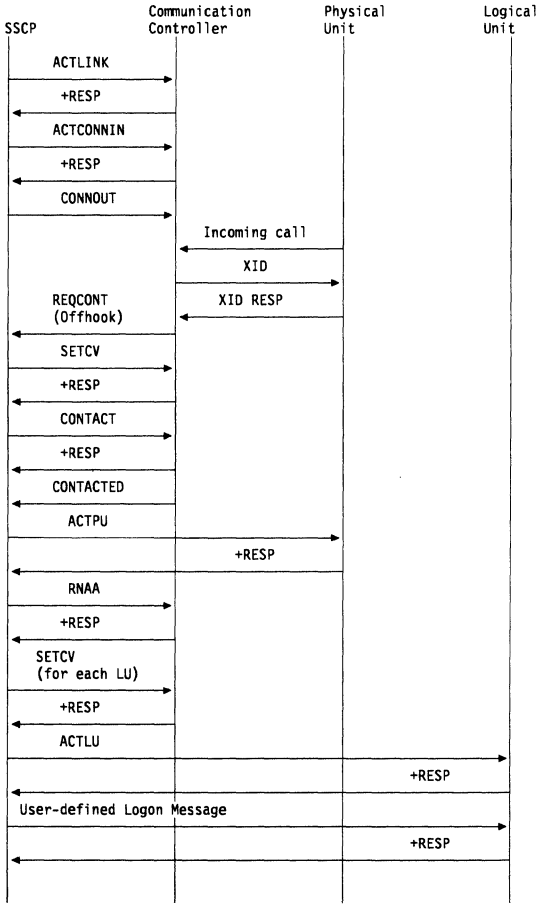


Figure 107. Establishing a Switched Connection

To establish a switched connection, the SSCP sends an Activate Link request to indicate that the link is active. An Activate Connect In request is sent to enable the communication controller to answer incoming calls. (Instead of Activate Connect In, Dial could be sent to initiate an outbound call.) When a call comes in, the communication controller sends an exchange identification (XID) and the physical unit responds with its ID (station address). The communication controller sends a Request Contact (Offhook) request to the SSCP. The SSCP sends a Set Control Vector request containing address and pacing information to the communication controller. The standard activation sequence then occurs.

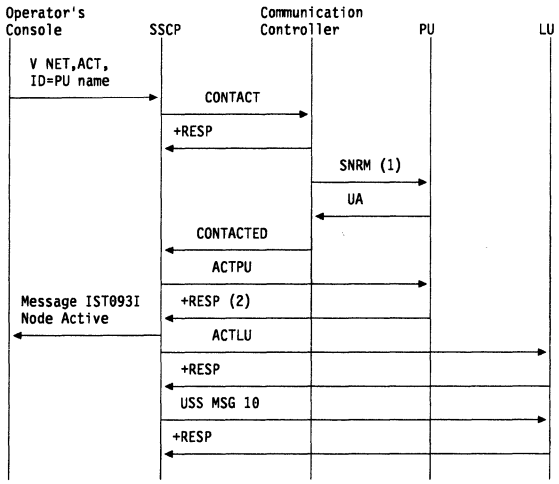


Figure 108. Activating a Physical Unit Type 2.0

1. An XID, instead of a SNRM, will flow to a switched line.
2. Additional RUs flow if the physical unit must be loaded. These RUs are shown in Figure 109 on page 939.

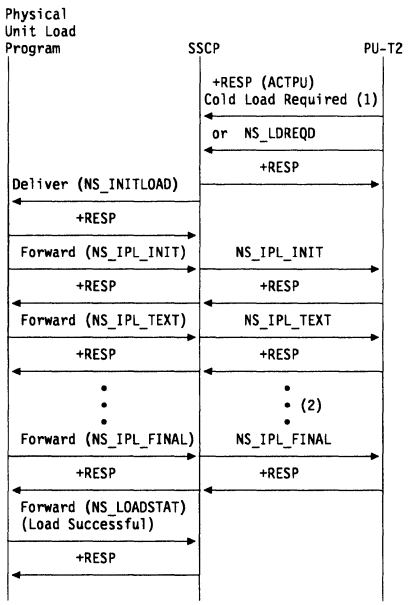


Figure 109. Activating a Physical Unit Type 2.0 with Load Required

1. This figure shows only the RUs that flow when a type 2 physical unit requires loading. For the RUs that flow before and after those in this figure, see Figure 108 on page 938.
2. NS_IPL_TEXT and the response might repeat.

For type 2.0 physical units that require loading before they can be activated, the request for load is indicated in the ACTPU response. (Following activation, the physical unit might request loading with an NS_LDREQD RU.)

The SSCP formats the load request into a network services (NS) RU to initiate the load. The management services subcomponent of the SSCP then sends the embedded request to the physical unit load program of the Downstream Load Utility.

If the physical unit load program **is not** available, it sends a negative response to the SSCP's Deliver RU. The SSCP then sends an NS_IPL_ABORT RU to the physical unit for deactivation processing. (If the load was requested with an NS_LDREQD RU, the physical unit is not deactivated; in fact, it might try the load request again.)

If the physical unit load program **is** available, as in Figure 109, it sends a positive response to the SSCP's Deliver RU. When the load program is complete, it sends a Forward RU, containing an NS_LOADSTAT RU, to relay the status of the load operation.

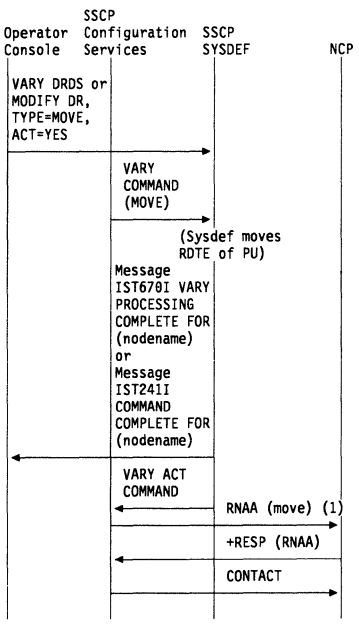


Figure 110. Moving a SYSGENed Physical Unit

1. RNAA flow is as normal. RNAA does not flow for MODIFY DR, TYPE=MOVE, ACT=NO.

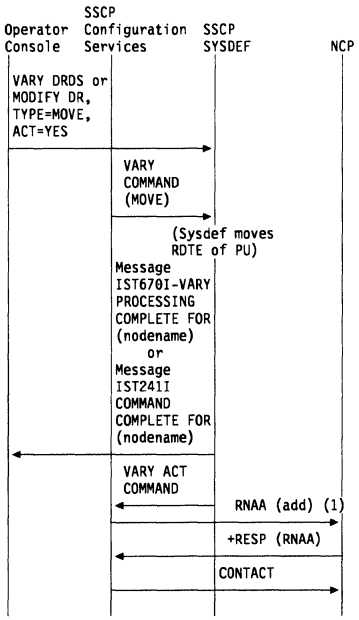


Figure 111. Moving a Dynamically Added Physical Unit

1. RNAA flow is as normal. RNAA does not flow for MODIFY DR, TYPE=MOVE, ACT=NO.

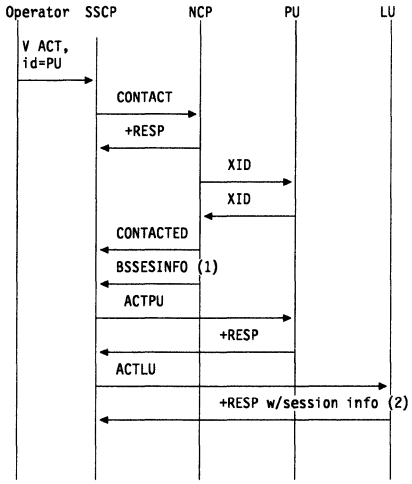


Figure 112. SSCP Takeover of Peripheral Node Logical Units

Note: The following conditions are assumed for this example:

- The physical units being taken over are defined with ANS=CONTINUE, specifying that any LU-LU sessions that are active at SSCP-failure time will continue.
- There are some LU-LU sessions active at failure time under the physical unit being taken over.

1. Independent logical unit only (possible multiple RUs)
2. Dependent logical unit only.

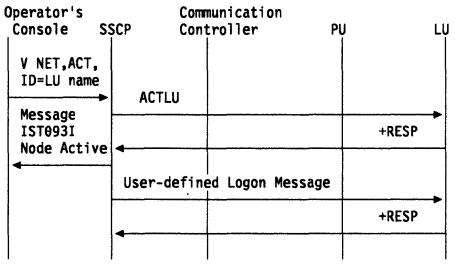


Figure 113. Activating a Logical Unit

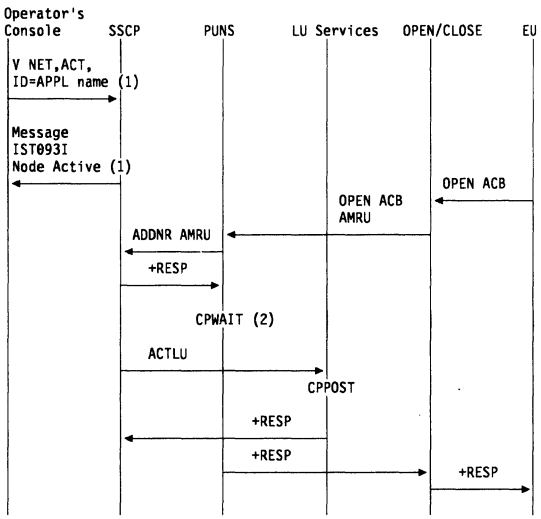


Figure 114. Activating an Application Program and Processing an OPEN ACB Request

1. These do not flow for OPEN ACB processing.
2. PUNS cannot send a response to the OPEN ACB request until LUS receives an ACTLU request for the application program. Therefore, PUNS issues CPWAIT and waits for LUS to post it. Once LUS has received the ACTLU, it posts PUNS, which then sends a response to the OPEN ACB request.

For the close ACB flow see Figure 163 on page 990.

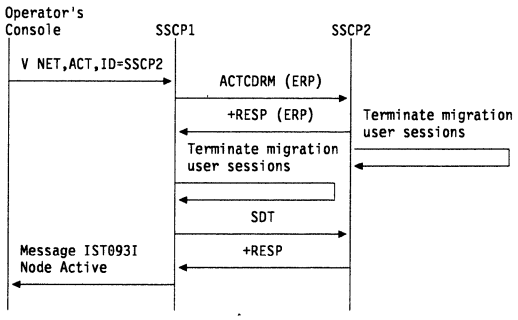


Figure 115. Activating CDRM with ERP Response

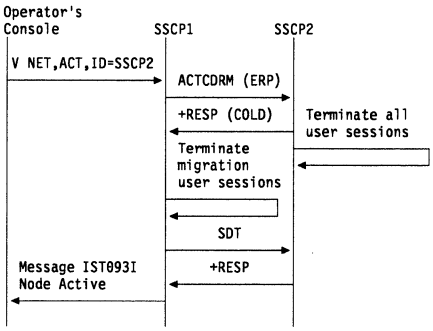


Figure 116. Activating CDRM with COLD Response

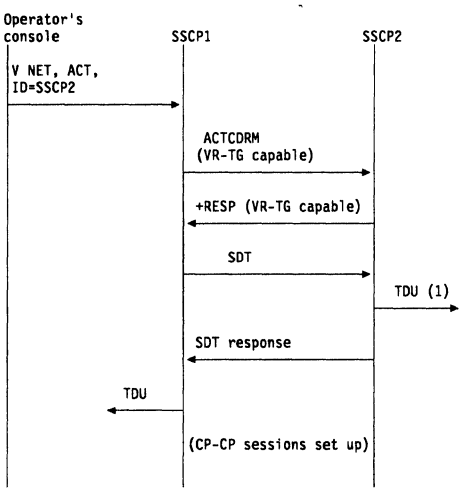


Figure 117. Activating a CDRM with a Virtual-Route-Based Transmission Group

1. If the transmission group is an intermediate routing transmission group (NN–NN), the topology database update (TDU) will be built and broadcast. If the host is a migration data host, the topology database update (TDU) will be built and sent to its server.

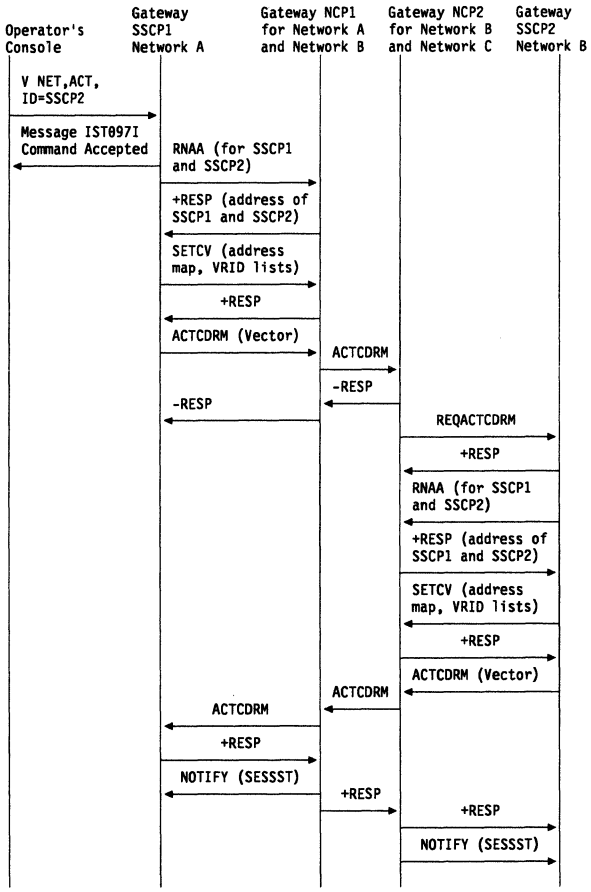


Figure 118. Back-to-Back Gateway NCPs Request Sessions MVS VM

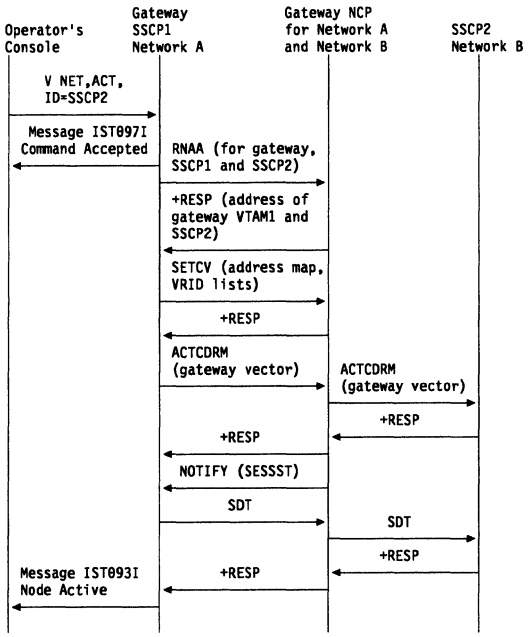


Figure 119. Gateway VTAM Requests Session MVS VM

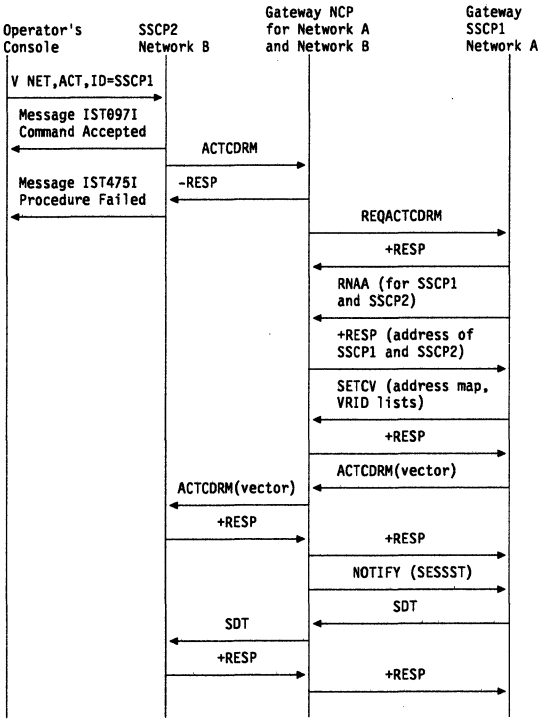


Figure 120. Non-Gateway VTAM Requests Session MVS VM

Session Establishment Flows

Figure 121 on page 951 through Figure 147 on page 975 show the flow of requests and responses to establish single network and cross-network LU-LU sessions.

Index of Session Establishment Flows

Table 67 lists the session establishment flows that are illustrated here.

Table 67 (Page 1 of 2). Index of Session Establishment Flows

Flow	Page
Failed session establishment	
Failure (CDINIT rejection) of session initiation by a secondary logical unit (SLU) for single gateway VTAM and single gateway NCP	Figure 145 on page 973
Failure (CINIT rejection) of setup procedure initiated by a secondary logical unit (SLU) for single gateway VTAM and single gateway NCP	Figure 147 on page 975
Failure (SETCV failure) of session initiation by a secondary logical unit (SLU) for single gateway VTAM and single gateway NCP	Figure 146 on page 974
Gateway VTAM	
Default partitioning of gateway VTAM responsibility spanning three networks	Figure 139 on page 966
Multiple gateway VTAMs and back-to-back gateway NCPs	Figure 122 on page 952
PLU availability for autologon, notification of	Figure 144 on page 972
Requests initiated by primary logical units (PLUs)	
Dependent PLU initiating cross-domain session with independent SLU	Figure 128 on page 957
Independent PLU initiating cross-domain session with independent SLU	Figure 127 on page 956
Independent PLU requesting session with independent SLU through a single gateway VTAM and single gateway NCP	Figure 130 on page 959
OPNDST ACQUIRE	Figure 123 on page 953
PLU initiating request for single gateway VTAM and single gateway NCP	Figure 129 on page 958
PLU initiating request setup queued for single gateway VTAM and single gateway NCP	Figure 131 on page 960
SIMLOGON	Figure 124 on page 953
SIMLOGON(RELREQ)	Figure 125 on page 954
SIMLOGON(RELREQ): Session is pending active or already in progress	Figure 126 on page 955
Requests initiated by secondary logical units (SLUs)	
Dependent SLU initiating cross-domain session with application logical unit (LU)	Figure 136 on page 963
INIT SELF	Figure 134 on page 962
LOGON	Figure 132 on page 961
Predesignated control of gateway NCP by middle host	Figure 139 on page 966
REQSESS	Figure 133 on page 961
Sending an unformatted request to the SSCP	Figure 135 on page 963

Table 67 (Page 2 of 2). Index of Session Establishment Flows

Flow	Page
Single gateway connecting three or more networks	Figure 138 on page 965
Single gateway VTAM and single gateway NCP	Figure 137 on page 964
Requests initiated by third parties	
CLSDST PASS	Figure 140 on page 968
CLSDST PASS with NOTIFY	Figure 141 on page 969
Request spanning three networks	Figure 142 on page 970
VARY LOGON or LOGAPPL processing	Figure 143 on page 971

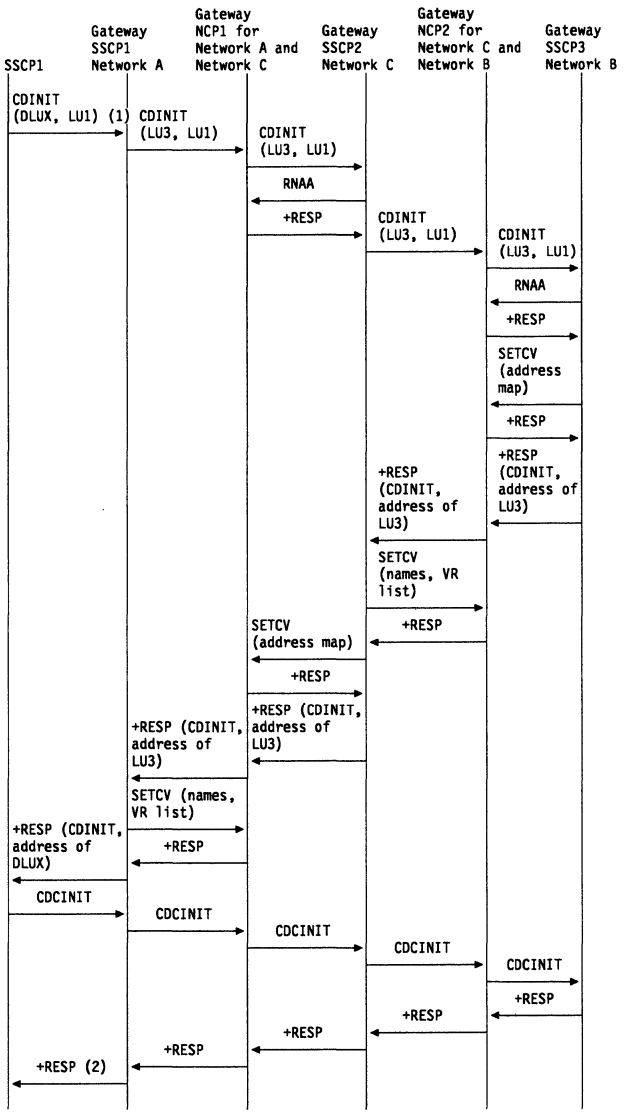


Figure 121. Default Partitioning of Gateway VTAM Responsibility Spanning Three Networks
MVS VM

1. Only the SSCP-SSCP session communication is shown. Assume LU1 (a logical unit owned by SSCP1 in Network A) requests a session with DLUX (an alias for LU3 in Network B; LU3 is a logical unit owned by gateway VTAM3). LU1 is the SLU in the request session.
2. Session setup proceeds as shown in the basic flows. BIND flows from LU3 to gateway NCP2, to gateway NCP1, and to LU1.

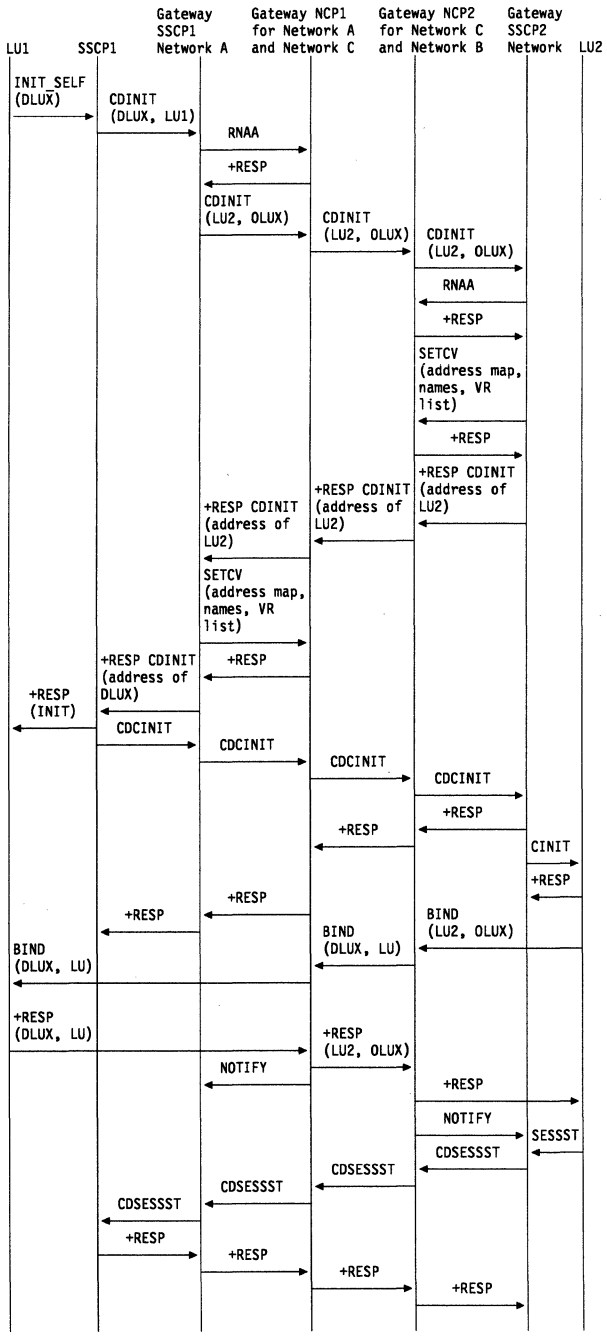


Figure 122. Multiple Gateway VTAMs and Back-to-Back Gateway NCPs MVS VM

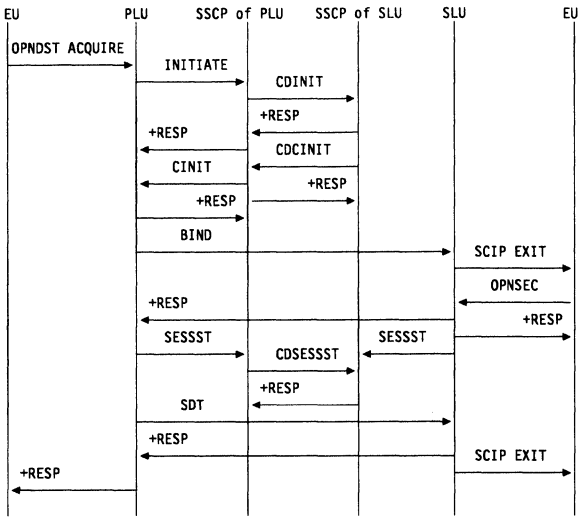


Figure 123. Primary Logical Unit Initiate (OPNDST ACQUIRE)

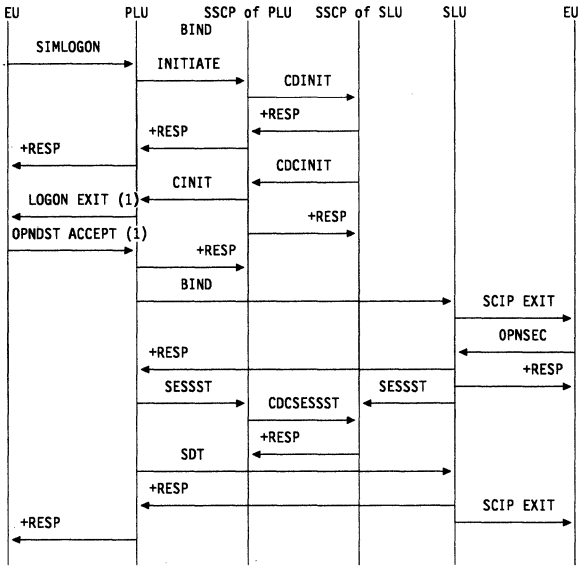


Figure 124. Primary Logical Unit Initiate (SIMLOGON)

1. LOGON EXIT and OPNDST ACCEPT flow only when the PLU is associated with an application program. It does not appear in the flow if the PLU is a device-type logical unit. This is true for many following flows with LOGON EXIT and OPNDST ACCEPT.

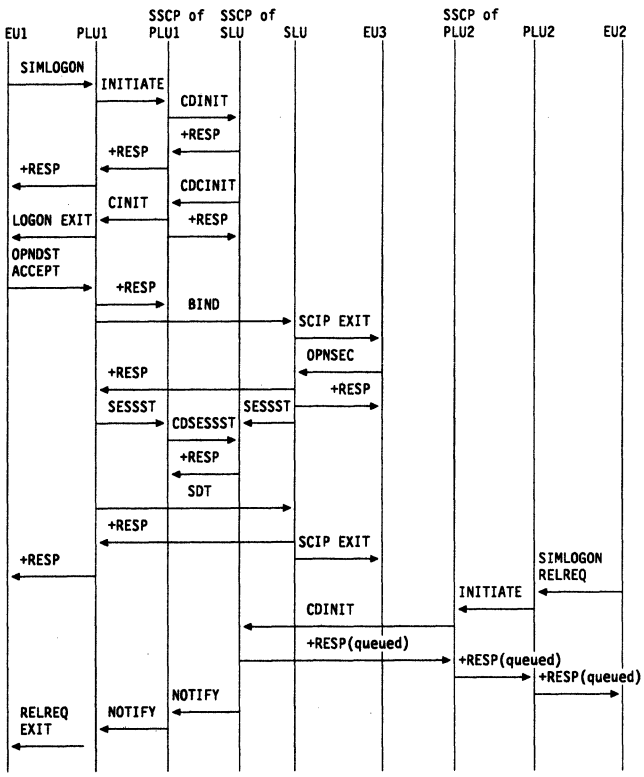


Figure 125. Primary Logical Unit Initiate (SIMLOGON(RELREQ))

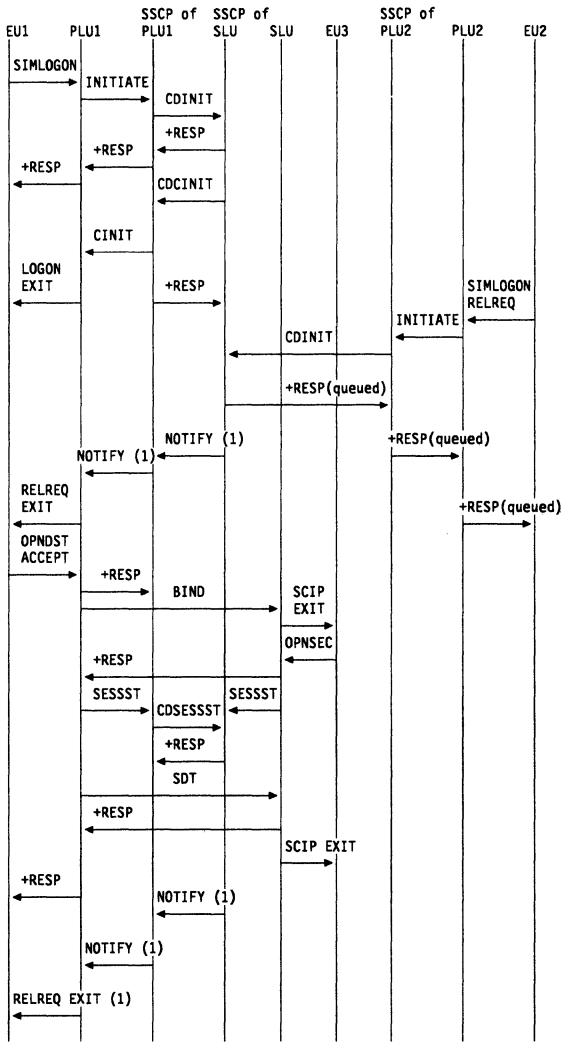


Figure 126. Primary Logical Unit Initiate (SIMLOGON(RELREQ)): Session Is Pending Active or Already in Progress

1. If the session is pending active, the NOTIFY request and the RELREQ EXIT request flow after the response to the CDINIT request from PLU2's SSCP. If the session is already in progress, these requests flow after the response to the OPNDST ACCEPT.

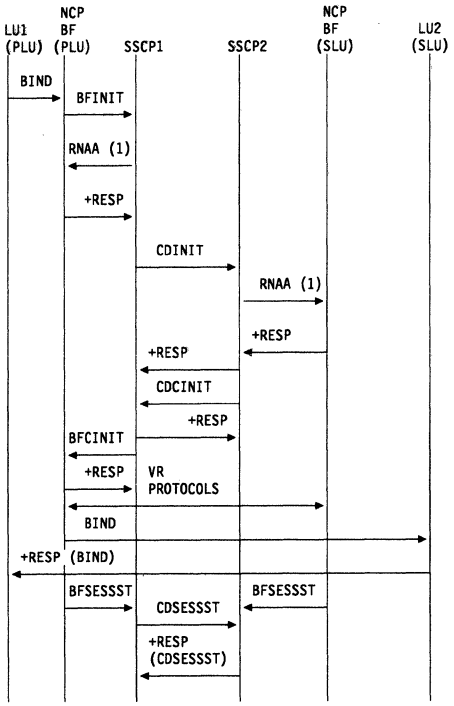


Figure 127. Independent PLU Initiating Cross-Domain Session with Independent SLU

1. RNAA flows only if the network address is needed.

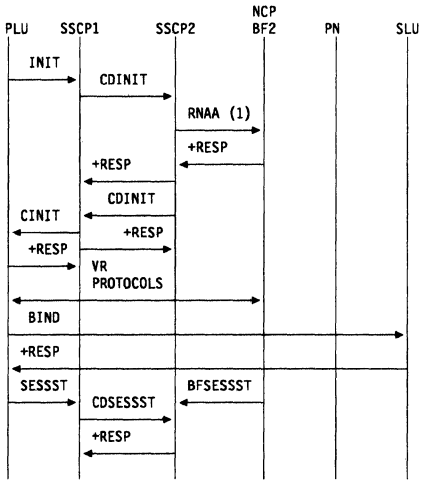


Figure 128. Dependent PLU Initiating Cross-Domain Session with Independent SLU

1. RNAA flows only if the network address is needed.

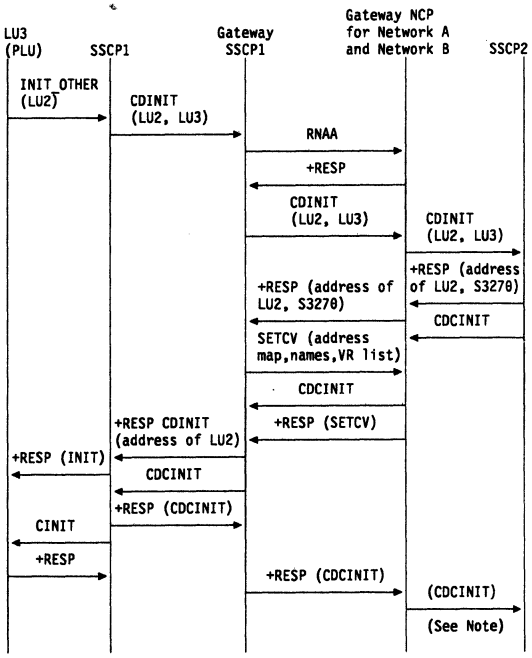


Figure 129. PLU-Initiated Request for Single Gateway VTAM and Single Gateway NCP
MVS VM

Note: Session setup continues as in the flow for a SLU-initiated session.

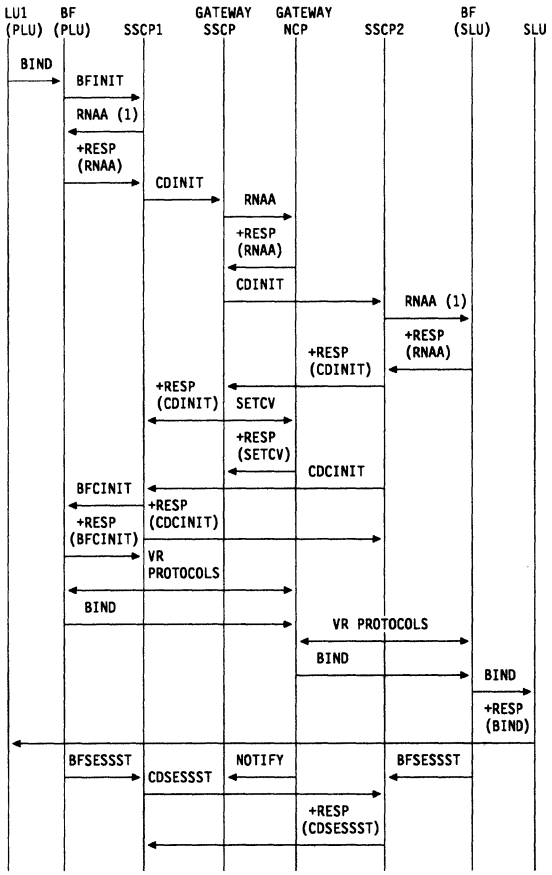


Figure 130. Independent PLU Requesting Session with Independent SLU through Single Gateway VTAM and Single Gateway NCP MVS VM

1. RNAA flows only if the network address is needed.

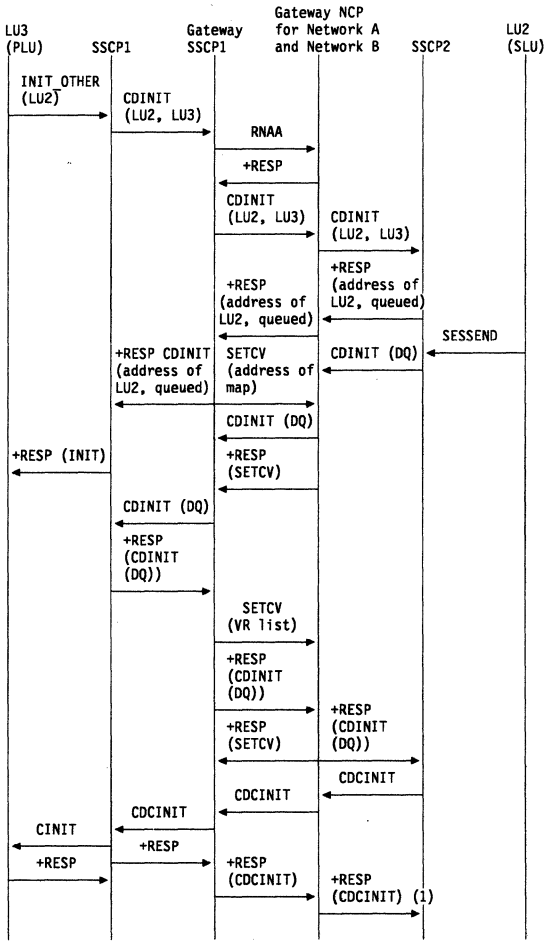


Figure 131. PLU-Initiated Request Setup Queued for Single Gateway NCP and Single Gateway VTAM MVS VM

1. Session setup continues as in the flow for an SLU-initiated session.

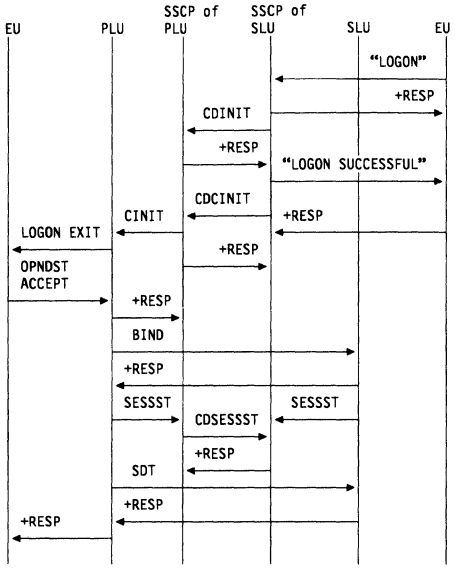


Figure 132. Secondary Logical Unit Initiate (LOGON)

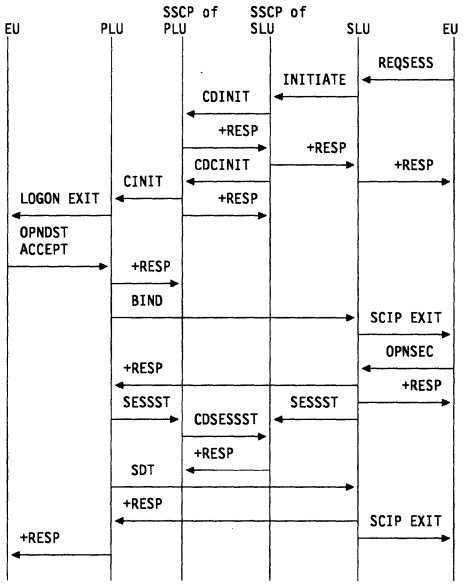


Figure 133. Secondary Logical Unit Initiate (REQSESS)

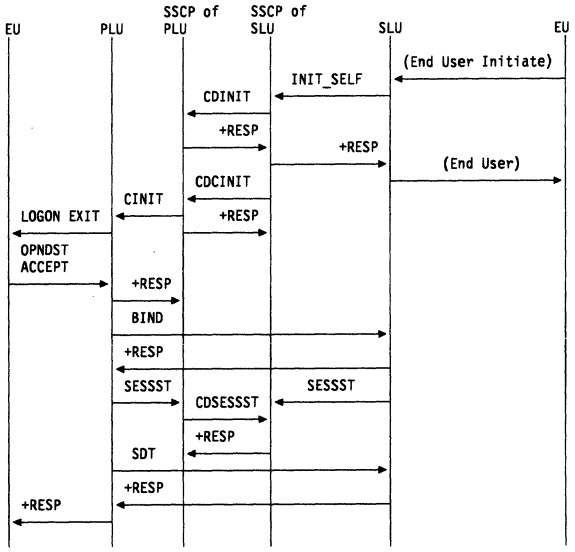


Figure 134. Secondary Logical Unit Initiate (INIT SELF)

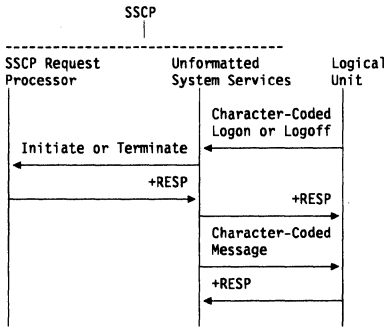


Figure 135. Sending an Unformatted Request to the SSCP

In this example, the logical unit sends a character-coded logon or logoff to the SSCP. The unformatted system services portion of SSCP converts the logon into a field-formatted Initiate Self or Terminate Self request. The request is then passed to the SSCP request processor.

If the return code indicates an unsuccessful transmission, the unformatted system services portion of SSCP converts the request into a form that can be understood by the terminal logical unit.

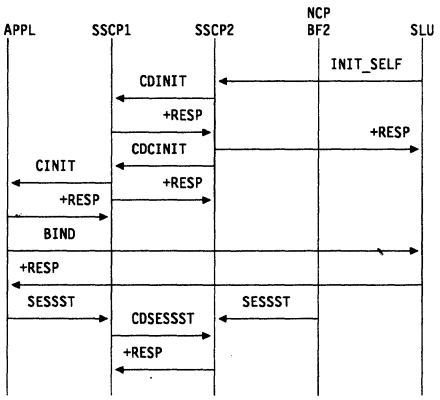


Figure 136. Dependent SLU Initiating a Cross-Domain Session with Application LU

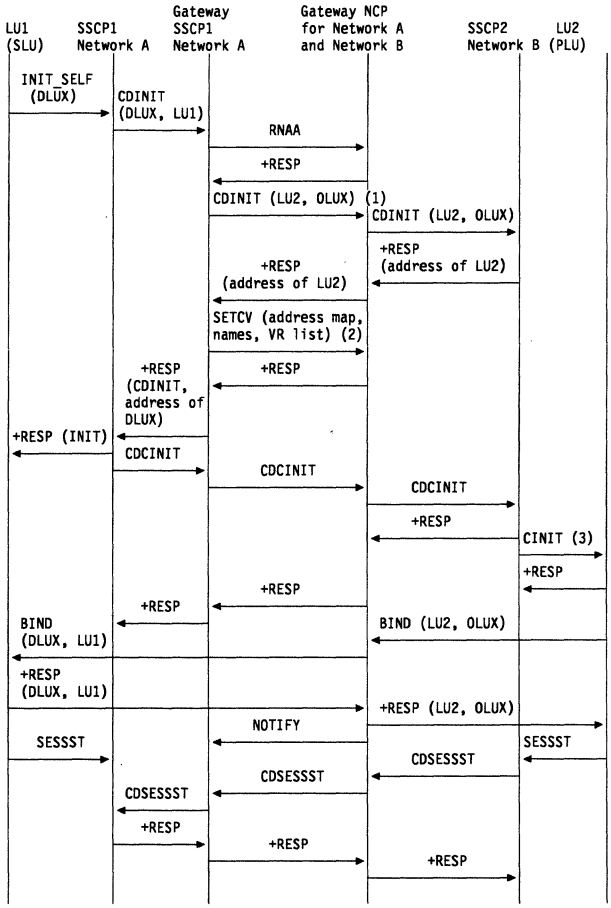


Figure 137. SLU Initiating Request for Single Gateway VTAM and Single Gateway NCP
MVS VM

1. LU1 is initiating a session with DLUX. Using alias name translation, SSCP1 translates DLUX to LU2 and LU1 to OLUX.
2. Names are sent to allow substitution in the BIND.
3. The CINIT drives the logon exit.

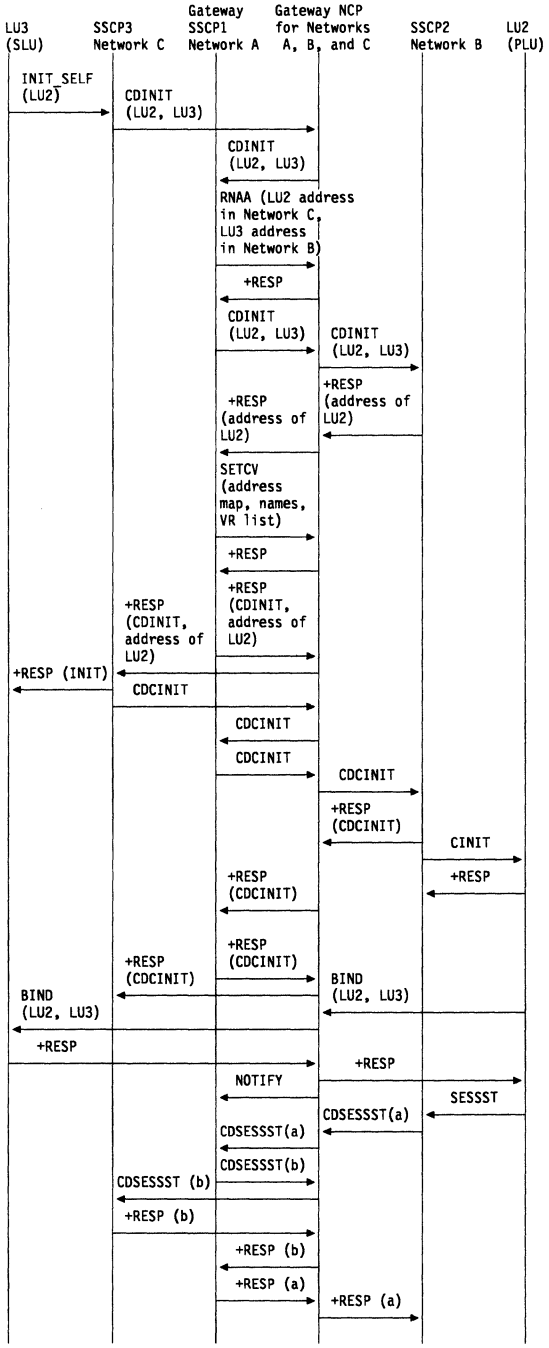


Figure 138. SLU -Initiating Request for Single Gateway Connecting Three or More Networks
MVS VM

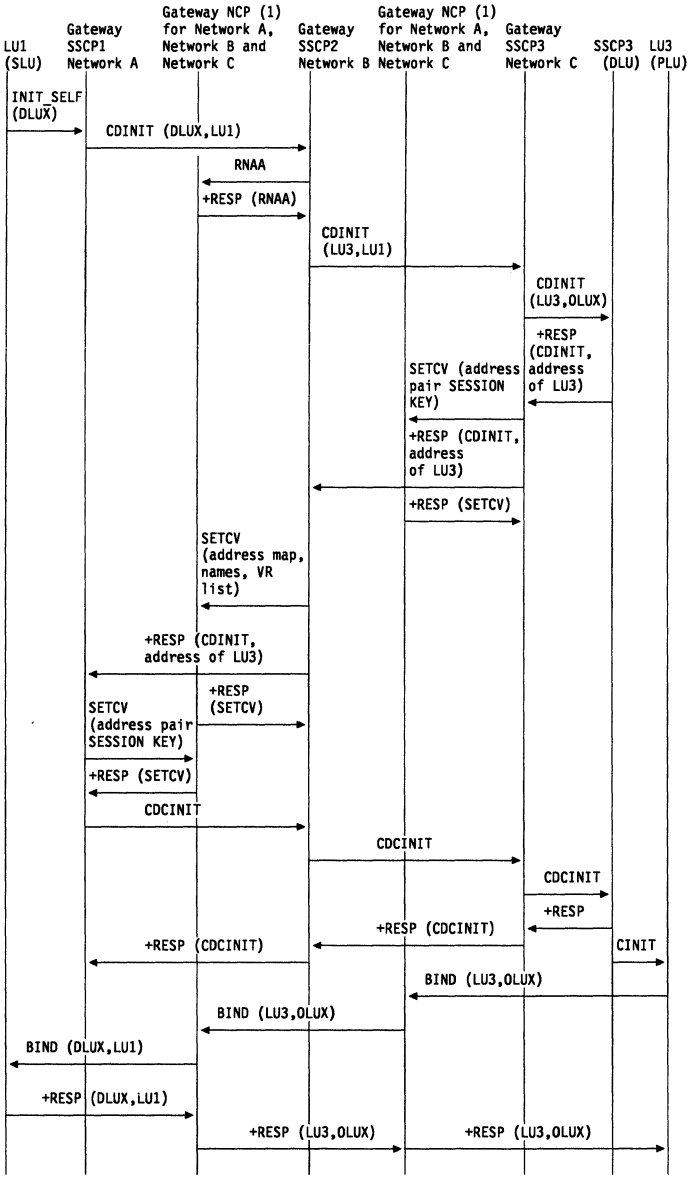


Figure 139 (Part 1 of 2). SLU Initiating Request for Predesignated Control of Gateway NCP by Middle Host MVS VM

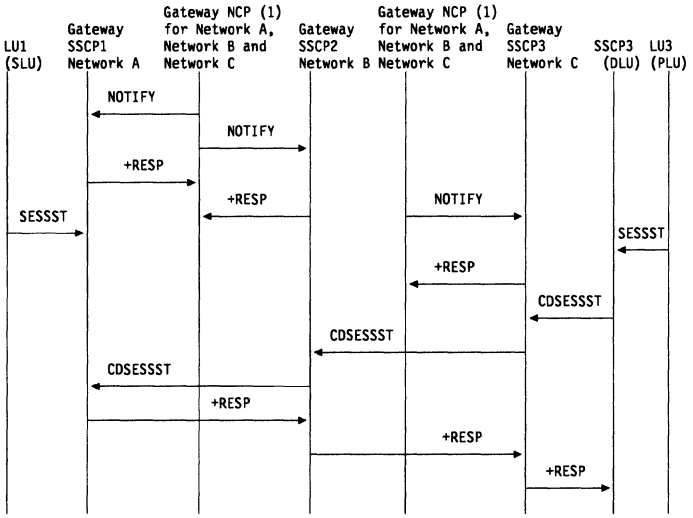


Figure 139 (Part 2 of 2). SLU Initiating Request for Predesignated Control of Gateway NCP by Middle Host MVS VM

1. To simplify the flow, the gateway NCP has been shown twice in this flow.

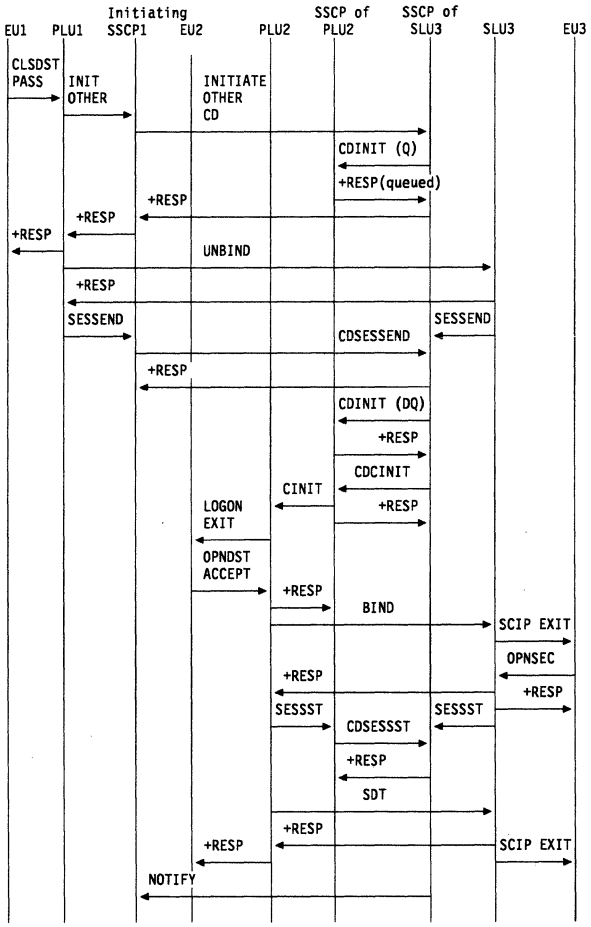


Figure 140. Third-Party Initiating CLSDST PASS

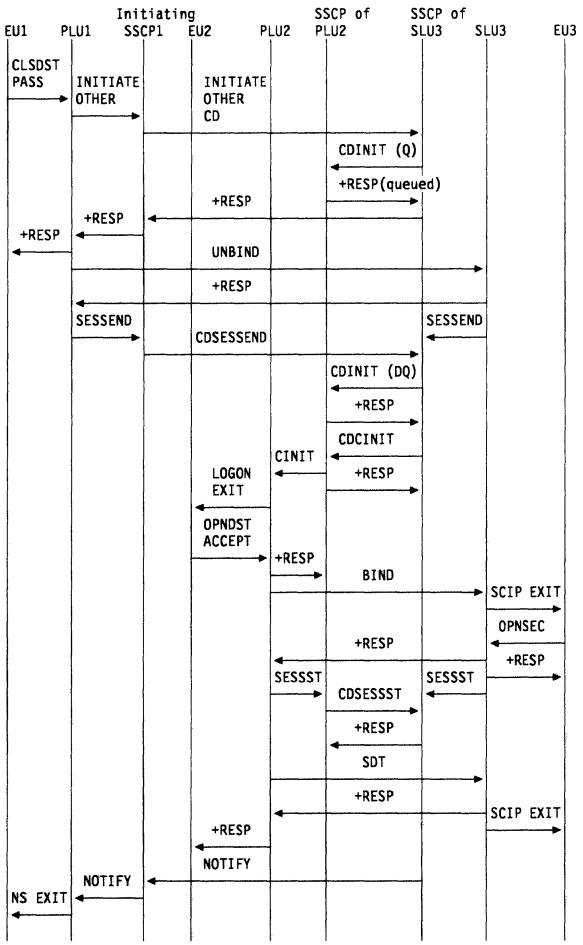


Figure 141. Third-Party Initiating CLSDST PASS with NOTIFY

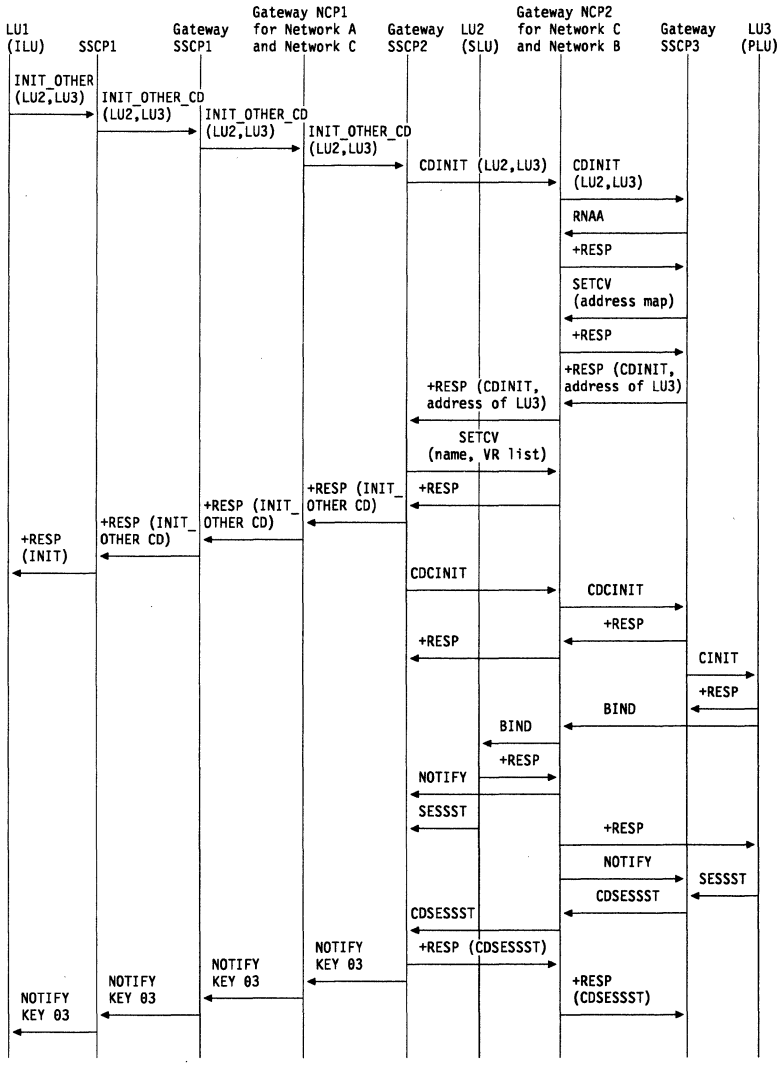


Figure 142. Third Party Initiating Request Spanning Three Networks MVS VM

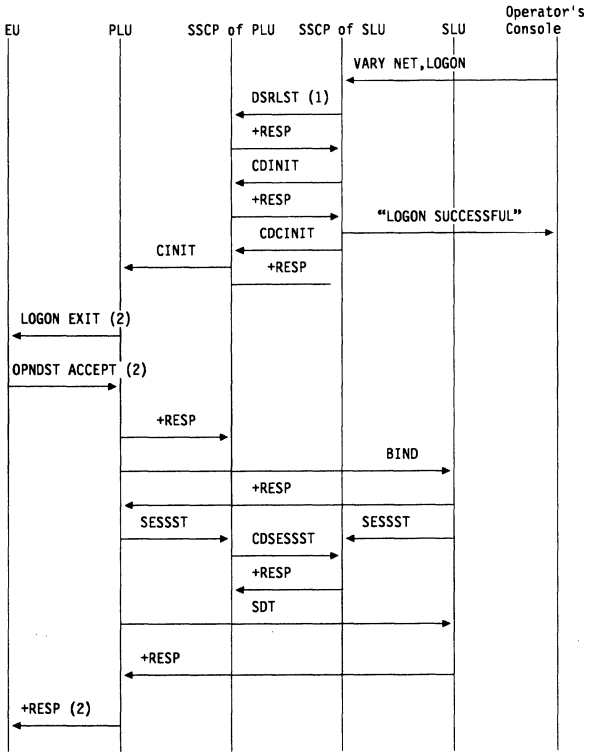


Figure 143. Initiating Session Using VARY NET,LOGON or LOGAPPL

1. Optional; occurs only when SLU is a dial device.
2. This applies only when the PLU is associated with an application program. It does not appear in the flow if the PLU is a device-type logical unit.

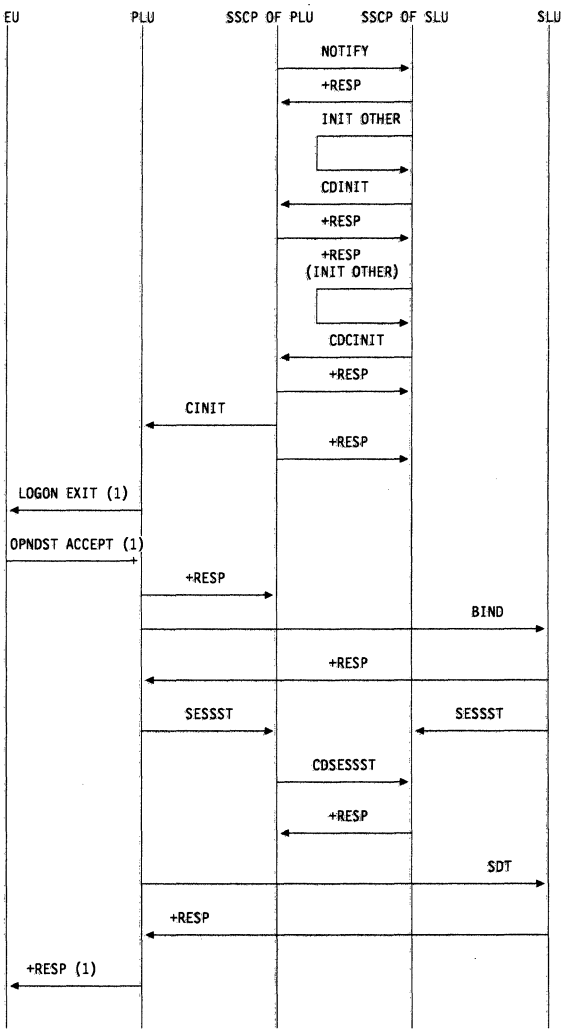


Figure 144. Notification of PLU Availability for Autologon

1. This applies only when the PLU is associated with an application program. It does not appear in the flow if the PLU is a device-type logical unit.

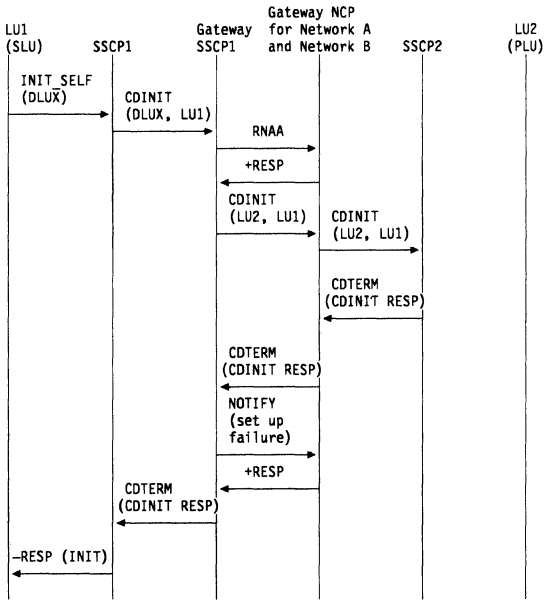


Figure 145. Failure (CDINIT Rejection) of Session Initiation by an SLU for Single Gateway VTAM and Single Gateway NCP MVS VM

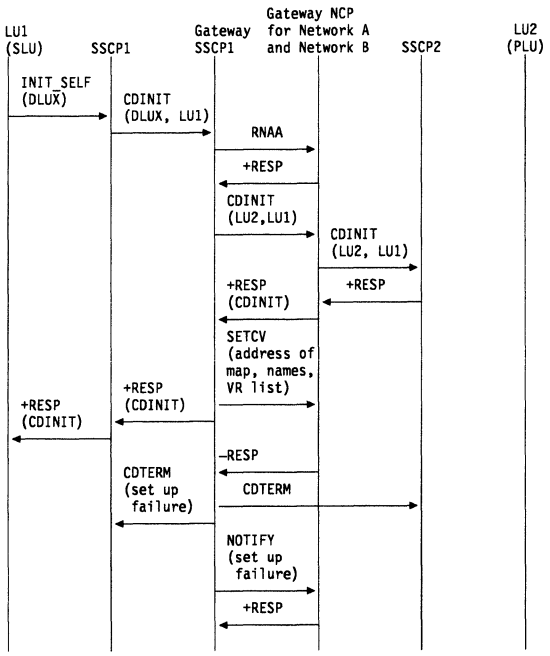


Figure 146. Failure (SETCV Failure) of Session Initiation by an SLU for Single Gateway VTAM and Single Gateway NCP MVS VM

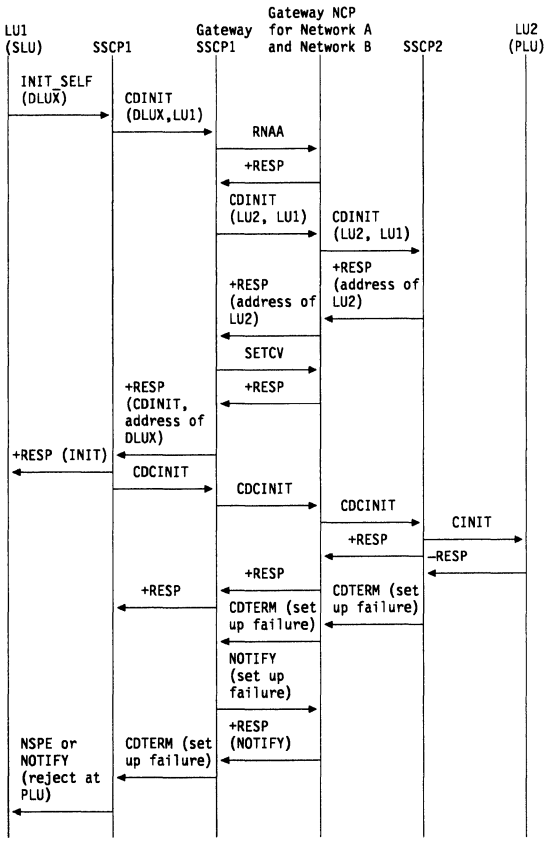


Figure 147. Failure (CINIT Rejection) of Setup Procedure Initiated by an SLU for Single Gateway VTAM and Single Gateway NCP MVS VM

Deactivation and Session Termination Flows

Figure 148 on page 978 through Figure 173 on page 999 show the flow of requests and responses between the SSCP and logical and physical units to deactivate resources and end sessions.

Index of Deactivation and Session Termination Flows

Table 68 lists the deactivation and session termination flows that are illustrated here.

Table 68 (Page 1 of 2). Index of Deactivation and Session Termination Flows

Flow	Page
CLOSE ACB processing	Figure 163 on page 990
Deactivating an application program	Figure 164 on page 991
Deactivating a CDRM	
Forced	Figure 167 on page 993
Forced, without affecting active sessions	Figure 169 on page 995
Forced or immediate, VTAM releases prior to V3R4.1	Figure 170 on page 996
Immediate	Figure 166 on page 993
Immediate, without affecting active sessions	Figure 168 on page 994
Normal	Figure 165 on page 992
Deactivating a logical unit (LU), single network	
Forced	Figure 149 on page 978
Immediate	Figure 148 on page 978
VARY NET,TERM Cleanup	Figure 161 on page 988
VARY NET,TERM Unconditional	Figure 160 on page 987
With Giveback	Figure 150 on page 979
Deactivating a logical unit (LU), multiple networks	
Independent PLU sends BFCLEANUP for independent SLU	Figure 151 on page 980
Independent PLU sends UNBIND for independent SLU	Figure 152 on page 980
PLU sends UNBIND for multiple gateway VTAMs and single gateway NCP	Figure 153 on page 981
PLU sends UNBIND for single gateway VTAM and single gateway NCP	Figure 154 on page 981
SLU requests TERMINATE SELF (CLEANUP) for single gateway VTAM and single gateway NCP	Figure 156 on page 983
SLU requests TERMINATE SELF for multiple gateway VTAMs and back-to-back gateway NCPs	Figure 155 on page 982
SLU requests TERMINATE SELF for single gateway VTAM and single gateway NCP	Figure 157 on page 984
Type 2.1 nodes, active termination	Figure 158 on page 985
Deactivating a physical unit (PU) acting as an adjacent link station for independent logical unit (LU) sessions	Figure 159 on page 986
Queued session, terminating	Figure 162 on page 989
Route failure	
Route failure in intermediate network causes termination of LU-LU sessions	Figure 172 on page 998

Table 68 (Page 2 of 2). Index of Deactivation and Session Termination Flows

Flow	Page
Route failure in intermediate network causes termination of SSCP-SSCP sessions	Figure 173 on page 999
SSCP-SSCP session termination causes LU-LU sessions to be broken	Figure 171 on page 997

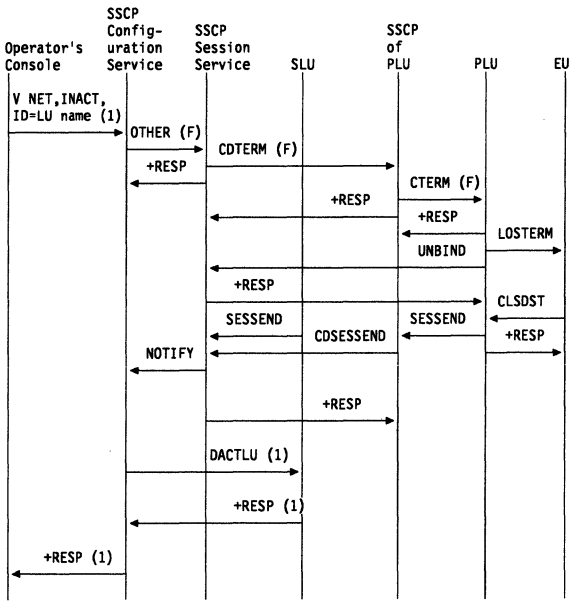


Figure 148. Deactivating a Logical Unit: Immediate

1. These flow only when the operator deactivates a specific logical unit. For example, they do not flow during immediate deactivation of a CDRM.

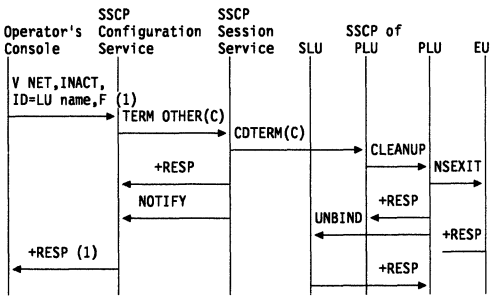


Figure 149. Deactivating a Logical Unit: Forced

1. These flow only when the operator deactivates a specific logical unit. For example, they do not flow during forced deactivation of a CDRM.

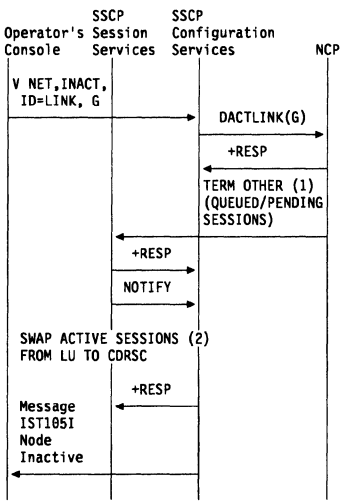


Figure 150. Deactivating a Logical Unit with Giveback

1. The DACTLINK X'02' terminates only queued and pending LU-LU sessions. Active LU-LU sessions remain active.
2. After session services transfers SIBs of ACTIVE logical units to the CDRSC, configuration services SRTADDs the CDRSCs as real resources, and the logical units are ADDed as shadow resources. If a CDRSC for a particular logical unit does not exist, a dynamic CDRSC is allocated for the logical unit.

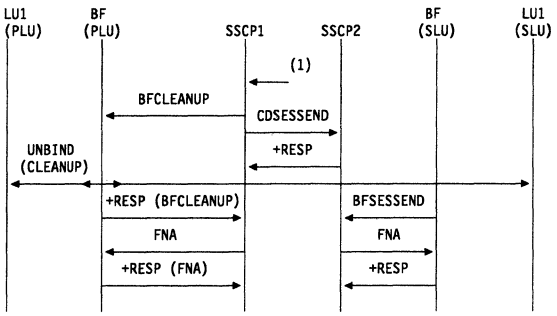


Figure 151. Independent Primary Logical Unit (PLU) Sends BFCLEANUP for Cross-Domain LU-LU Session with Independent Secondary Logical Unit (SLU) MVS VM

1. BFCLEANUP can be sent by the SSCP(PLU) for several reasons, including:
   - A network operator at the SSCP(PLU) issues a VARY NET,TERM,UNCOND, generating an internal TERM-OTHER(forced).
   - A network operator at either SSCP issues a VARY NET,INACT,ID=cdrm, deactivating all cross-domain sessions between the SSCPs.

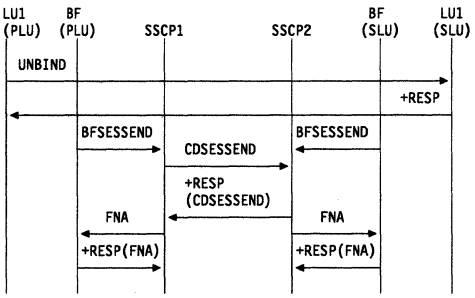


Figure 152. Independent Primary Logical Unit (PLU) Sends UNBIND for Cross-Domain LU-LU Session with Independent Secondary Logical Unit (SLU) MVS VM

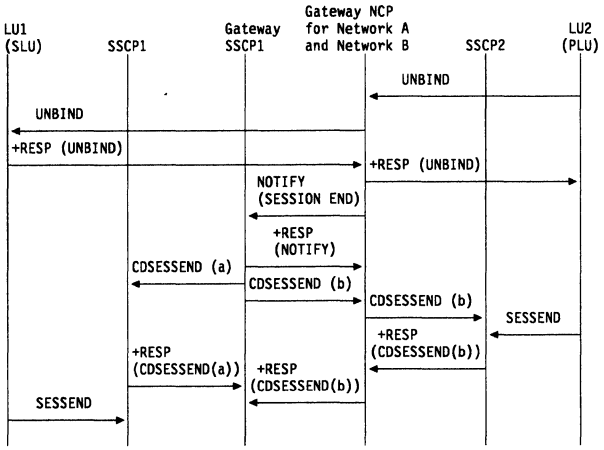


Figure 153. Primary Logical Unit (PLU) Sends UNBIND for Multiple Gateway VTAMs and Single Gateway NCP MVS VM

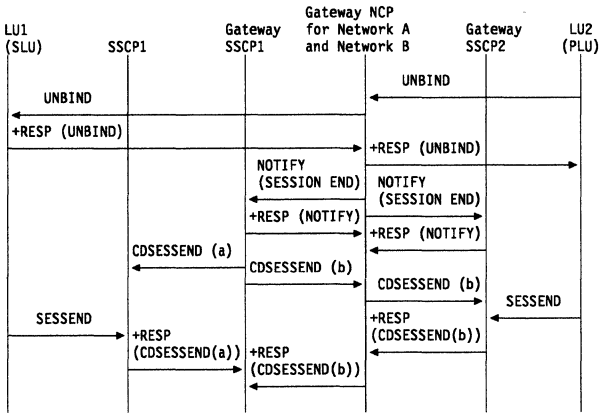


Figure 154. Primary Logical Unit (PLU) Sends UNBIND for Single Gateway VTAM and Single Gateway NCP MVS VM

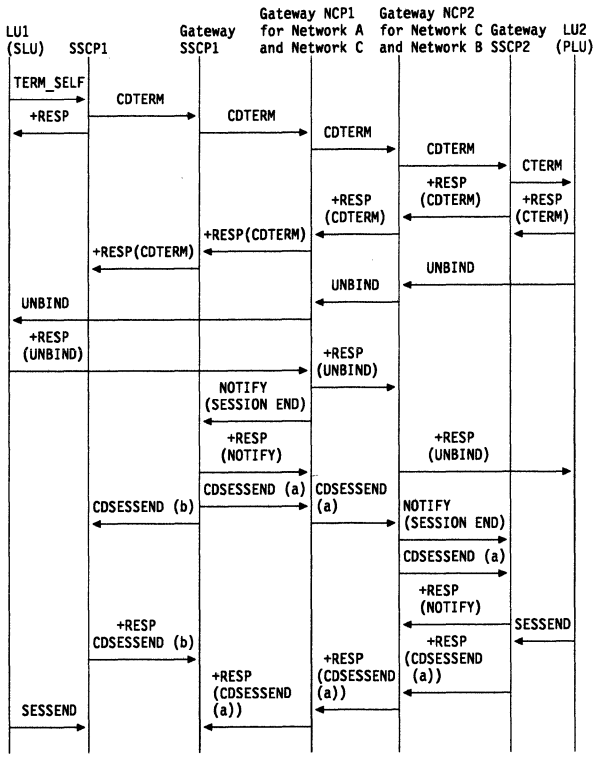


Figure 155. Secondary Logical Unit (SLU) Requests TERMINATE SELF for Multiple Gateway VTAMs and Back-to-Back Gateway NCPs MVS VM

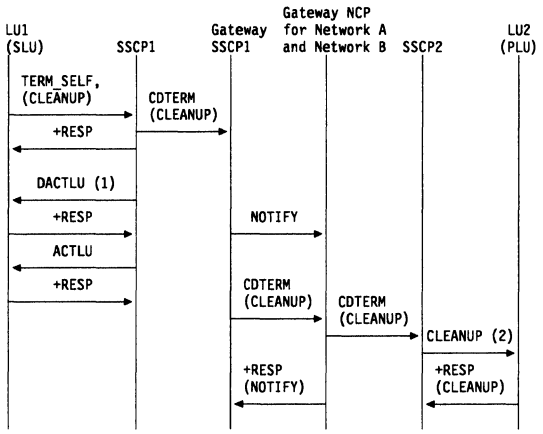


Figure 156. Secondary Logical Unit (SLU) Requests TERMINATE SELF (CLEANUP) for Single Gateway VTAM and Single Gateway NCP MVS VM

Note: The UNBIND can flow from the SLU, the PLU, or the gateway NCP.

1. A DACTLU does not flow to a binary synchronous communication (BSC) terminal.
2. You might receive sense code 081E0003, indicating that cleanup has already occurred.

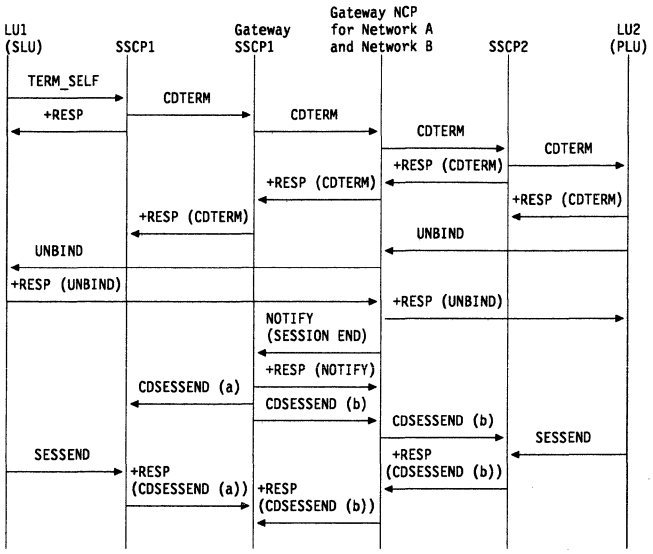


Figure 157. Secondary Logical Unit (SLU) Requests Terminate Self for Single Gateway VTAM and Single Gateway NCP MVS VM

Note: (a) and (b) are used here to differentiate between similar request units.

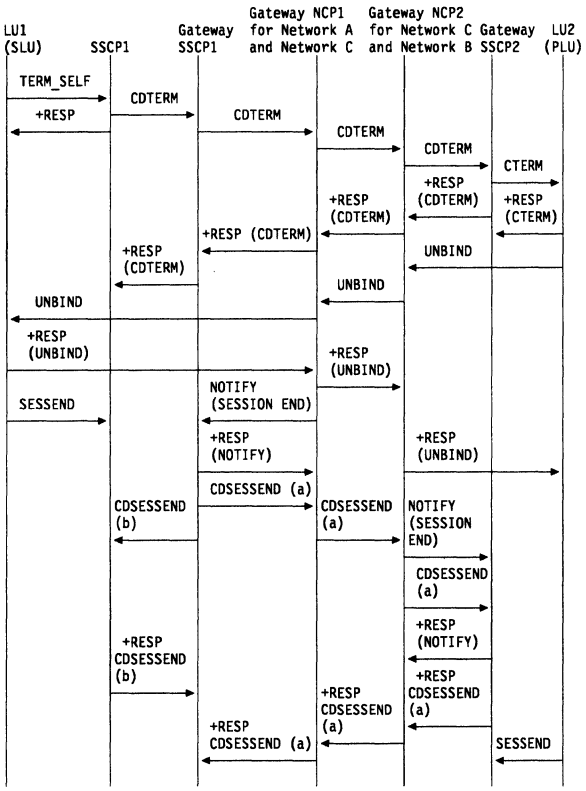


Figure 158. Active Session Termination of Type 2.1 Nodes MVS VM

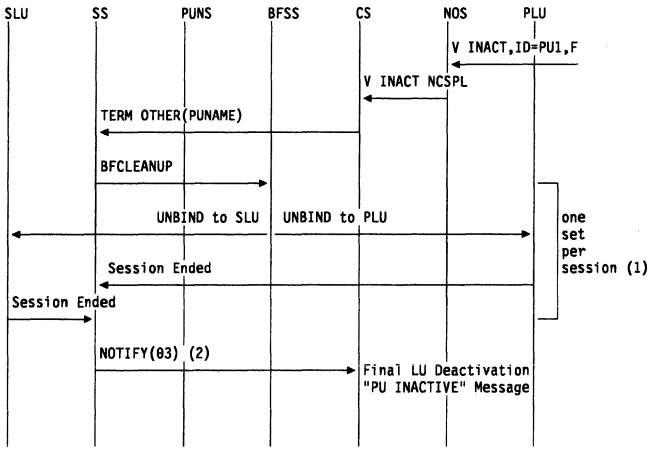


Figure 159. Deactivating a PU Acting as an Adjacent Link Station for Independent LU Sessions

1. Examines PLU and SLU chain and sends (BF)CLEANUP for each ILU session found.
2. When all Session Ended (or BFSESSEND) signals are received, NOTIFY is sent to CS for PU.

In this example, PU1 is a fictitious adjacent link station. When the PU is deactivated, configuration services sends a TERMINATE containing the PU name to session services. Session services examines the adjacent link station's SIB chains and sends CLEANUP to terminate the sessions. When all sessions are down, NOTIFY flows to configuration services so the final deactivation can occur.

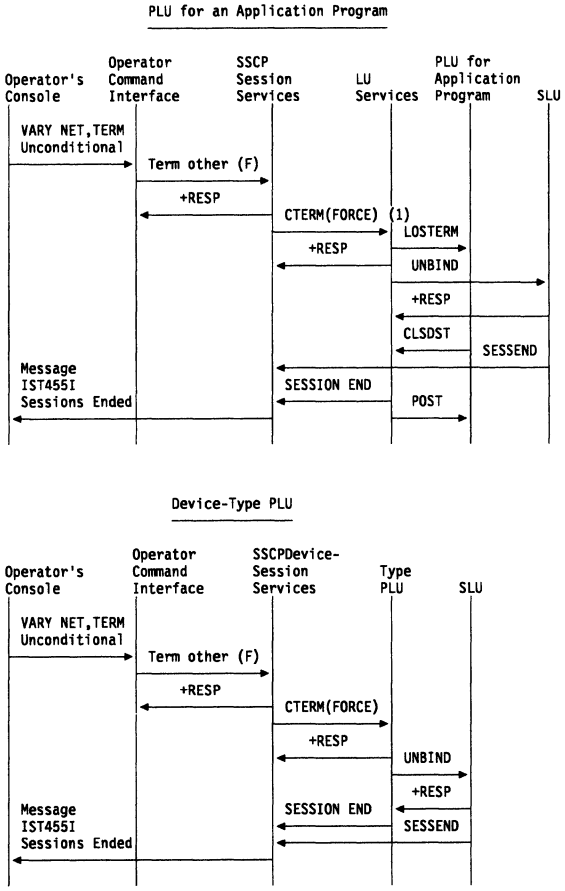


Figure 160. Deactivating Sessions or LUs Using VARY NET,TERM Unconditional

1. If the LOSTERM exit routine is already scheduled with a reason code 32 caused by a CTERM (orderly) request that was received before, the CTERM (force) request is upgraded to a CLEANUP RU and VTAM drives an NSEXIT exit routine.

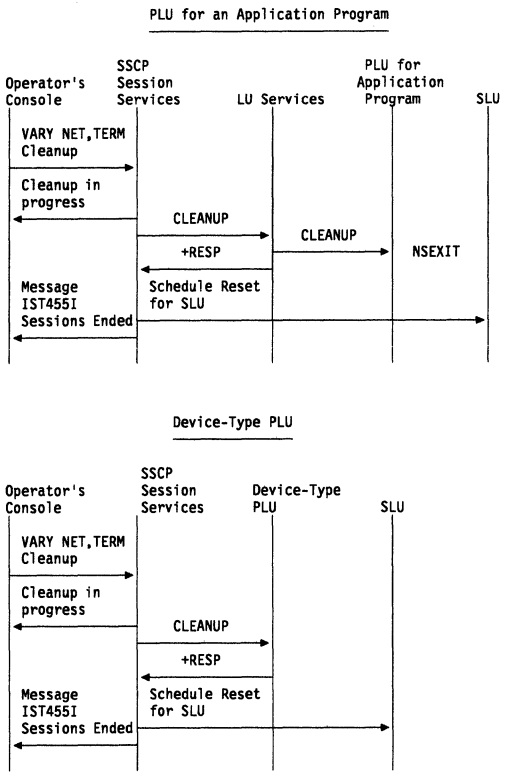


Figure 161. Deactivating Sessions or LUs Using VARY NET,TERM Cleanup

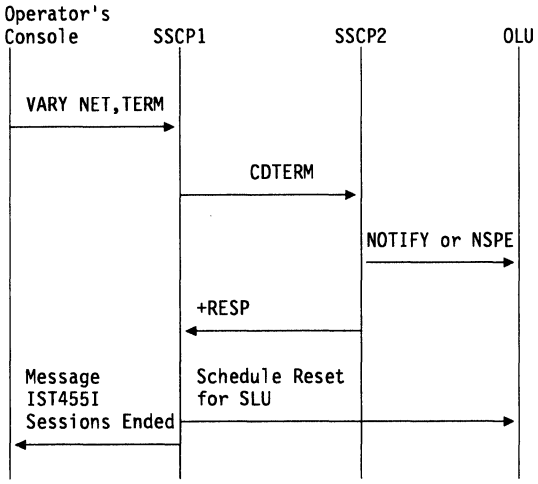


Figure 162. Terminating A Queued Session

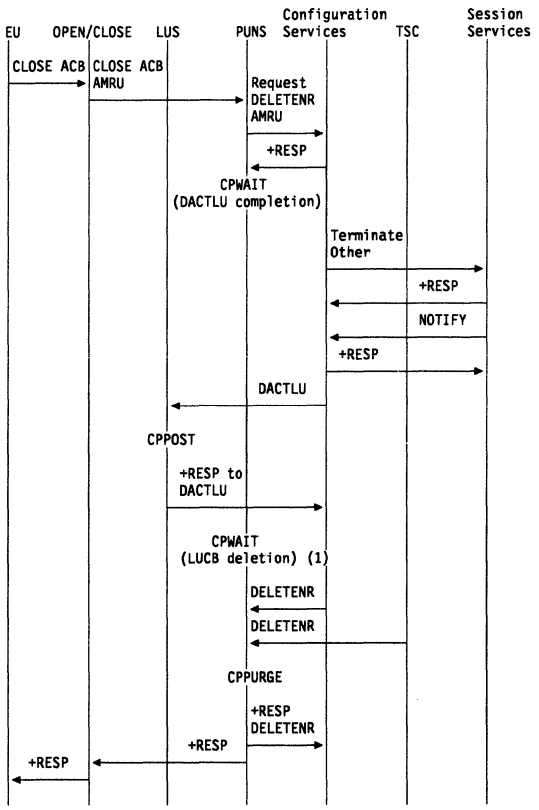


Figure 163. CLOSE ACB Processing

1. PUNS cannot send a response to the CLOSE ACB AMRU until DACTLU processing is complete and the LUCB has been deleted. Therefore, after requesting that configuration services deactivate the logical unit, PUNS issues CPWAIT and waits for LUS to post it when the logical unit has been deactivated. Once it is posted, PUNS waits to be notified that there are no more active sessions for the application program. PUNS issues CPWAIT and waits for configuration services and TSC to send a request to delete the LUCB. PUNS posts itself when it has processed each of these requests and sends a response to configuration services to notify it that the LUCB has been deleted. After sending this response, PUNS sends a response to the CLOSE ACB AMRU.

For the open ACB flow see Figure 114 on page 943.

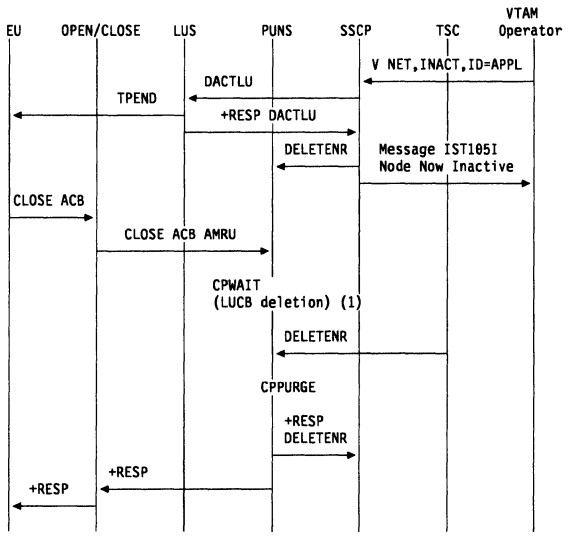


Figure 164. Deactivating an Application Program

1. PUNS cannot send a response to the CLOSE ACB AMRU until there are no more active sessions for the application program. Therefore, after the SSCP sends a request to delete the LUCB, PUNS waits for OPEN/CLOSE to send it a CLOSE ACB AMRU. Once it has received this request, PUNS issues CPWAIT and waits for TSC to send a request to delete the LUCB. PUNS posts itself when it has deleted the LUCB and sends a response to configuration services. After sending this response, PUNS sends a response to the CLOSE ACB AMRU.

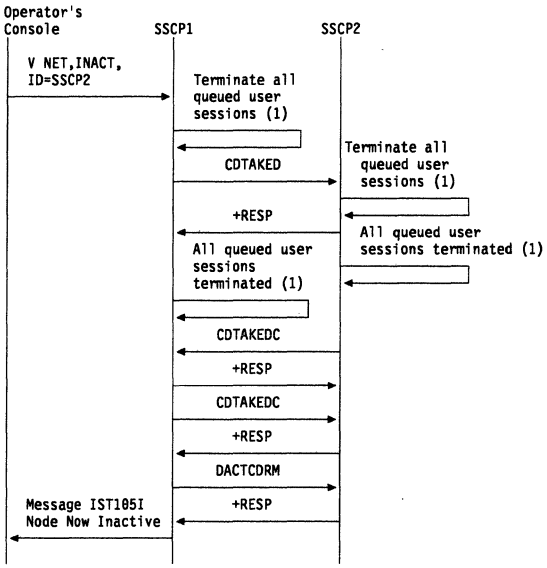


Figure 165. Deactivating a CDRM: Normal

Note: When the CDRM is actually deactivated, immediate processing takes place. See Figure 166 on page 993 for the RUs that flow for immediate deactivation of a CDRM.

1. See Figure 162 on page 989 for the RUs that flow for termination of a queued session.

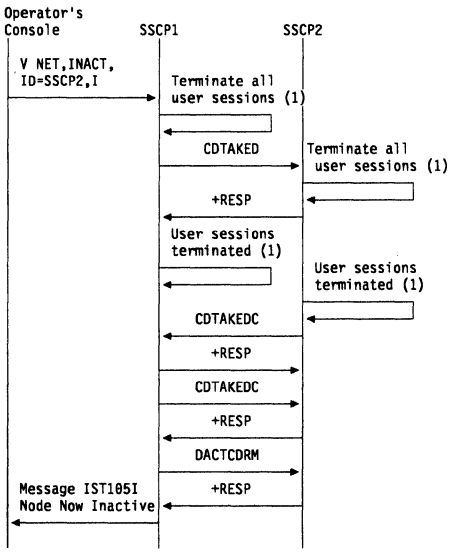


Figure 166. Deactivating a CDRM: Immediate

1. The logical unit will not be deactivated. See Figure 161 on page 988 for the RUs that flow for immediate deactivation of a logical unit.

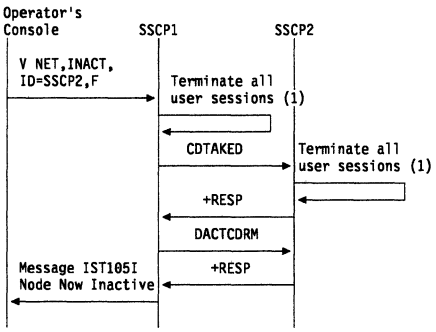


Figure 167. Deactivating a CDRM: Forced

1. The logical unit will not be deactivated. See Figure 161 on page 988 for the RUs that flow for forced deactivation of a logical unit.

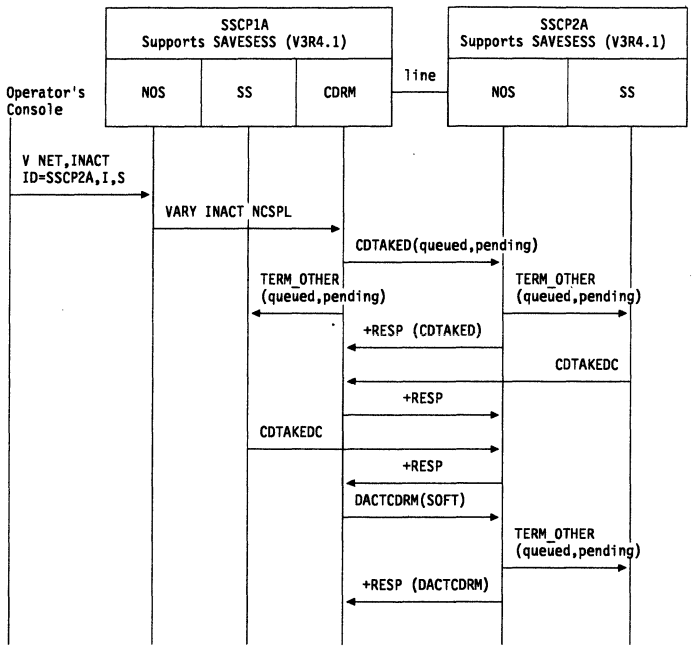


Figure 168. Deactivating a CDRM Without Affecting Active Sessions: Immediate MVS

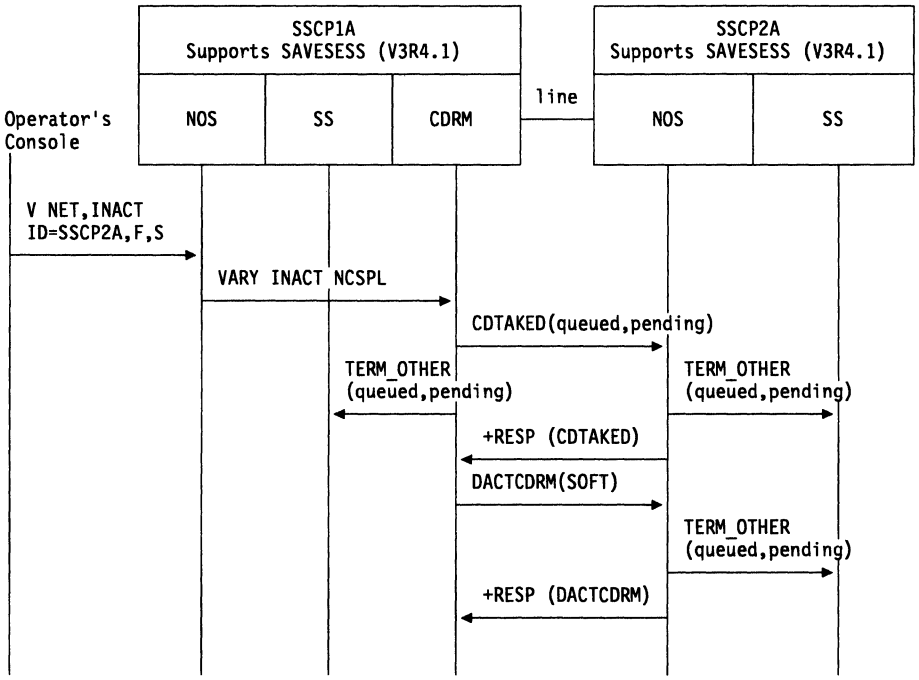


Figure 169. Deactivating a CDRM Without Affecting Active Sessions: Forced MVS

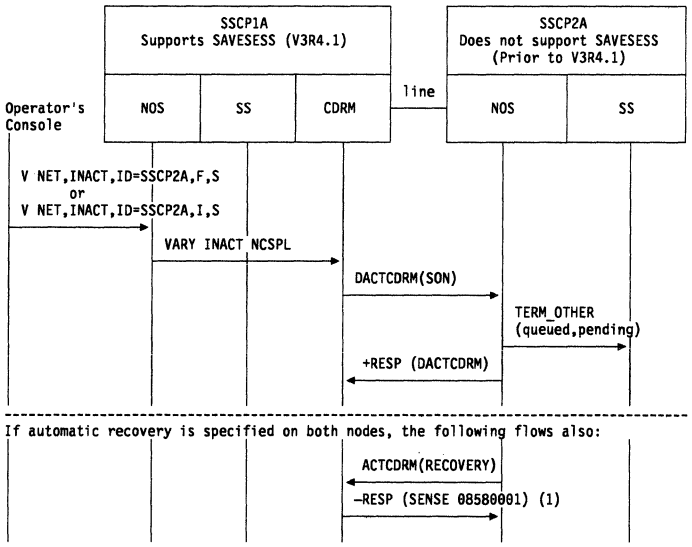


Figure 170. Deactivating a CDRM on a VTAM Level Prior to V3R4.1: Forced or Immediate
MVS

1. When automatic recovery is specified on both nodes, a CDRM V3R4.1 responds to ACTCDRM(RECOVERY) by sending a negative ACTCDRM response with sense code 08580001 indicating that it rejects the attempt to restart the session that was terminated using a nondisruptive deactivation request. Active LU-LU sessions remain active. The external CDRM in the migration SSCP becomes inactive with sessions, and the external CDRM in the V3R4.1 SSCP becomes inactive.

   See *VTAM Messages and Codes* for information on sense code 08580001.

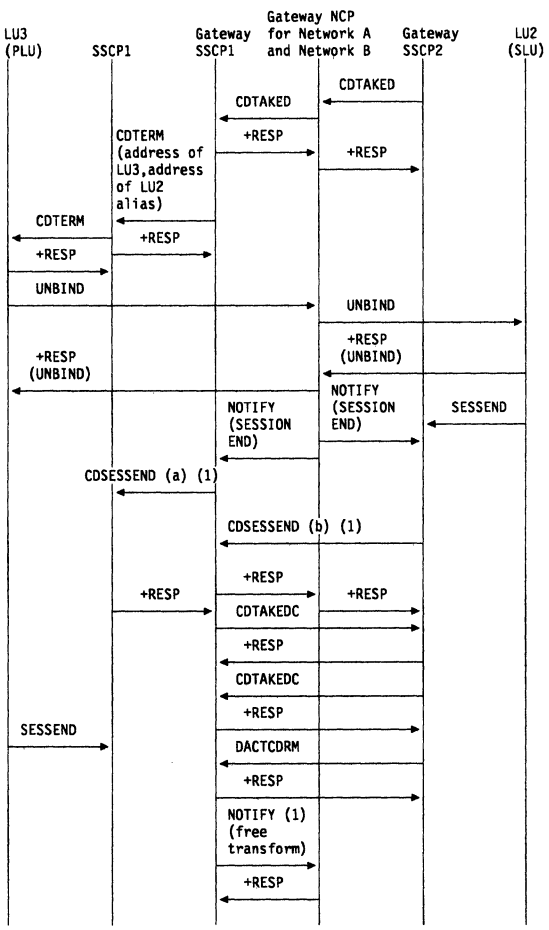


Figure 171. SSCP-SSCP Session Termination Causes LU-LU Sessions to Be Broken MVS VM

Notes:

- A cross-network LU-LU session exists between LU3 and LU2.
- This flow assumes that the gateway VTAM1 established the network address translation for the gateway VTAM1-to-gateway VTAM2 session with the RNAA RU. The NOTIFY to free the transform is sent only if the RNAA that established the address specified "retain address."
- (a) and (b) are used here to differentiate between similar request units.

1. For details of CDSESSEND processing, see the other flow diagrams listed in "Index of Deactivation and Session Termination Flows" on page 976.

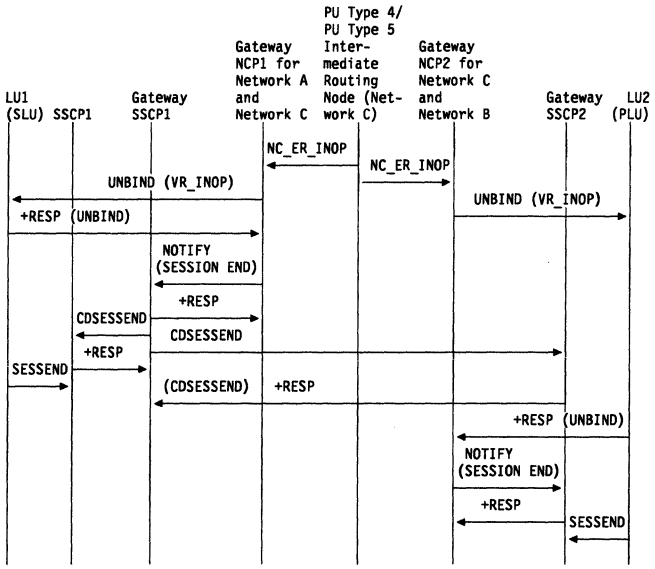


Figure 172. Route Failure in Intermediate Network Causes Termination of LU-LU Sessions
MVS VM

Note: An outage occurs on the route in Network C used by the LU1_LU2 session. ER_INOP reports the failure to gateway NCP1 and gateway NCP2.

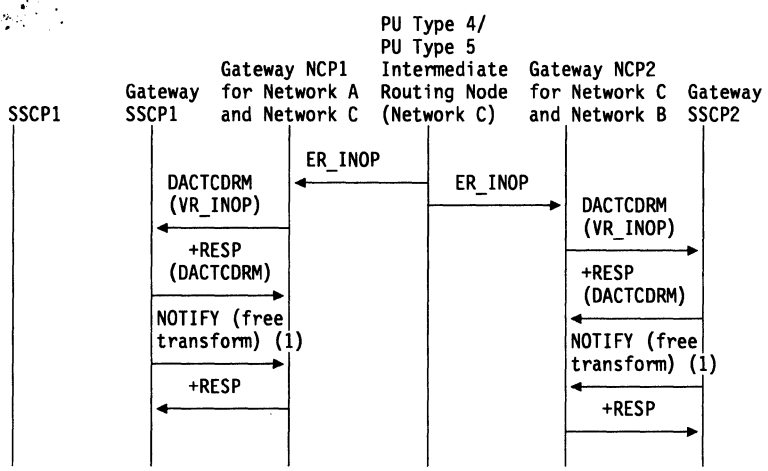


Figure 173. Route Failure in Intermediate Network Causes Termination of SSCP-SSCP Sessions MVS VM

Note: An outage occurs on the route in Network C used by the gateway VTAM1-to-gateway VTAM2 session. The failure is reported to gateway NCP1 and gateway NCP2 with an ER INOP RU.

1. The NOTIFY to free the transform is sent only if the RNAA that established the address specified "retain address."

Error Detection and Recovery and SSCP Management Services

Figure 174 on page 1001 through Figure 176 on page 1003 show the flow of requests and responses between the SSCP and logical and physical units to handle error recovery processing (ERP) and route Forward and Deliver RUs. Figure 177 on page 1003 through Figure 179 on page 1005 show the requests and responses between the NetView program, VTAM, the communications adapter and the local modem for LPDA-2 processing. Figure 180 on page 1006 through Figure 183 on page 1010 show the flow of requests and responses between the SSCP and logical and physical units to handle MVS extended recovery facility (XRF) session establishments and takeovers with USERVARs.

Index of Error Detection and Recovery and SSCP Management Services Flows

Table 69 lists the error detection and recovery and SSCP management services flows that are illustrated here.

Table 69. Index of Error Detection and Recovery and SSCP Management Services Flows

Flow	Page
Error recovery processing (ERP)	
Hard INOP	Figure 175 on page 1002
Soft INOP	Figure 174 on page 1001
LPDA-2 processing	
Unsolicited LPDA-2 test on permanent link error with two link segments	Figure 179 on page 1005
Unsolicited LPDA-2 test on thresholds reached for an LPDA-2 physical unit (PU) with one link segment	Figure 177 on page 1003
Unsolicited LPDA-2 test on thresholds reached for an LPDA-2 physical unit (PU) with two link segments	Figure 178 on page 1004
SSCP management services processing	
FORWARD and DELIVER Routing	Figure 176 on page 1003
XRF processing	
Secondary logical unit (LU) initiate with USERVAR (LOGON)	Figure 182 on page 1009
Third-party initiate (CLSDST PASS)	Figure 183 on page 1010
XRF primary and backup sessions, establishment of	Figure 180 on page 1006
XRF session switch (takeover)	Figure 181 on page 1008

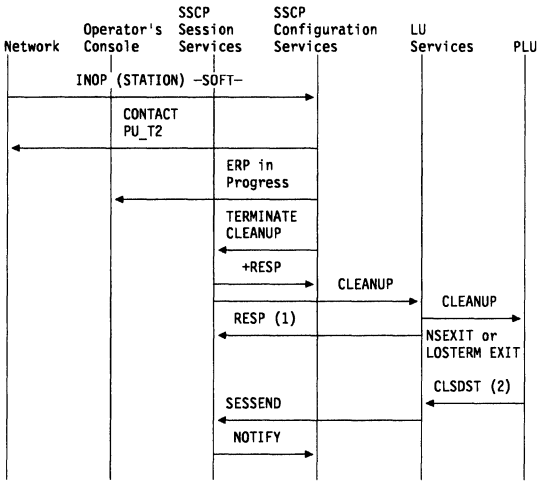


Figure 174. Error Recovery Processing: Soft INOP

1. If the NSEXIT exit routine is scheduled, LUS cleans up the session and sends a positive response to the cleanup request. If the LOSTERM exit routine is scheduled, LUS **does not** clean up the session, and it sends a negative response to the cleanup request.
2. CLSDST flows only if the LOSTERM exit routine is scheduled.

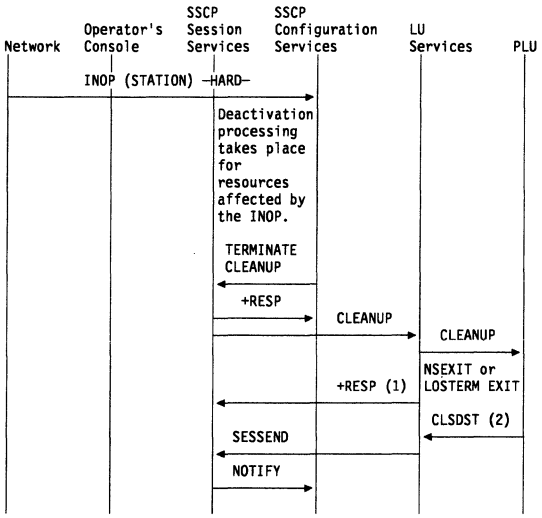


Figure 175. Error Recovery Processing: Hard INOP

1. If the NSEXIT exit routine is scheduled, LUS cleans up the session and sends a positive response to the cleanup request. If the LOSTERM exit routine is scheduled, LUS **does not** clean up the session, and it sends a negative response to the cleanup request.
2. CLSDST flows only if the LOSTERM exit routine is scheduled.

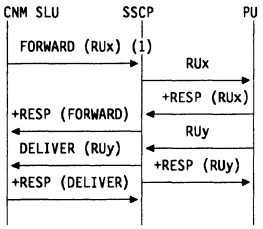


Figure 176. FORWARD and DELIVER Routing

1. RUx is a maintenance service RU.
2. Either RUy contains data in reply to RUx, or it is an unsolicited RU.

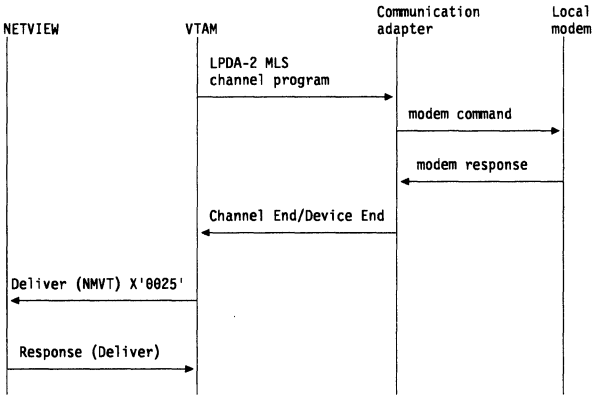


Figure 177. Unsolicited LPDA-2 Test on Thresholds Reached for an LPDA-2 PU with One Link Segment

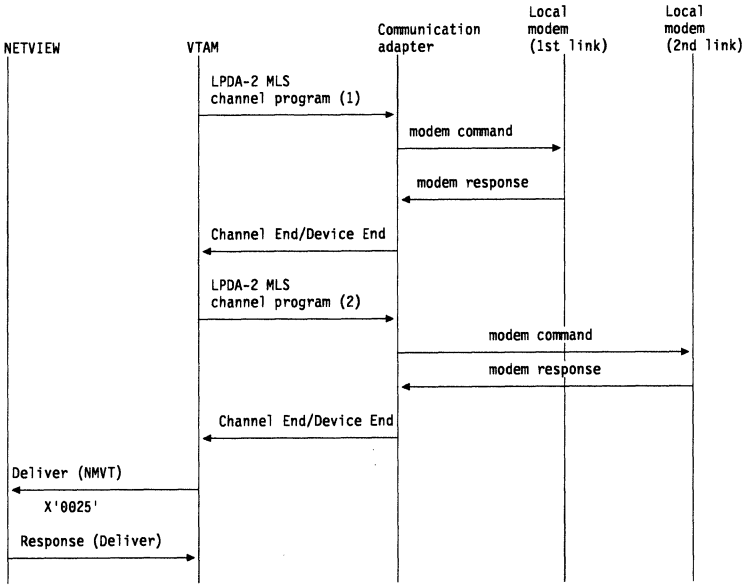


Figure 178. Unsolicited LPDA-2 Test on Thresholds Reached for an LPDA-2 PU with Two Link Segments

1. This MLS (modem and link status) command is for the first link segment.
2. This MLS command is for the second link segment.

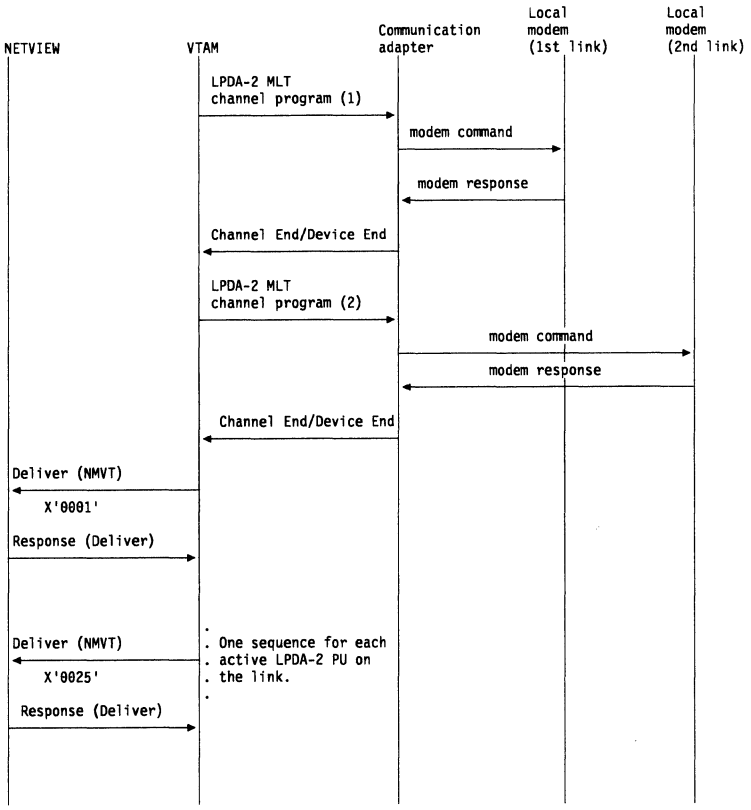


Figure 179. Unsolicited LPDA-2 Test on Permanent Link Error with Two Link Segments

1. This MLT (modem and link test) command is for the first link segment.
2. This MLT command is for the first active LPDA-2 PU on the second link segment.

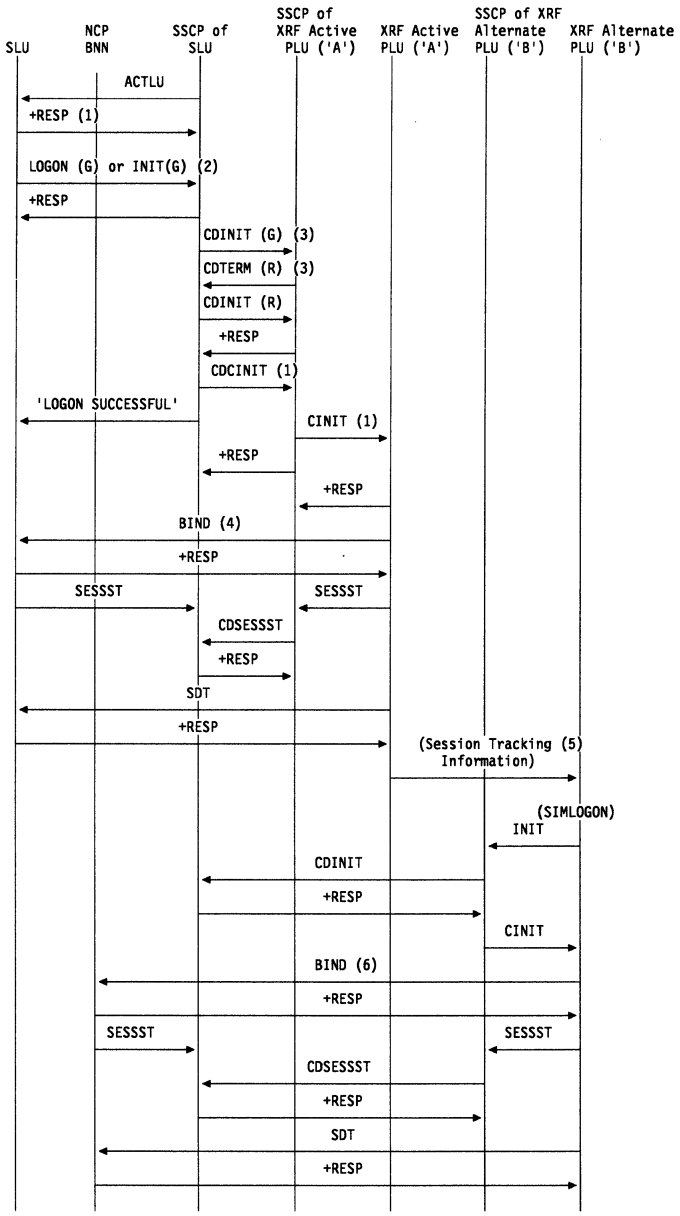


Figure 180. Establishment of XRF Primary and Backup Sessions MVS

G Represents a generic USERVAR name

R Represents a resolved USERVAR name

1. These RUs or responses contain control vectors which indicate whether or not the SLU is eligible for an XRF backup session.
2. A LOGON or INIT_SELF to an XRF application will generally use a generic (USERVAR) application name which is translated by VTAM to the real application name of the PLU that is currently the XRF active.
3. These RUs are used to translate the generic (USERVAR) name used in the LOGON or INIT_SELF to the real name of the application that is currently the XRF active. They are present only if the SLU's SSCP does not already know the current value of the USERVAR or if the USERVAR's type is VOLATILE.

4. The BIND includes a correlator ID which will be used to relate this primary session to the corresponding backup session.
5. The XRF Active application communicates information about the newly-established session, including the BIND correlator ID, to the XRF Alternate application. The XRF Alternate application uses this information when establishing the corresponding backup session.
6. The BIND includes a correlator ID which will be used to relate this backup session to the corresponding primary session. The BIND for the backup session terminates at the NCP; it is not visible to the actual device.

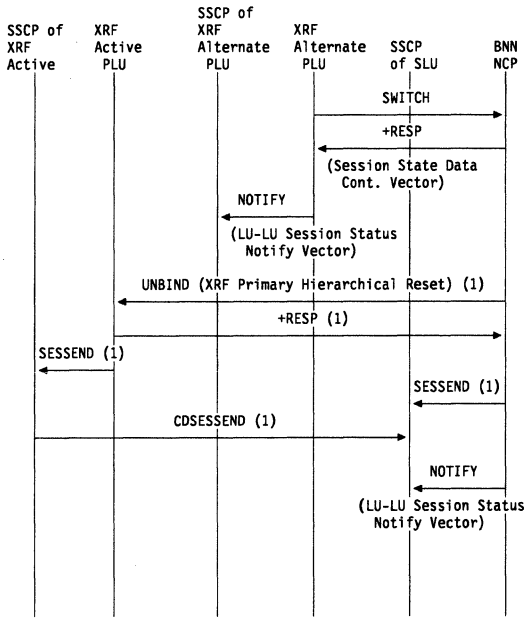


Figure 181. XRF Session Switch (Takeover) MVS

1. Not present if XRF active PLU is not operational.

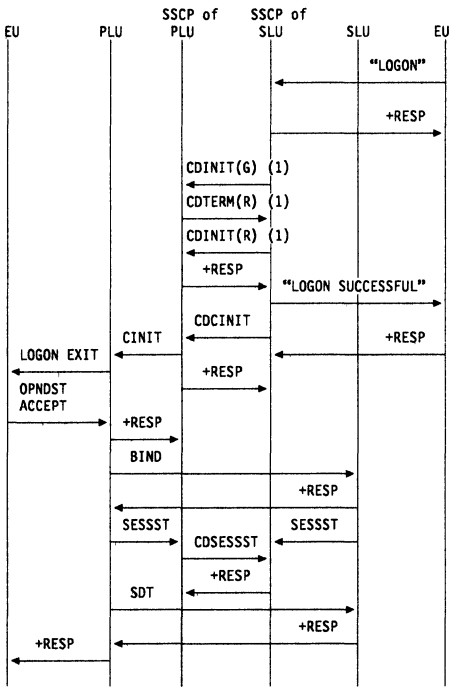


Figure 182. Secondary Logical Unit Initiate with USERVAR (LOGON) MVS

G Represents a generic USERVAR name

R Represents a resolved USERVAR name

1. These RUs are used to translate the generic (USERVAR) name used in the LOGON or INIT_SELF to the real name of the application that is currently the XRF active. They are present only if the SLU's SSCP does not already know the current value of the USERVAR or if the USERVAR's type is VOLATILE.

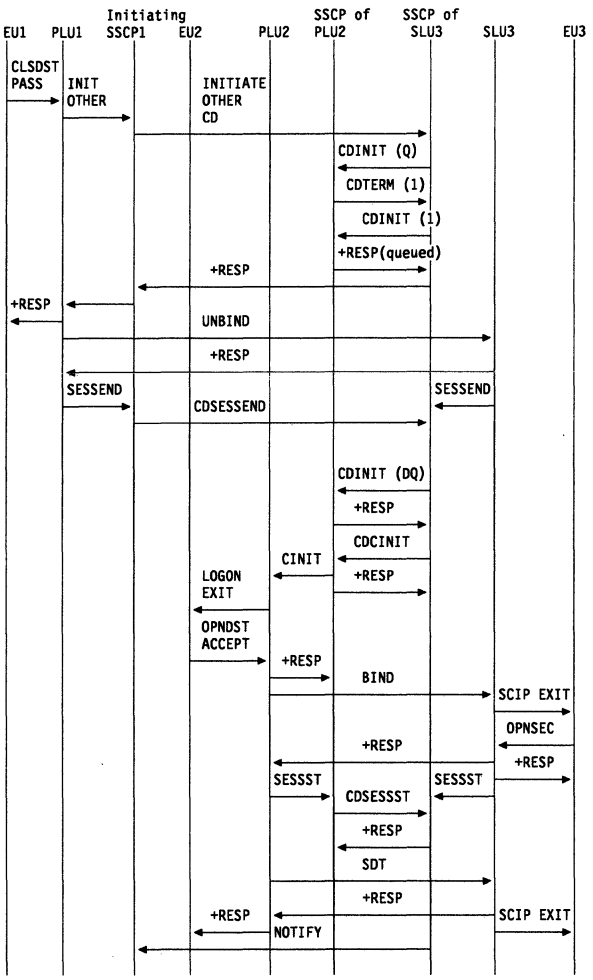


Figure 183. Third-Party Initiate (CLSDST PASS) MVS

1. These RUs are used to translate the generic (USERVAR) name used in the LOGON or INIT_SELF to the real name of the application. They are present only if the SLU's SSCP does not already know the current value of the USERVAR or if the USERVAR's type is VOLATILE.

Dial and Answer for X.21 Switched Lines (VSE)

Figure 184 illustrates the flow of requests and responses between the SSCP and X.21 switched lines attached through a communication adapter for dial and answer processing.

Index of X.21 Switched Lines Flows

Table 70 lists the X.21 Switched Lines flows illustrated here.

Table 70. Index of X.21 Switched Lines Flows

Flow	Page
Dial and answer for X.21 switched lines	Figure 184

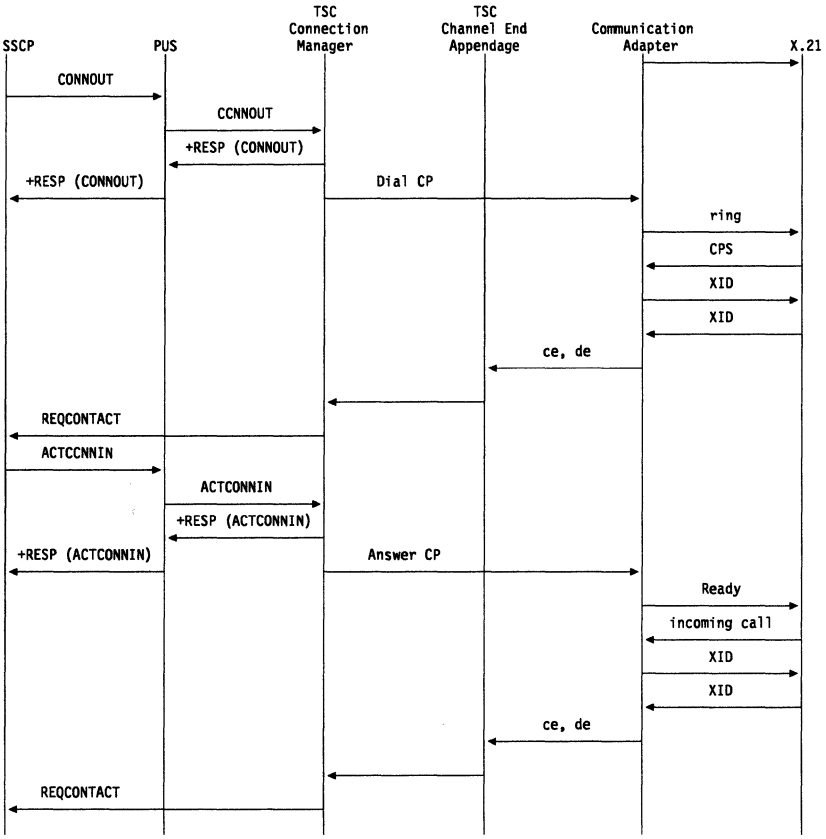


Figure 184. Dial and Answer for X.21 Switched Lines

Appendix F. APPN Flows

This appendix describes the flows between APPN end nodes, network nodes, interchange nodes, and the subarea network. The flow diagrams are divided into the following categories:

Table 71 on page 1014 lists all the APPN flows illustrated here.

Table 71 (Page 1 of 3). Index of APPN Flows

Table 71 (Page 2 of 3). Index of APPN Flows

Directory Services Flow	Page
Locate resource	
APPN and Subarea Network	Figure 202 on page 1048
APPN network, complex	Figure 199 on page 1040
Complex APPN network using more than one CDS	Figure 200 on page 1042
CP-CP session terminates	Figure 204 on page 1054
CP network broadcast initiation	Figure 197 on page 1037
EN to NN	Figure 193 on page 1032
EN to NN to EN	Figure 194 on page 1033
EN to NN to NN to NN	Figure 196 on page 1036
EN to NN to subarea network	Figure 198 on page 1039
EN to NN to two ENs	Figure 195 on page 1034
NN receives network broadcast request	Figure 205 on page 1055
NNS of the OLU is at pre-V4R2 level	Figure 201 on page 1045
SLU-initiated session	Figure 203 on page 1052
Register resource	
EN to NN to CDS	Figure 191 on page 1029
With error recovery	Figure 192 on page 1031
LU-LU Sessions Flow	**Page**
APPN network...NNS--EN (PLU)	
SLU-initiated, no queueing	Figure 213 on page 1066
SLU-initiated, queued by the PLU	Figure 214 on page 1067
APPN network...NNS--EN(SLU)	
PLU-initiated, no queueing	Figure 211 on page 1064
PLU-initiated, queued by the SLU	Figure 212 on page 1065
APPN network (PLU)...ICN==SA(SLU), PLU-Initiated	
Directed search without required precompututed RSCV	Figure 236 on page 1090
No queueing	Figure 235 on page 1089
Queued by SLU	Figure 238 on page 1092
Search-only flow transformed into a DSRLST	Figure 234 on page 1088
USERVAR resolution required	Figure 237 on page 1091
APPN network (PLU)...ICN==VR-based TG==ICN...APPN network (SLU)	
PLU-initiated	Figure 242 on page 1097
APPN network (SLU)...ICN==SA(PLU)	
Autologon, PLU not available initially	Figure 241 on page 1096
SLU-initiated, no queueing	Figure 239 on page 1093
SLU-initiated, queued by the PLU	Figure 240 on page 1094

Table 71 (Page 3 of 3). Index of APPN Flows

CLSDST PASS; SLU is single-session capable	
From APPN to subarea	Figure 231 on page 1085
Through APPN	Figure 230 on page 1084
EN-NN-EN, PLU-initiated, no queueing (including BIND flows for intermediate network node)	Figure 232 on page 1086
EN (PLU)--NNS...APPN network	
PLU-initiated, no queueing	Figure 206 on page 1059
PLU-initiated, queued by the PLU	Figure 207 on page 1060
PLU-initiated, queued by the SLU	Figure 208 on page 1061
EN (SLU)--NNS...APPN network	
SLU-initiated, no queueing	Figure 209 on page 1062
SLU-initiated, queued by the PLU	Figure 210 on page 1063
Intermediate Network Node (INN) BIND. The LOCATE did not go through this node.	Figure 233 on page 1087
SA (PLU)==ICN...APPN network (SLU)	
DSRLIST transforming into PLU-initiated, search-only	Figure 215 on page 1068
PLU-initiated, no queueing	Figure 216 on page 1069
PLU-initiated, queued by the SLU	Figure 218 on page 1071
PLU-initiated, USERVAR resolution required	Figure 217 on page 1070
SA (SLU)==ICN....APPN network (PLU)	
Autologon, PLU not available initially	Figure 221 on page 1076
SLU-initiated, no queueing	Figure 219 on page 1073
SLU-initiated, queued by the PLU	Figure 220 on page 1074
Lu-LU Session Flow	**Page**
Session release request	
SA(PLU)==ICN...APPN network(SLU)	Figure 228 on page 1083
SA(SLU)==ICN...APPN network(PLU)	Figure 229 on page 1083
Session termination, forced	
SA(PLU)==ICN...APPN Network(SLU), pending active session. PLU is accessible without going into APPN.	Figure 225 on page 1080
SA(PLU)==ICN...APPN Network(SLU), queued session	Figure 226 on page 1081
SA(PLU)==ICN...APPN Network(SLU), queued session. PLU is accessible without going into APPN.	Figure 227 on page 1082
SA(SLU)==ICN...APPN Network(PLU), pending active session	Figure 224 on page 1079
Session termination, orderly	
SA(PLU)==ICN...APPN Network(SLU), active session	Figure 222 on page 1077
APPN Network(PLU)...ICN==SA(SLU), active session	Figure 223 on page 1078

Many abbreviations are shown at the top of the flow diagrams. The following list gives the meaning of those abbreviations:

APPC Advanced-program-to-program communication program
BF Boundary function
CDS Central Directory Server
CP Control point
DLUR Dependent logical unit requestor
DLUS Dependent logical unit server

EN	End node
LU	Logical unit
NN	Network node
NNS	Network node server
PLU	Primary logical unit
PU	Physical unit
PUNS	Physical unit services
SLU	Secondary logical unit
SSCP	System services control point
TP	Transaction program
TSC	Transmission subsystem component

CP-CP Session Flows

This section illustrates communication protocols between nodes in a mixed APPN and subarea network. Use these flows as guidelines to help analyze and isolate network problems caused by unexpected network events, such as protocol violations.

Index of CP-CP Session Flows

Table 72 lists the CP-CP session flows illustrated here.

Table 72. Index of CP-CP Session Flows

Flow	Page
Activating a CP-CP conloser session	Figure 186 on page 1022
Activating a CP-CP contention winner session	Figure 185 on page 1019
Activating an APPN Host-to-Host Channel	Figure 190 on page 1027
Activating a leased APPN Node type 2.1	Figure 189 on page 1026
Host CP initiating deactivation of CP-CP session	Figure 187 on page 1024
Remote node initiating deactivation of CP-CP session	Figure 188 on page 1025

Activating a CP-CP Contention Winner Session

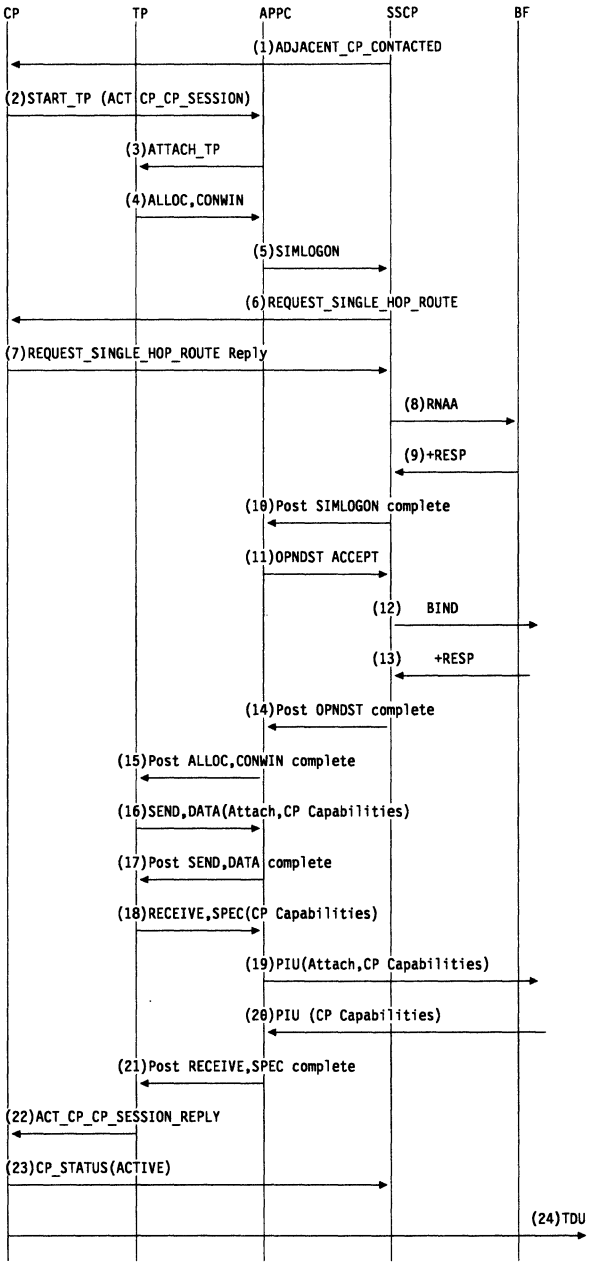


Figure 185. CP-CP Contenion Winner Session Activation

1. Configuration services sends an ADJACENT_CP_CONTACTED signal to the CP when the first link supporting CP-CP sessions is activated with an adjacent CP.
2. The CP sends a START_TP signal to the APPC PAB to request initiation of the Request CP Capabilities Transaction Program (TP). An ACT_CP_CP_SESSION request is queued to the START_TP signal for processing by the Request CP Capabilities TP. The ACT_CP_CP_SESSION initiates the activation of a contention winner CP-CP session with an adjacent node.

3. APPC sends an ATTACH_TP AMU for the Request CP Capabilities TP to TP Services.
4. The Request CP Capabilities TP issues an APPCCMD CONTROL=ALLOC, QUALIFY=CONWIN macroinstruction to allocate a conversation between the TP and the partner TP and a contention winner session between the local LU and the remote LU.
5. APPC issues a VTAM SIMLOGON macroinstruction to initiate a session in which APPC acts as the PLU.
6. Subarea session services sends a REQUEST_SINGLE_HOP_ROUTE signal to the CP to request the least-weight single-hop route from the origin to the destination.
7. The CP sends to subarea session services the information requested in a REQUEST_SINGLE_HOP_ROUTE_REPLY.
8. RNAA flows if the CP-CP session is being set up over a type 2.1 link and a network address is needed. If the CP-CP session is being set up over a VR-based transmission group, CDINIT format 5 is sent to the session partner to get a network address.
9. The response is received from the boundary function.
10. LUS posts the SIMLOGON complete.
11. APPC issues an OPNDST ACCEPT macroinstruction to continue establishment of a session between APPC in this node (acting as the PLU) and APPC in the adjacent node (acting as the SLU).
12. The BIND for the contention winner session is transmitted to the adjacent node.
13. The BIND response for the contention winner session is received from the adjacent node.
14. LUS posts the OPNDST ACCEPT complete.
15. APPC posts the APPCCMD CONTROL=ALLOC,QUALIFY=CONWIN complete, supplying the Request CP Capabilities TP a conversation ID and a contention winner conversation group ID (CGID).
16. The Request CP Capabilities TP issues an APPCCMD CONTROL=SEND, QUALIFY=DATA macroinstruction to initiate the sending of this CP's CP Capabilities data to the adjacent CP.
17. APPC posts the APPCCMD CONTROL=SEND,QUALIFY=DATA instruction complete, indicating that the output buffer has been filled with this CP's CP Capabilities data.
18. At the request of the Request CP Capabilities TP, the Receive and Check CP Capabilities TP issues an APPCCMD CONTROL=RECEIVE, QUALIFY=SPEC macroinstruction to cause the transmission of this CP's CP Capabilities data to the adjacent CP, and to initiate the receiving of CP Capabilities data from the adjacent CP.
19. APPC sends to the adjacent node a PIU with this CP's CP Capabilities data. The PIU also carries a request that TP services in the adjacent node attach its CP Capabilities TP.
20. APPC receives from the adjacent node a PIU containing the adjacent CP's CP Capabilities.

21. APPC posts the APPCCMD CONTROL=RECEIVE,QUALIFY=SPEC macroinstruction complete.
22. The Request CP Capabilities TP responds to the successful completion of the RECEIVE macroinstruction by sending to the CP an ACT_CP_CP_SESSION_REPLY, which contains both the contention winner CGID and the CP Capabilities data just received from the adjacent CP.
23. The CP sends a CP_STATUS(ACTIVE,BOTH) signal to configuration services if both the contention winner and conloser CP-CP sessions are both active at this point.
24. If both CPs are network nodes, a topology database update (TDU) will flow once the contention winner session is active. The TDU is used to update the partner regarding changes to network topology that have occurred since the two CPs were last in session.

    A TDU will also flow over the contention winner session from a VTAM end node to its network node server, if the network node server is a VTAM. This TDU carries information on changes that have occurred to the end node's connections since the end node and the network node server were last in session.

Activating a CP-CP Conloser Session

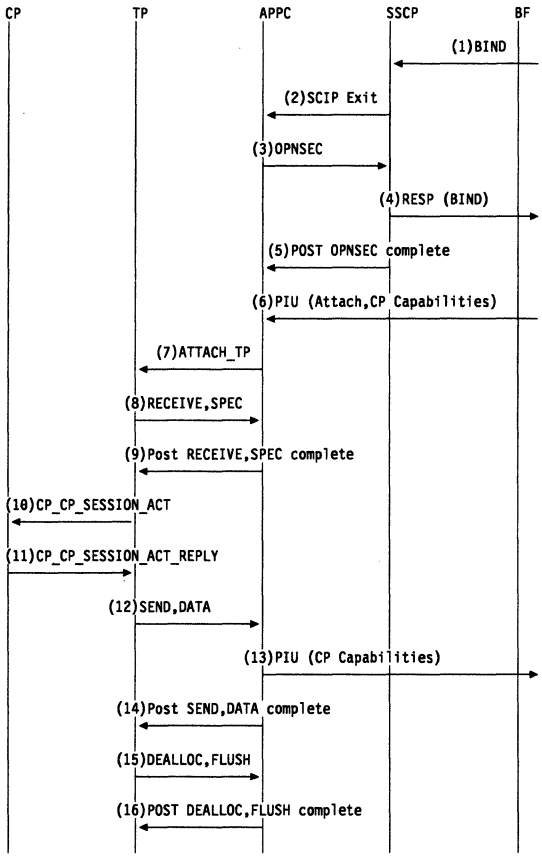


Figure 186. CP-CP Conloser Session Activation

1. The BIND for the conloser CP-CP session is received from the adjacent node.
2. The BIND drives the SCIP exit routine.
3. The SCIP exit causes APPC to issue an OPNSEC macroinstruction to establish a session between APPC (operating as the SLU) and the PLU that sent the BIND.
4. The OPNSEC macroinstruction causes a BIND response to be sent to the adjacent node.
5. The OPNSEC is posted complete.
6. APPC receives from the adjacent node a PIU containing the adjacent CP's CP Capabilities. The PIU also contains an FMH-5 specifying that the CP Capabilities Transaction Program (TP) is to be attached.
7. APPC sends an ATTACH_TP AMU to TP Services to request the attachment of the CP Capabilities TP.
8. After attachment, the CP Capabilities TP issues an APPCCMD CONTROL=RECEIVE,QUALIFY=SPEC instruction to receive the adjacent CP's CP Capabilities data.
9. APPC posts the APPCCMD CONTROL=RECEIVE,QUALIFY=SPEC macroinstruction complete.

10. The CP Capabilities TP builds and sends to the CP a CP_CP_SESSION_ACT signal, containing the conloser Conversation Group Identifier (CGID) and the adjacent CP's CP Capabilities data.
11. The CP sends to the CP Capabilities TP a CP_CP_SESSION_ACT_REPLY. It contains the CP Capabilities of this node and also indicates whether or not SSC was able to successfully process the CONLOSER activation request.
12. After receiving the CP_CP_SESSION_ACT_REPLY, the CP Capabilities TP issues an APPCCMD CONTROL=SEND,QUALIFY=DATA to send this CP's CP Capabilities to the adjacent CP.
13. A PIU with this CP's CP Capabilities is transmitted to the adjacent CP.
14. When the transmission is complete, the APPCCMD CONTROL=SEND, QUALIFY=DATA macroinstruction is posted complete by APPC.
15. The CP Capabilities TP issues an APPCCMD CONTROL=DEALLOC, QUALIFY=FLUSH macroinstruction to flush the local LU's send buffer and deallocate the conversation normally.
16. APPC posts the APPCCMD CONTROL=DEALLOC,QUALIFY=FLUSH macroinstruction back upon completion.

Host CP Initiating Deactivation of CP-CP Session

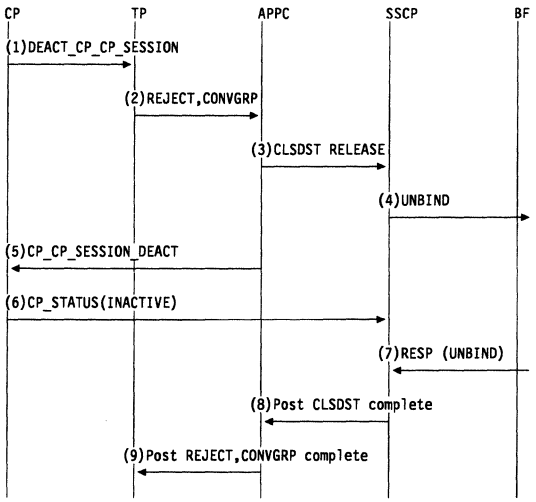


Figure 187. Host CP Initiating Deactivation of CP-CP Session

1. The CP sends a DEACT_CP_CP_SESSION to the SEND_REJECT_CONVGRP Transaction Program (TP). The CP sends this signal when it needs to have a particular CP-CP session with an adjacent node unbound.
2. The SEND_REJECT_CONVGRP TP issues an APPCCMD CONTROL=REJECT, QUALIFY=CONVGRP macroinstruction to deactivate the specified session.
3. APPC, responding to the APPCCMD CONTROL=REJECT,QUALIFY=CONVGRP macroinstruction, issues a CLSDST RELEASE macroinstruction to terminate the session.
4. APPC also builds and sends a CP_CP_SESSION_DEACT for the specified session to the CP. This signal is sent by APPC to notify the CP that a CP-CP session outage is detected. It contains the session type and CGID of the CP-CP session that the outage is detected on.
5. The CLSDST RELEASE causes SSCP to send an UNBIND for the particular session to the partner LU.
6. The CP sends a CP_STATUS(INACTIVE) signal for the specified session to the SSCP.
7. A response to the UNBIND is received by the SSCP from the partner LU.
8. Having received the UNBIND response, the SSCP posts complete the CLSDST RELEASE.
9. APPC posts the TP's APPCCMD CONTROL=REJECT,QUALIFY=CONVGRP macroinstruction complete.

Remote Node Initiating Deactivation of CP-CP Session

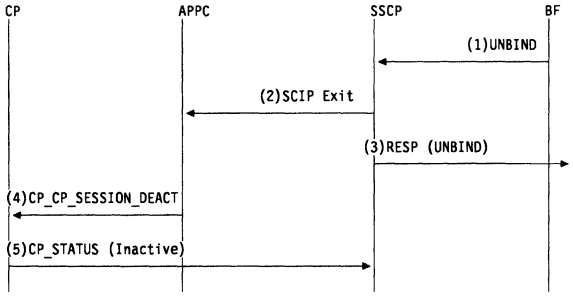


Figure 188. Remote Node Initiating Deactivation of CP-CP Session

1. An UNBIND is received from the adjacent node for a specific CP-CP session. The UNBIND carries a sense code associated with the session outage.
2. The UNBIND drives the SCIP exit routine.
3. The SSCP sends an UNBIND response to the adjacent node.
4. APPC sends a CP_CP_SESSION_DEACT to the CP for the session specified in the UNBIND. The CP_CP_SESSION_DEACT carries the sense code originally carried by the UNBIND.
5. The CP notifies the SSCP that the specified session is now inactive.

Activating a Leased APPN Node Type 2.1

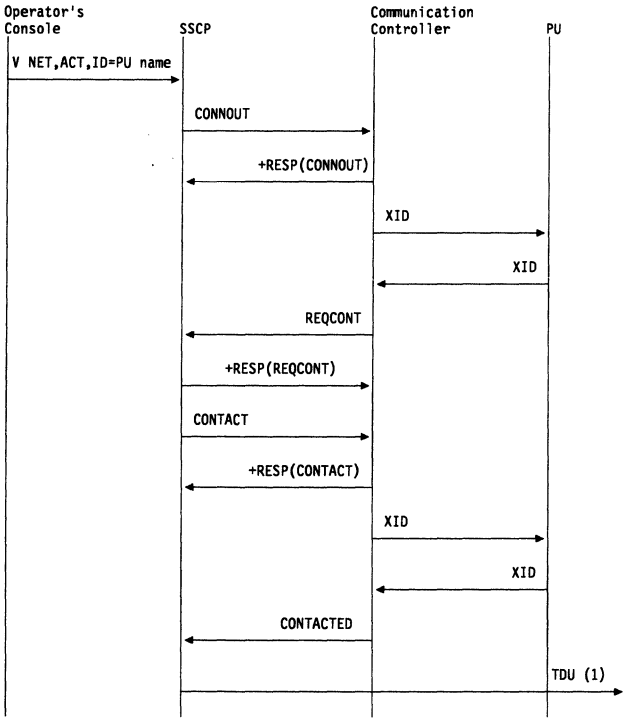


Figure 189. Activating a Leased APPN Node Type 2.1

1. If this is a network node, a topology database update (TDU) is sent to all CP-CP session partners, informing them that this new APPN connection is available. If this is an end node, and the network node server is a VTAM, a TDU will be sent to the server, informing it of the new connection.

Leased APPN PUs, as opposed to non-APPN PUs, have an additional CONNOUT, XID, and REQCONT flow. This flow allows transmission group (TG) negotiation during pre-negotiation XID exchanges.

Activating an APPN Host-to-Host Channel

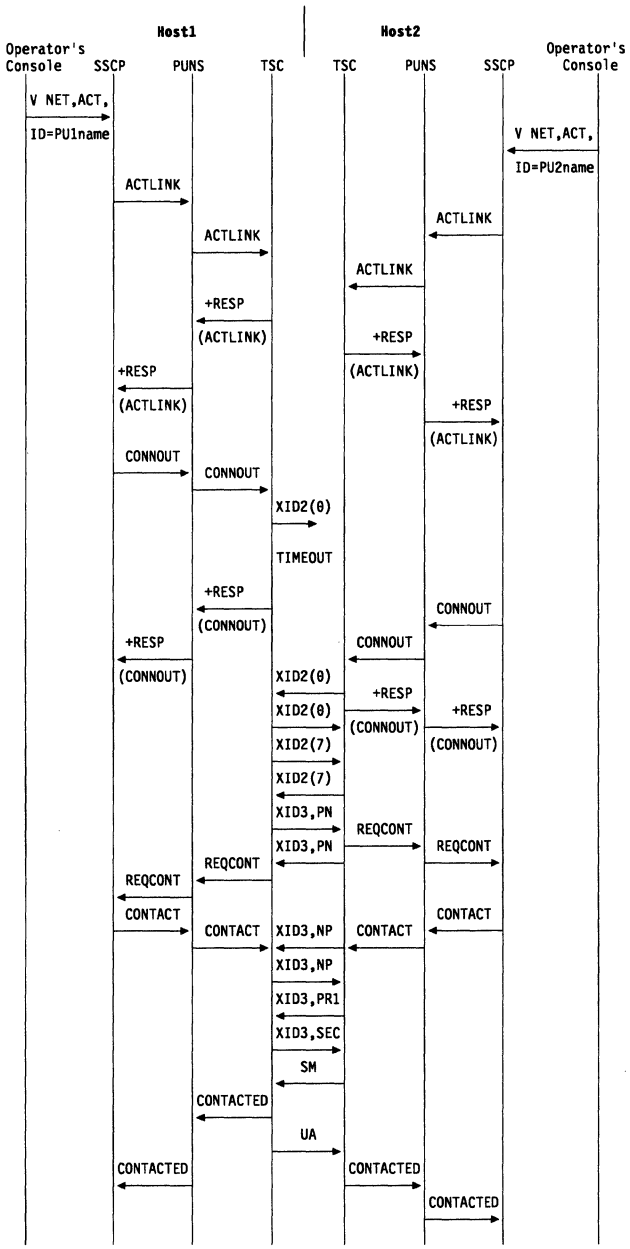


Figure 190. Activating an APPN Host-to-Host Channel

Note: XID2 Type 0 and Type 7 are sent across each subchannel associated with an APPN host-to-host channel connection. The XID3s are sent by choosing one of the available write subchannels for transmission.

Directory Services Flows

To understand the following flows, it is helpful to understand the concept of resource registration and the implications of the values coded on the REGISTER operand. For information on resource registration and registering application programs, see "Registering Resources" in *VTAM Network Implementation Guide*. For information about coding the REGISTER operand, see "Registering Application Programs" in the *VTAM Resource Definition Reference*.

In these directory services flows, assume the following, unless stated otherwise:

- Sessions are initiated by the primary logical unit (PLU).
- The flows illustrate search-only requests.

Index of Directory Services Flows

Table 73 lists the directory services flows illustrated here.

Table 73. Index of Directory Services Flows

Flow	Page
Locate resource	
APPN and Subarea Network	Figure 202 on page 1048
APPN network, complex	Figure 199 on page 1040
Complex APPN network using more than one CDS	Figure 200 on page 1042
CP-CP session terminates	Figure 204 on page 1054
CP network broadcast initiation	Figure 197 on page 1037
EN to NN	Figure 193 on page 1032
EN to NN to EN	Figure 194 on page 1033
EN to NN to NN to NN	Figure 196 on page 1036
EN to NN to subarea network	Figure 198 on page 1039
EN to NN to two ENs	Figure 195 on page 1034
NN receives network broadcast request	Figure 205 on page 1055
NNS of the OLU is at pre-V4R2 level	Figure 201 on page 1045
SLU-initiated session	Figure 203 on page 1052
Resource registration flows	
EN to NN to CDS	Figure 191 on page 1029
With error recovery	Figure 192 on page 1031

Register Resource Flows

Figure 191 and Figure 192 on page 1031 show the process of registering resources.

Resource Registration: EN to NN to CDS

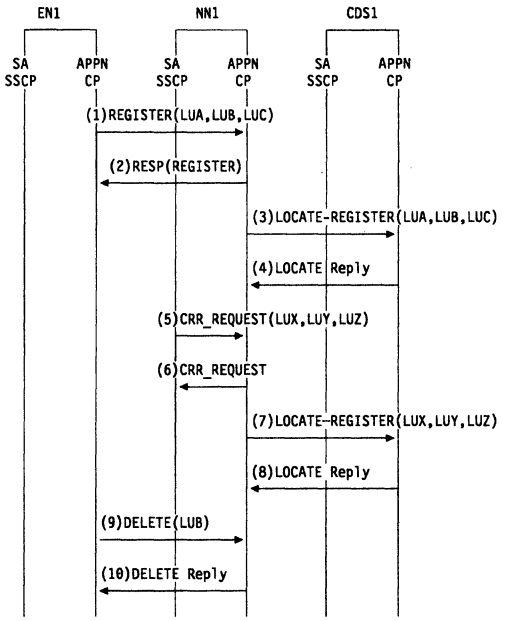


Figure 191. Resource Registration: EN to NN to CDS

Note: In this figure, the end node registers its resources to NN1:

- LUA
- LUB
- LUC.

NN1 owns the following resources:

- LUX
- LUY
- LUZ.

NN1 registers the following resources to the central directory server (CDS):

- LUA
- LUB
- LUC
- LUX
- LUY
- LUZ.

1. The VTAM operator in the end node activates a major node containing LUA, LUB, and LUC. The end node sends a registration request to NN1.
2. NN1 adds entries to the directory database and then sends a registration reply to EN1.

3. NN1 sends a central registration request to CDS1. The registration request travels with a LOCATE GDS variable because the central directory server can be several nodes away.
4. CDS1 returns a reply to NN1.
5. The VTAM operator activates a major node containing applications X, Y, and Z. The subarea SSCP notifies the APPN control point (CP) of resources owned by NN1 that should be centrally registered. (CRR stands for central resource registration.)
6. APPN CP sends an immediate reply to the subarea SSCP.
7. NN1 sends a central registration request to CDS1.
8. CDS1 returns a reply to NN1.
9. The end node sends a DELETE request to NN1.
10. NN1 removes LUB from its directory database and returns a reply to the end node. NN1 does not forward the DELETE request to the central directory server.

Resource Registration with Error Recovery

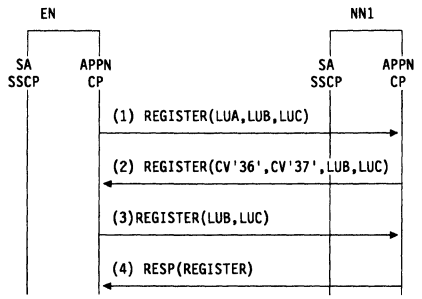


Figure 192. Resource Registration with Error Recovery

1. The VTAM operator in the end node activates a major node containing LUA, LUB, and LUC. The end node sends a registration request to NN1.
2. The network node begins adding resources to its directory database. The network node successfully adds LUA. However, it encounters a problem and cannot continue adding resources to the directory database. The network node sends a negative reply to the end node to indicate which resource the network node was trying to add when it encountered the problem. The CV'36' indicates the sense code. The CV'37' indicates where the network node stopped adding to its directory database.
3. The end node tries again to register those resources that were not successfully registered before.
4. The network node successfully adds to the directory database and returns a reply.

Locate Resource Flows

Figure 193 through Figure 205 on page 1055 show the search process.

Locate Resource: EN to NN

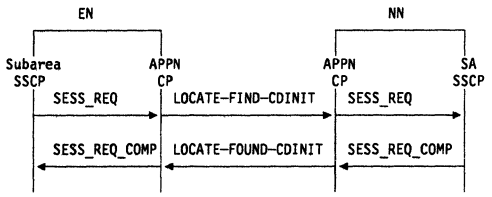


Figure 193. Locate Resource: EN to NN

Locate Resource: EN to NN to EN

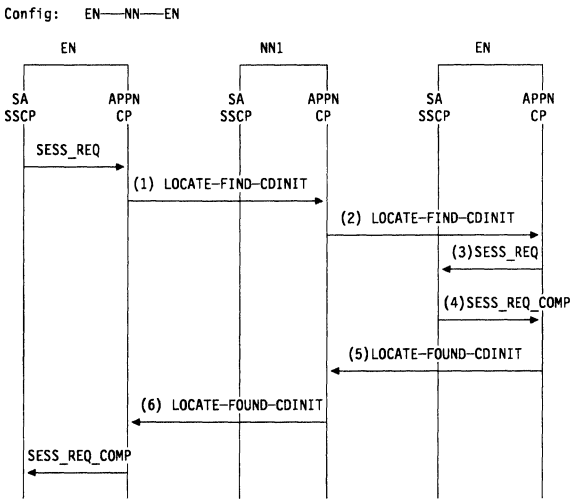


Figure 194. Locate Resource: EN to NN to EN

Note: Nodes are connected by CP-CP sessions only.

1. The end node sends a search request for a target resource to the network node server. As the network node server of the originating LU, NN1 looks for the target resource in the directory database. NN1 has knowledge in the directory that the target resource resides on a served end node.
2. NN1 sends a search request to the end node.
3. The APPN control point (CP) sends a SESS_REQ signal to the SSCP.
4. The SSCP sends a SESS_REQ_COMP signal to the APPN CP, indicating that the target resource is located.
5. The end node sends a LOCATE reply to the network node server.
6. NN1 sends a LOCATE reply to the end node.

Locate Resource: EN to NN to Two ENs

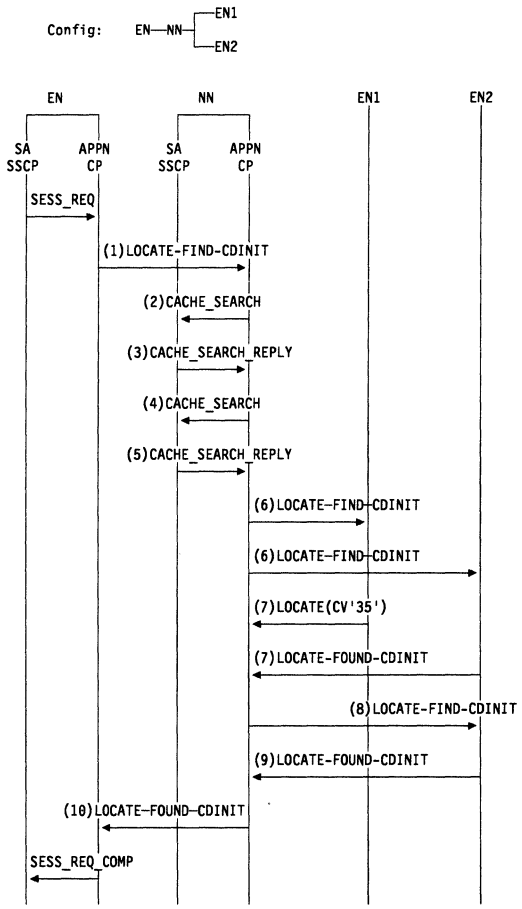


Figure 195. Locate Resource: EN to NN to Two ENs

Note: Nodes are connected by CP-CP sessions only.

1. The end node sends a search request for a target resource to the network node server. The network node server looks for the target resource in the directory database. The end node that owns the resource did not register its resources.
2. The APPN control point (CP) requests that the subarea SSCP check for information about the location of the resource.
3. The subarea SSCP replies that the target resource is not known.
4. Because NN doesn't have the target resource in either its APPN or subarea directories, it initiates a resource discovery search. The research discovery search starts at the beginning of the search logic with a "generic" request, which is not linked to the original OLU. Because the resource discovery search starts at the beginning of the search logic, another CACHE_SEARCH is performed.
5. The subarea SSCP replies that the target resource is not known.
6. The network node server performs a domain broadcast by sending the search request to all served end nodes that indicate on the CP_CAPABILITIES exchange that they are to be searched on domain broadcast.

7. Each end node that receives the request replies to the search request. EN1 replies that the resource is not found. EN2 replies that it owns the resource.
8. Because the resource discovery search found the resource, the NN sends a search to the target, containing the original session-specific information.
9. The target is found.
10. The network node server replies to the end node.

Locate Resource: EN to NN to NN to NN

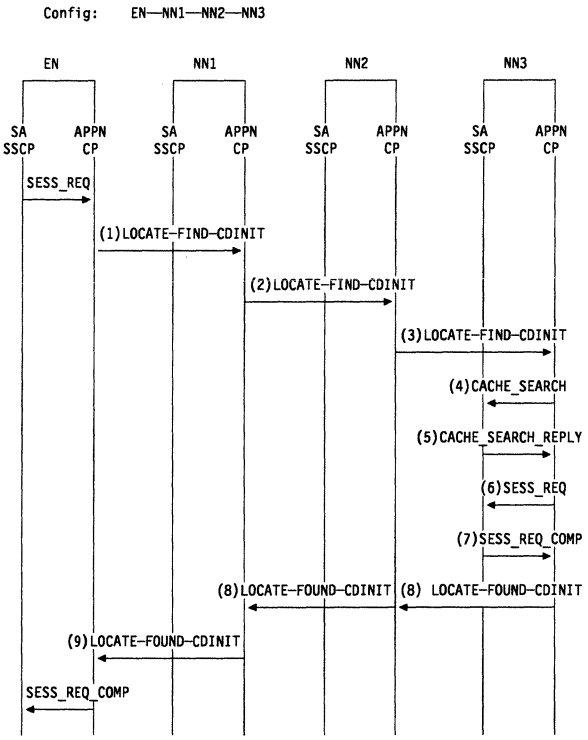


Figure 196. Locate Resource: EN to NN to NN to NN

Note: Nodes are connected by CP-CP sessions only.

1. The end node sends a search request for a target resource to the network node server. The network node looks for the target resource in the directory database. The network node has knowledge in the directory that the target resource resides on NN3.
2. The network node sends a directed search request to NN3. Because NN1 does not have direct CP-CP sessions with NN3, NN1 sends the directed search request to NN3 through NN2.
3. NN2 is not the destination of the directed search; therefore, NN2 simply forwards the request to NN3.
4. APPN control point (CP) sends a request for information to the subarea SSCP.
5. The subarea SSCP replies that the target resource can be found in the subarea network.
6. APPN control point (CP) sends a SESS_REQ signal to the subarea SSCP.
7. The subarea SSCP sends a SESS_REQ_COMP signal to the APPN CP, indicating that the target resource is located.
8. NN3 sends a LOCATE reply to NN2, which forwards the reply back to NN1.
9. NN1 sends a LOCATE reply to the end node.

Locate Resource: CP Network Broadcast Initiation

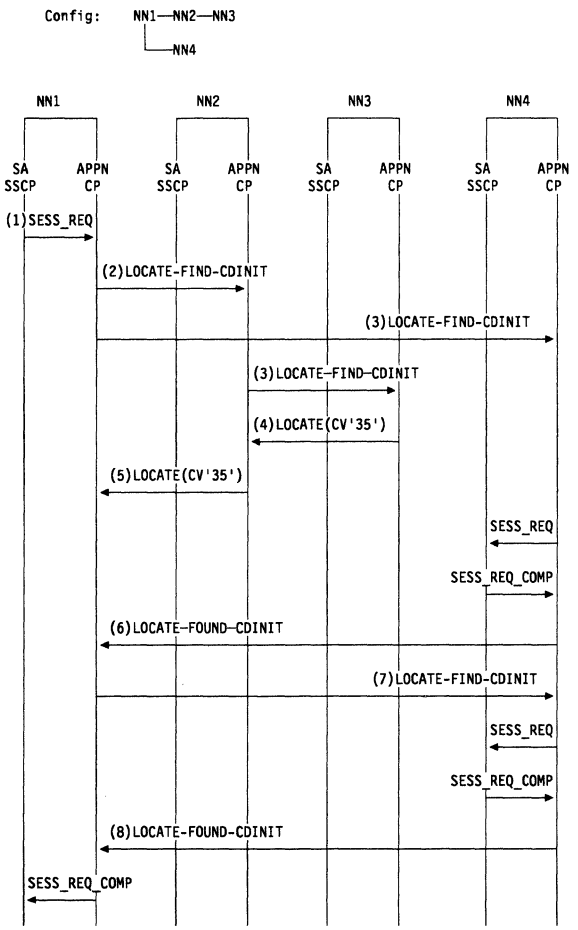


Figure 197. Locate Resource: CP Network Broadcast Initiation

1. A resource in NN1 requests a search for a resource on NN4. As network node server of the originating LU, NN1 looks for the target resource in the directory database. NN1 has no knowledge of the location of the target resource. NN1 initiates a resource discovery search for the target, which contains no session-specific information.
2. NN1 has no APPN-domain end nodes, therefore no domain broadcast occurs. There is no central directory server in the network; therefore, NN1 sends a broadcast search request to every network node with which NN1 has CP-CP sessions.
3. Each network node that receives the network broadcast request forwards the request to every network node with which it has CP-CP sessions. (It does not forward the request to the node from which it received the broadcast request.) Those nodes then begin searching their respective domains for the target resource. (For broadcast-specific flows for those nodes, see Figure 205 on page 1055.)
4. NN2 searches its domain for the resource. (Flows are not shown; see Figure 205 on page 1055) NN2 does not locate the resource. However, NN2 does not reply to NN1 until it has received a reply from all of the nodes to

which it forwarded the request. NN3 does not locate the resource in its own domain and replies to NN2.

5. NN2 now returns a negative reply to NN1 because NN2 has exhausted its search logic.
6. NN4 owns the resource; therefore, it returns a positive reply to NN1.
7. Because the resource discovery search located the resource, NN1 sends a search to the target containing the original session-specific information.
8. The target is found.

Locate Resource: EN to NN to Subarea Network

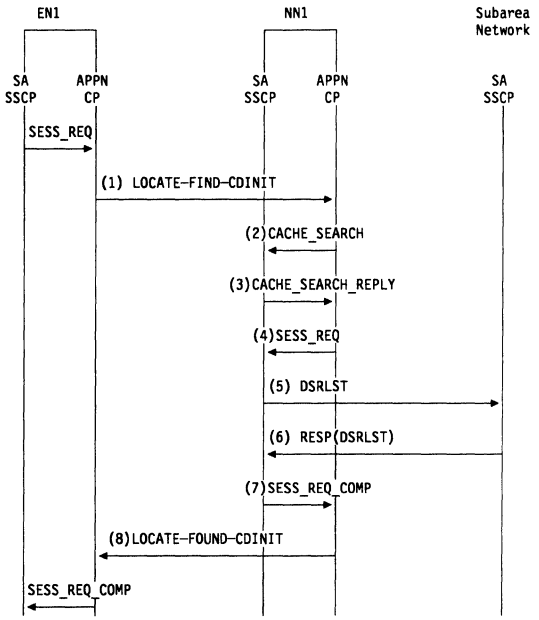


Figure 198. Locate Resource: EN to NN to Subarea Network

1. The end node sends a search request for a target resource to the network node server. As network node server of the originating LU, NN1 looks for the target resource in the directory database. NN1 has no knowledge of the target resource in the directory database.
2. The APPN control point (CP) requests that the subarea SSCP check for information about the location of the resource.
3. The subarea SSCP replies that an entry for the target resource is found.
4. The APPN CP requests that the subarea SSCP send the search request to the target resource.
5. The request is sent to the owning SSCP.
6. The owning SSCP indicates that it owns the target resource.
7. The subarea SSCP replies to the APPN CP that the target resource is found.
8. NN1 returns a positive reply to EN1.

Locate Resource: Complex APPN Network

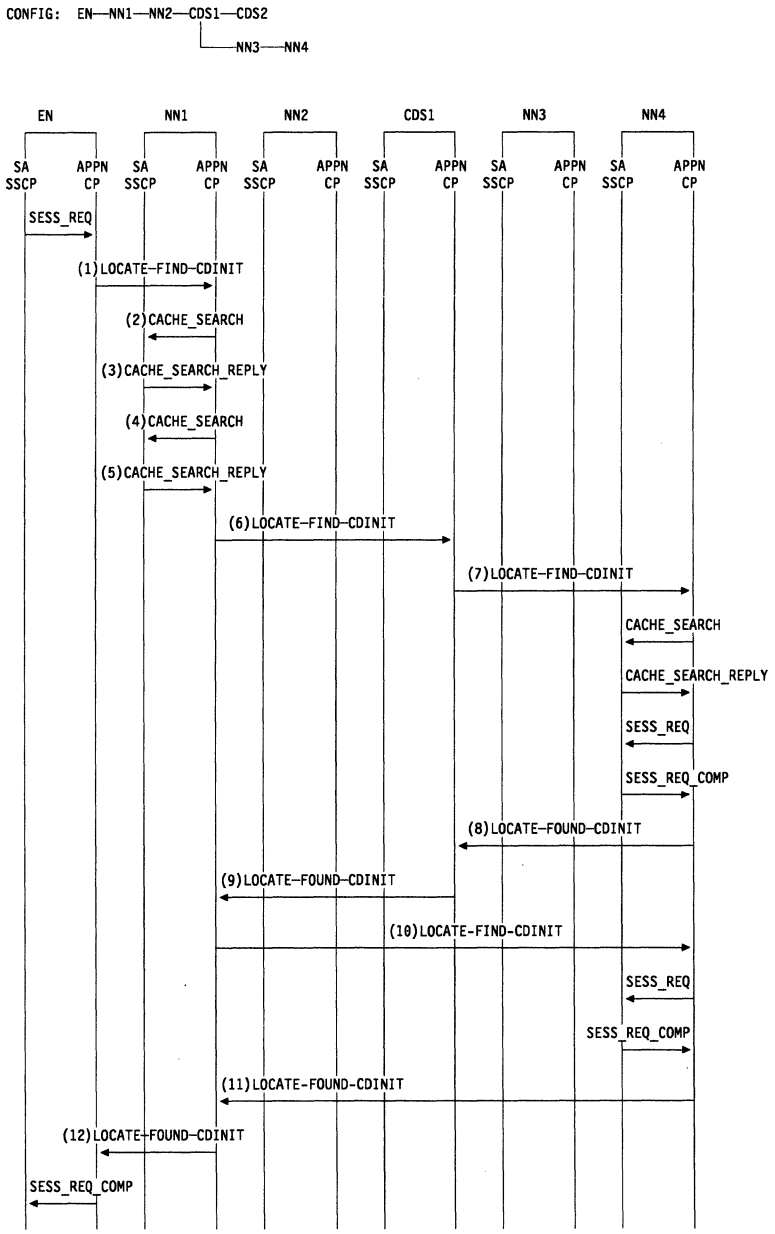


Figure 199. Locate Resource: Complex APPN Network

1. The end node sends a search request for a target resource to the network node server. As network node server of the originating LU, NN1 looks for the target resource in the directory database. NN1 has no knowledge of the target resource in the directory database.
2. The APPN control point (CP) requests that the subarea SSCP check for information about the location of the resource.
3. The subarea SSCP replies that the target resource is not known.
4. Because NN1 doesn't have the target resource in either its APPN or subarea directories, it initiates a resource discovery search for the resource. Because the resource discovery search starts at the beginning of the search logic,

another CACHE_SEARCH is performed. NN1 has no APPN-domain end nodes; therefore, no domain broadcast occurs.

5. The subarea SSCP replies that the target resource is not known.
6. NN1 does not initiate a network broadcast because there is a central directory server (CDS) in the network; therefore, NN1 sends a request to this CDS.
7. CDS1 receives the request and performs origin CDS logic. CDS1 looks in its directory database for the target resource and has an entry that indicates that the target resource resides on NN4. CDS1 sends the request to NN4.
8. NN4 owns the resource; therefore, NN4 returns a positive reply.
9. CDS1 replies to NN1.
10. Because the resource discovery search found the resource, NN1 sends a search to the target, containing the original session-specific information.
11. The target is found.
12. NN1 replies to the end node.

Locate Resource: Complex APPN Network Using More Than One CDS

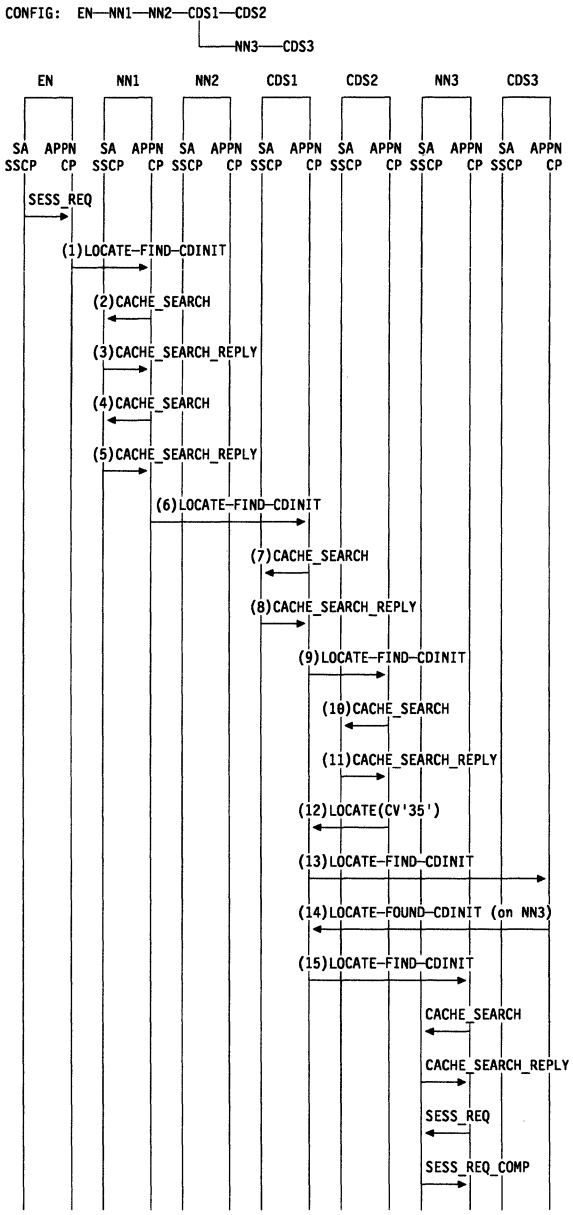


Figure 200 (Part 1 of 2). Locate Resource: Complex APPN Network Using More Than One CDS

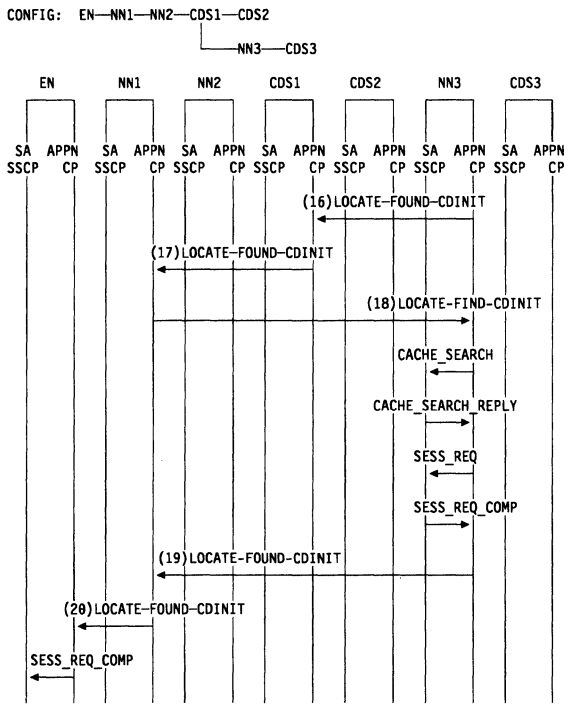


Figure 200 (Part 2 of 2). Locate Resource: Complex APPN Network Using More Than One CDS

1. The end node sends a search request for a target resource to the network node server. As network node server of the originating LU, NN1 looks for the target resource in the directory database. NN1 has no knowledge of the target resource in the directory database.
2. The APPN control point (CP) requests that the subarea SSCP check its resource information for information about the location of the resource.
3. The subarea SSCP replies that the target resource is not known.
4. Because NN1 doesn't have the target resource in either its APPN or subarea directories, it initiates a resource discovery search for the resource. Because the resource discovery search starts at the beginning of the search logic, another CACHE_SEARCH is performed.
5. The subarea SSCP replies that the target resource is not known.
6. NN1 has no APPN-domain end nodes; therefore, no domain broadcast occurs. NN1 does not initiate a network broadcast because there is a central directory server (CDS) in the network. NN1 sends a request to CDS1.
7. CDS1 receives the request and performs origin CDS logic. CDS1 looks in its directory database for the target resource and does not have an entry. Therefore, the APPN control point (CP) requests the location of the resource from the subarea SSCP.
8. The subarea SSCP replies that the target resource is not known.
9. CDS1 has no domain end nodes; therefore, no domain broadcast occurs. CDS1 begins an alternate CDS search by sending a request to CDS2.
10. CDS2 looks in its directory database for the target resource and does not have an entry. The APPN CP requests that the subarea SSCP check for information about the location of the resource.

11. The subarea SSCP replies that the target resource is not known.
12. CDS2 has no domain end nodes to which to send a domain broadcast. Therefore, CDS2 returns a negative reply to CDS1.
13. CDS1 continues the alternate CDS search by sending a request to CDS3.
14. CDS3 looks in its directory database and finds an entry that indicates that the target resource resides on NN3. CDS3 replies to CDS1 with this information.
15. CDS1 sends a request to NN3 to verify that the target actually resides there.
16. NN3 replies to CDS1 that it does, indeed, own the target resource.
17. CDS1 returns a reply to NN1.
18. Because the resource discovery search found the resource, NN1 sends a search to the target, containing the original session-specific information.
19. The target is found.
20. NN1 returns a reply to the end node.

Locate Resource: Network Node Server, NN1, of the Originating Logical Unit (OLU) is at Pre-V4R2 Level

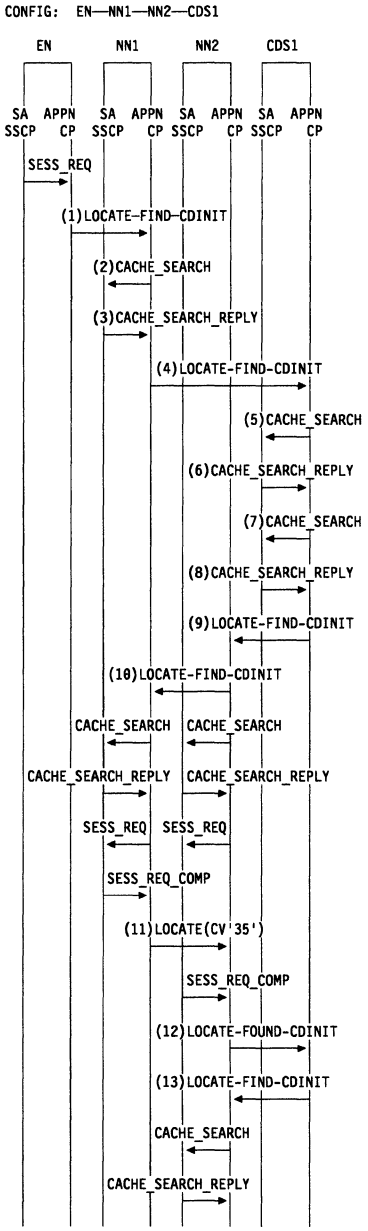


Figure 201 (Part 1 of 2). Locate Resource: Network Node Server, NN1, of the Originating Logical Unit (OLU) is at Pre-V4R2 Level

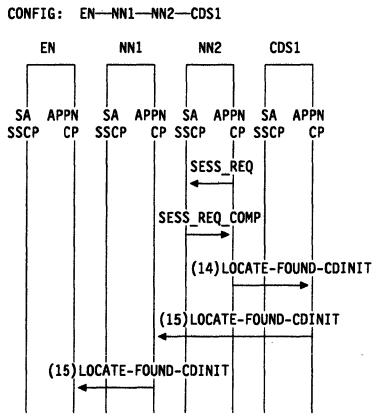


Figure 201 (Part 2 of 2). Locate Resource: Network Node Server, NN1, of the Originating Logical Unit (OLU) is at Pre-V4R2 Level

1. The end node sends a search request for a target resource to the network node server. As network node server of the originating LU, NN1 looks for the target resource in the directory database. NN1 has no knowledge of the target resource in the directory database.
2. The APPN control point (CP) requests that the subarea SSCP check its resource information for the location of the resource.
3. The subarea SSCP replies that the target resource is not known. NN1 has no APPN end nodes; therefore, no domain broadcast occurs.
4. Because there is a central directory server (CDS) in the network, NN1 does not initiate a network broadcast. Instead, NN1 sends a request to CDS1.
5. The CDS does not have knowledge of the target resource in its directory. The APPN control point (CP) requests that the subarea SSCP check its resource information for the location of the resource.
6. The subarea SSCP replies that the target resource is not known.
7. Because CDS1 does not have the target resource in either its APPN or subarea directories, it initiates a resource discovery search. Because the resource discovery search starts at the beginning of the search logic, another CACHE_SEARCH is performed.
8. The subarea SSCP replies that the target resource is not known.
9. CDS1 has no APPN end nodes; therefore, no domain broadcast occurs. CDS1 initiates a network broadcast for the target resource.
10. NN2 forwards the network broadcast request and then begins to search its domain.
11. After completing its search logic, NN1 returns a negative reply to the network broadcast request.
12. The subarea SSCP on NN2 indicates that the resource is found; therefore, a positive reply is returned.

13. Because the broadcast that was initiated by the resource discovery search found the resource, the original search request containing session-specific information is sent to the target's location, NN2.
14. NN2, the owner of the resource, returns a positive reply.
15. CDS1 replies to NN1, and NN1 replies to the end node.

Locate Resource: APPN and Subarea Network

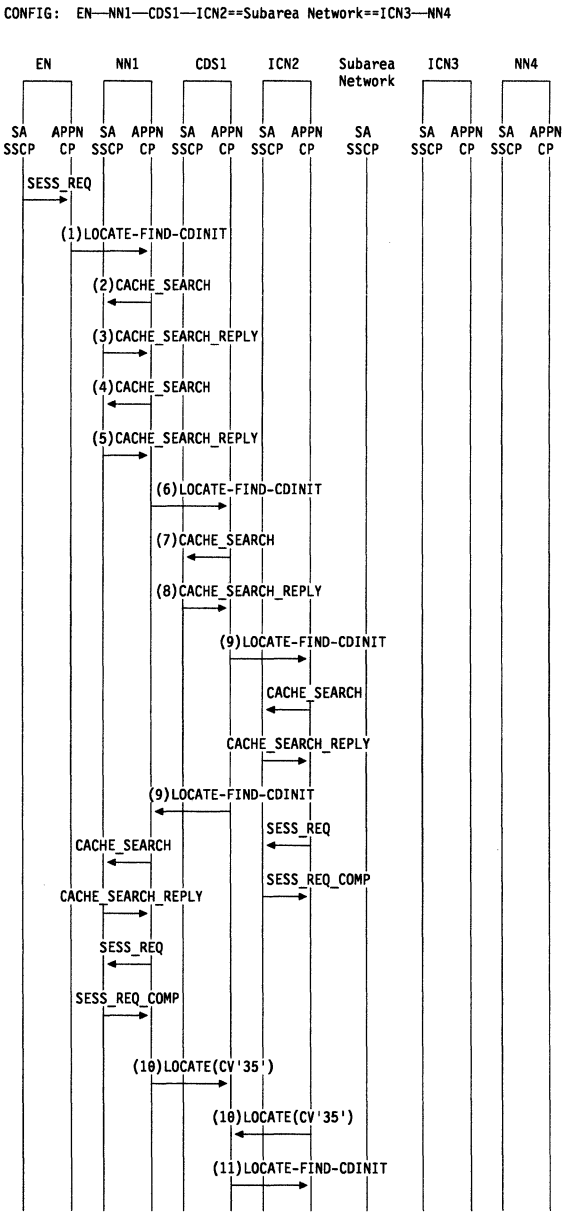


Figure 202 (Part 1 of 3). Locate Resource: APPN and Subarea Network

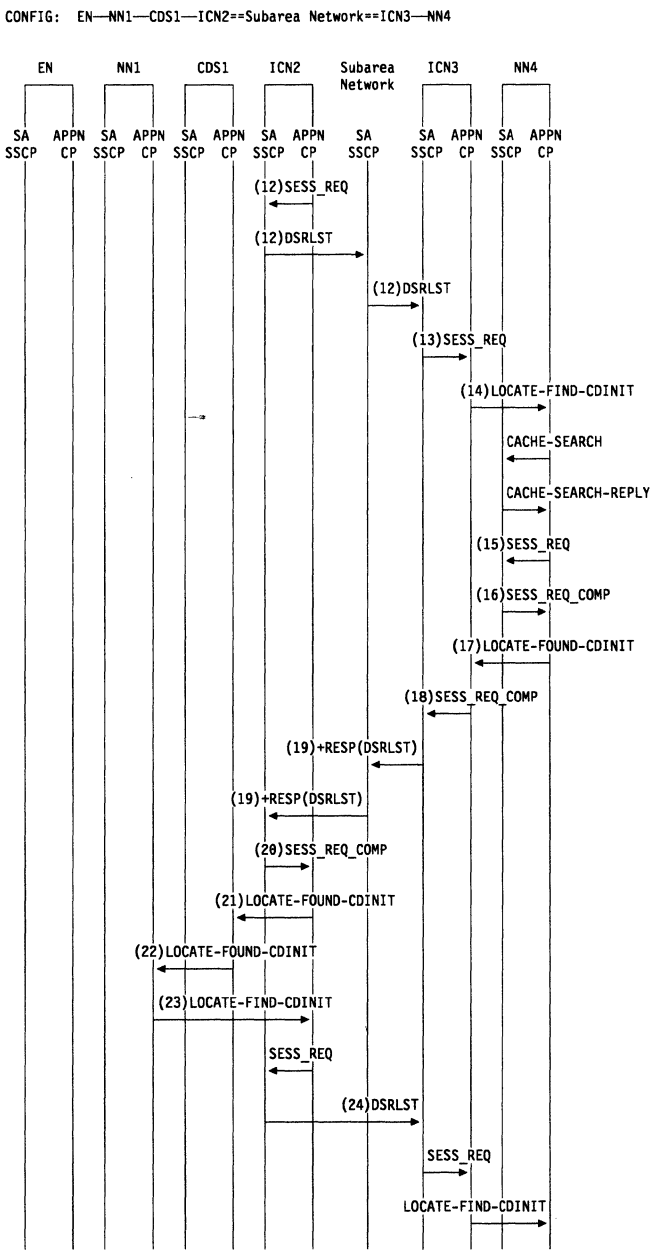


Figure 202 (Part 2 of 3). Locate Resource: APPN and Subarea Network

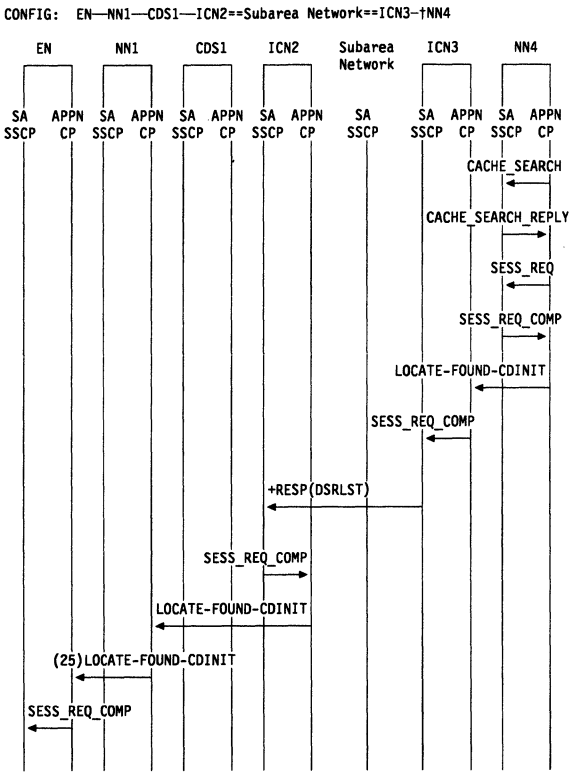


Figure 202 (Part 3 of 3). Locate Resource: APPN and Subarea Network

1. The end node sends a search request for a target resource to the network node server. As network node server of the originating LU, NN1 looks for the target resource in the directory database. NN1 has no knowledge of the target resource in the directory database.
2. The APPN control point (CP) requests that the subarea SSCP check for information about the location of the resource.
3. The subarea SSCP replies that the target resource is not known.
4. Because NN1 doesn't have the target resource in either its APPN or subarea directories, it initiates a resource discovery search for the resource. Because the resource discovery search starts at the beginning of the search logic, another CACHE_SEARCH is performed.
5. The subarea SSCP replies that the target resource is not known.
6. NN1 has no APPN-domain end nodes; therefore, no domain broadcast occurs. NN1 does not initiate a network broadcast because there is a central directory server (CDS) in the network; therefore, NN1 sends a request to CDS1.
7. CDS1 receives the request and performs origin CDS logic. CDS1 looks in its directory database for the target resource and does not have an entry. Therefore, the APPN control point (CP) requests the location of the resource from the subarea SSCP.
8. The subarea SSCP replies that the target resource is not known.
9. CDS1 has no domain end nodes; therefore, no domain broadcast occurs. There are no other CDSs in the network; therefore, no alternate CDS search occurs. CDS1 initiates a network broadcast. CDS1 sends the broadcast request to all nodes with which it has CP-CP sessions.

Notes:

a. CDS1 must send the broadcast request to NN1, even though NN1 originated the request to CDS1 because there can be parts of the APPN network that are reachable only through NN1 (these parts are not shown here).

b. The network broadcast sent by a CDS indicates that attached subarea networks should not be searched at this time.

10. Both NN1 and ICN2 respond that the target is not found.
11. After CDS1 has collected all the replies from the network broadcast, CDS1 continues the search with an interchange node search. CDS1 sends the interchange node search request to ICN2. This request indicates that the interchange node is to search its attached subarea network.
12. The APPN CP requests that the subarea SSCP initiate subarea routing. ICN2 sends DSRLST to adjacent SSCPs.
13. The subarea SSCP requests that the APPN CP initiate APPN searching.
14. ICN3 looks for the target resource in the directory database. ICN3 has knowledge in its directory database that the target resource resides in NN4. ICN3 sends a directed search request to NN4.
15. The APPN CP sends a SESS_REQ signal to the subarea SSCP.
16. The subarea SSCP sends a SESS_REQ_COMP to the APPN CP indicating that the target resource is located.
17. NN4 returns a positive reply to NN3.
18. The APPN CP replies to the subarea SSCP.
19. Positive responses to DSRLSTs are returned.
20. The subarea SSCP replies to the APPN CP.
21. ICN2 replies to CDS1.
22. CDS1 replies to NN1.
23. Because the resource discovery search located the resource, NN1 sends a search to the target, containing the original session-specific information.
24. A DSRLST, which contains session-specific information, is sent.
25. NN1 replies to the end node.

Locate Resource: SLU-Initiated Session

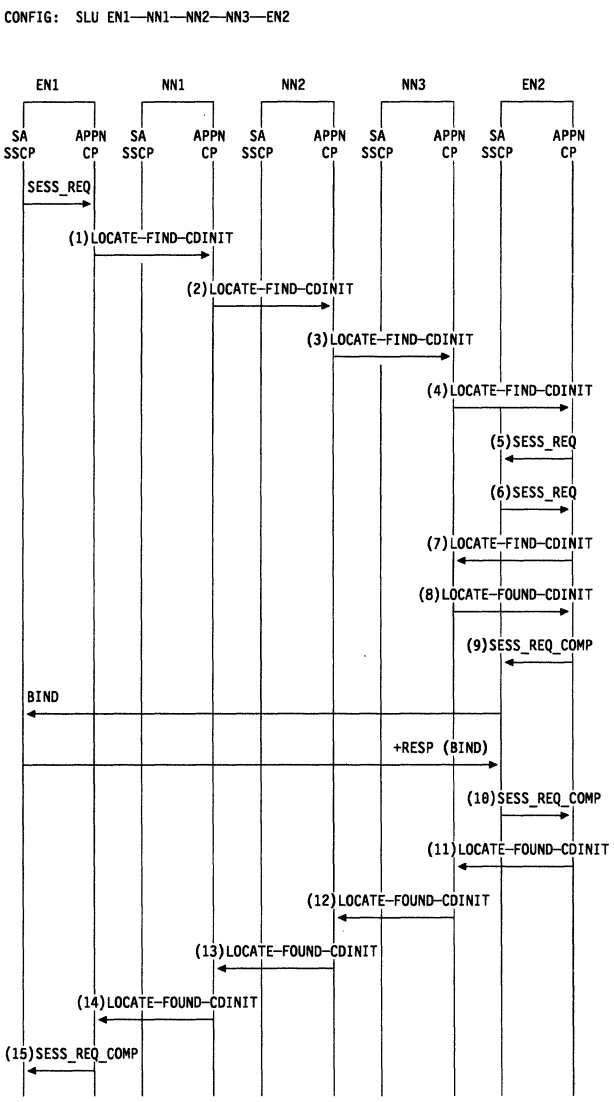


Figure 203. Locate Resource: SLU-Initiated Session

1. The end node sends a session request for a primary LU (PLU) to its network node server. NN1 looks for the PLU in its directory database. NN1 has knowledge in the directory database that the PLU resides on EN2, which is served by NN3.
2. The network node sends a directed search request to NN3. Because NN1 does not have direct CP-CP sessions with NN3, NN1 sends the directed search request to NN3 through NN2.
3. NN2 is not the destination of the directed search; therefore, NN2 simply forwards the request to NN3.
4. NN3 receives the request and forwards it to EN2.
5. The APPN CP sends a SESS_REQ signal to the subarea SSCP.

6. The PLU initiates a search for the secondary LU (SLU), indicating that the location of the target does not have to be verified but that an RSCV must be calculated.
7. The end node sends a search request for the SLU to its network node server.
8. NN3 looks for the SLU in its directory database. NN3 has knowledge in its directory database that the SLU resides on EN1. Further, NN3 has information that allows the search to succeed without verifying the location of the SLU. On behalf of EN1, NN3 returns a positive reply, which includes the RSCV for the session.
9. The APPN CP replies to the subarea SSCP for the PLU-initiated request.
10. The subarea SSCP replies to the APPN CP for the SLU-initiated request, indicating that the session is already active.
11. The end node sends a reply for the SLU-initiated request to NN3.
12. NN3 replies to NN2.
13. NN2 replies to NN1.
14. NN1 replies to EN1.
15. The APPN CP replies to the subarea SSCP.

Locate Resource: CP-CP Session Terminates

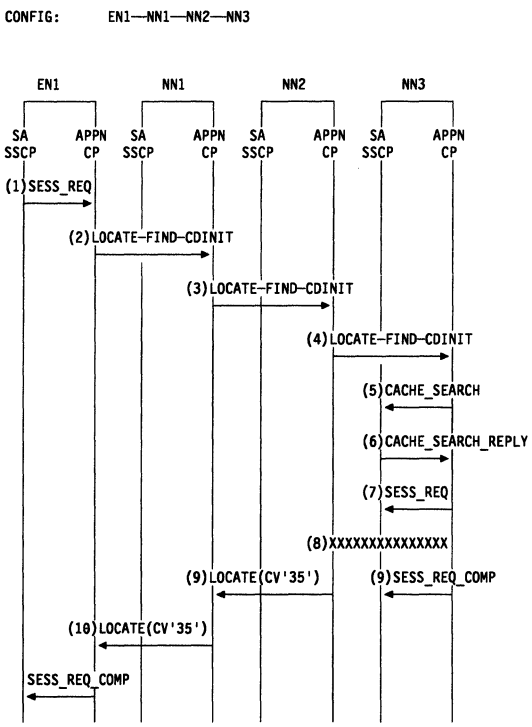


Figure 204. Locate Resource: CP-CP Session Terminates

1. The subarea SSCP requests a search for a target resource.
2. The end node sends a search request for a target resource to the network node server. The network node looks for the target resource in the directory database. The network node has knowledge in the directory that the target resource resides on NN3.
3. The network node sends a directed search request to NN3. Because NN1 does not have direct CP-CP sessions with NN3, NN1 sends the directed search request to NN3 through NN2.
4. NN2 is not the destination of the directed search; therefore, NN2 simply forwards the request to NN3.
5. The APPN control point (CP) requests that the subarea SSCP check its resource information for information about the location of the resource.
6. The subarea SSCP replies that the target resource is found.
7. The APPN CP requests that the subarea SSCP initiate a search for the target resource.
8. The CP-CP session goes down between NN2 and NN3.
9. NN2 sends a negative reply to NN1. NN3 cleans up its control blocks.
10. NN1 continues its search logic. If another path to the target exists, the target can be found; for example, through a network broadcast search. Otherwise, the search does not find the target resource.

Locate Resource: Network Node Receives Network Broadcast Request

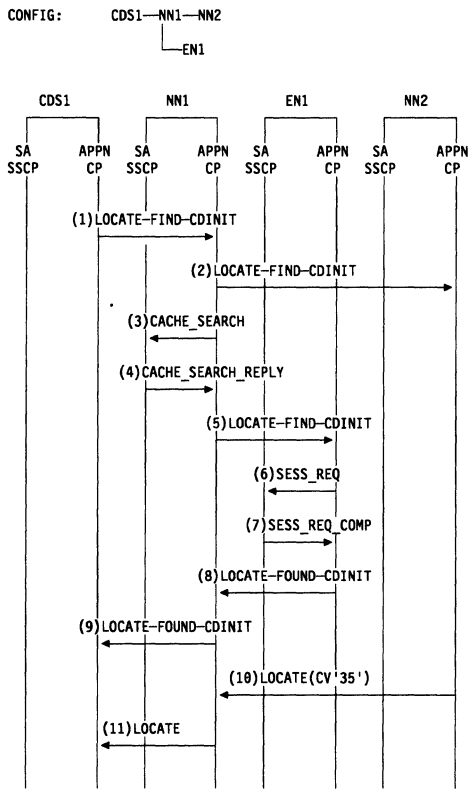


Figure 205. Locate Resource: Network Node Receives Network Broadcast Request

Note: This figure illustrates only the flows at NN1.

1. CDS1 has reached the point in its search logic where a network broadcast is performed. CDS1 sends the network broadcast to all network nodes with which it has CP-CP sessions.
2. NN1 recognizes that the request is a network broadcast. Therefore, it forwards the request immediately to all network nodes with which it has CP-CP sessions.
3. NN1 then begins to search itself and its domain. NN1 looks for the target resource in its directory database. NN1 does not have an entry for the target resource. The search continues with the APPN CP requesting that the subarea SSCP check its resource information for information about the location of the resource.
4. The subarea SSCP replies that the target is not known.
5. NN1 sends the domain broadcast request to all served ENs that indicated in the CP_CAPABILITIES exchange that they are to be searched on domain broadcasts.
6. The APPN CP forwards the search to the subarea SSCP.
7. The subarea SSCP indicates that the target resource was found.
8. EN1 returns a positive reply.
9. NN1 forwards the positive reply to CDS1 and continues to wait for a reply from NN2. NN1 indicates to CDS1 in its reply that NN1 has not yet received replies

to all of its searches. CDS1 forwards the positive reply to the originator of the search and continues to wait for a final reply from NN1.

10. NN2 returns a negative reply to NN1. Because NN1 has now completed all of its searches, NN1 returns a Neutral reply to CDS1 to indicate that NN1 has completed its search. A Neutral reply is one that contains neither a CV'35' (to indicate failure) or a FOUND GDS variable (to indicate success). A CDS receiving a "neutral" reply does not forward the neutral reply to the originator of the search. A CDS returns only one reply to the originator to indicate success or failure.

LU-LU Session Flows

Figure 206 on page 1059 through Figure 242 on page 1097 show the network flows to establish LU-LU sessions.

The figure captions for some of the figures indicate the configuration that the flow applies to and the type of session shown. For example, the caption `EN (PLU)–NNS...APPN Network; PLU-Initiated, with No Queueing` means *A primary logical unit (PLU) on an end node (EN) that is attached to an APPN network through a network node server (NNS).* The symbol "—" indicates a CP-CP session. The symbol "==" indicates a CDRM-CDRM session. The symbol "..." means that part of the network is not shown.

All of these flows assume that the directory services database has accurate information about the location of the destination LU.

The following terms are used in these figures:

Term	Meaning
Endpoint TGVs	A list of control vector pairs: CV X'46' and X'47'
RSCV	Route selection control vector, CV X'2B'
Scout search	Sent to find out the location of the destination LU (DLU) and to precompute the session RSCV, if either the origin LU (OLU) or the DLU is in a subarea network. Because it is necessary only to find the DLU and not to set up the session, it is not necessary to actually verify the location of the DLU or to reserve resources for the session.

Index of LU-LU Session Flows

Table 74 lists the LU-LU session flows illustrated here.

Table 74 (Page 1 of 2). Index of LU-LU Session Flows

Flow	Page
APPN network...NNS--EN (PLU)	
SLU-initiated, no queueing	Figure 213 on page 1066
SLU-initiated, queued by the PLU	Figure 214 on page 1067
APPN network...NNS--EN (SLU)	
PLU-initiated, no queueing	Figure 211 on page 1064
PLU-initiated, queued by the SLU	Figure 212 on page 1065
APPN network (PLU)...ICN==SA(SLU), PLU-Initiated	
Directed search without required precompututed RSCV	Figure 236 on page 1090
No queueing	Figure 235 on page 1089
Queued by SLU	Figure 238 on page 1092
Search-only flow transformed into a DSRLST	Figure 234 on page 1088
USERVAR resolution required	Figure 237 on page 1091
APPN network (PLU)...ICN==VR-based TG==ICN...APPN network (SLU)	
PLU-initiated	Figure 242 on page 1097

Table 74 (Page 2 of 2). Index of LU-LU Session Flows

Flow	Page
APPN network (SLU)...ICN==PLU	
Autologon, PLU not available initially	Figure 241 on page 1096
SLU-initiated, no queueing	Figure 239 on page 1093
SLU-initiated, queued by the PLU	Figure 240 on page 1094
CLSDST PASS; SLU is single-session capable	
From APPN to subarea	Figure 231 on page 1085
Through APPN	Figure 230 on page 1084
EN-NN-EN, PLU-initiated, no queueing (including BIND flows for intermediate network node)	Figure 232 on page 1086
EN (PLU)--NNS...APPN network	
PLU-initiated, no queueing	Figure 206 on page 1059
PLU-initiated, queued by the PLU	Figure 207 on page 1060
PLU-initiated, queued by the SLU	Figure 208 on page 1061
EN (SLU)--NNS...APPN network	
SLU-initiated, no queueing	Figure 209 on page 1062
SLU-initiated, queued by the PLU	Figure 210 on page 1063
Intermediate Network Node (INN) BIND. The LOCATE did not go through this node.	Figure 233 on page 1087
SA (PLU)==ICN...APPN network (SLU)	
DSRLIST transforming into PLU-initiated, search-only	Figure 215 on page 1068
PLU-initiated, no queueing	Figure 216 on page 1069
PLU-initiated, queued by the SLU	Figure 218 on page 1071
PLU-initiated, USERVAR resolution required	Figure 217 on page 1070
SA (SLU)==ICN....APPN network (PLU)	
Autologon, PLU not available initially	Figure 221 on page 1076
SLU-initiated, no queueing	Figure 219 on page 1073
SLU-initiated, queued by the PLU	Figure 220 on page 1074
Session release request	
SA(PLU)==ICN...APPN network(SLU)	Figure 228 on page 1083
SA(SLU)==ICN...APPN network(PLU)	Figure 229 on page 1083
Session termination, forced	
SA(PLU)==ICN...APPN Network(SLU), pending active session. PLU is accessible without going into APPN.	Figure 225 on page 1080
SA(PLU)==ICN...APPN Network(SLU), queued session	Figure 226 on page 1081
SA(PLU)==ICN...APPN Network(SLU), queued session. PLU is accessible without going into APPN.	Figure 227 on page 1082
SA(SLU)==ICN...APPN Network(PLU), pending active session	Figure 224 on page 1079
Session termination, orderly	
SA(PLU)==ICN...APPN Network(SLU), active session	Figure 222 on page 1077
APPN Network(PLU)...ICN==SA(SLU), active session	Figure 223 on page 1078

EN (PLU)--NNS...APPN Network, PLU-Initiated, with No Queueing

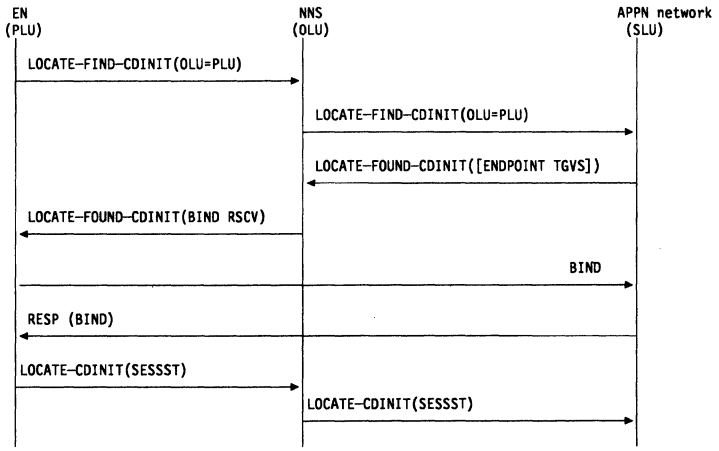


Figure 206. EN (PLU)—NNS...APPN Network, PLU-Initiated, with No Queueing

Note: The BIND does not have to take the same path as the LOCATE flow.

EN (PLU)--NNS...APPN Network, PLU-Initiated, Queued by the PLU

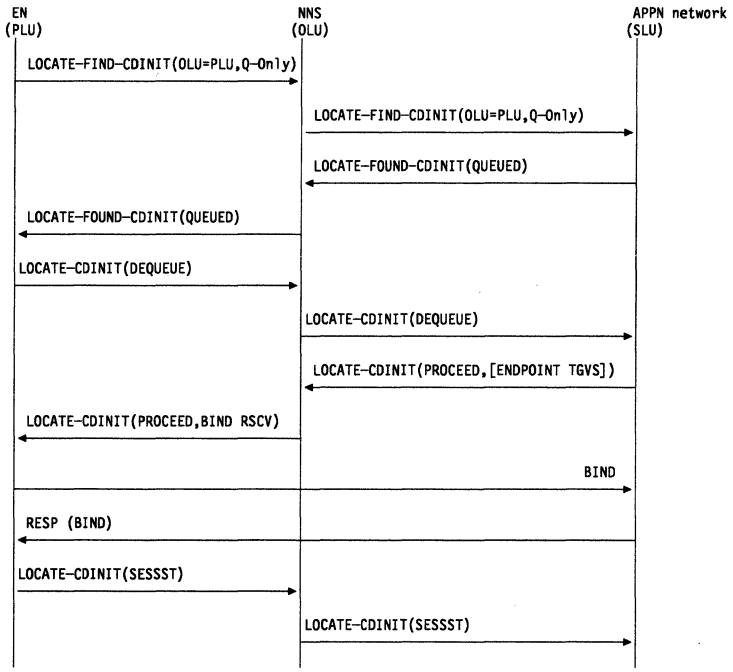


Figure 207. EN (PLU)—NNS...APPN Network, PLU-Initiated, Queued by the PLU

Note: The BIND does not have to take the same path as the LOCATE flow.

EN (PLU)--NNS...APPN Network, PLU-Initiated, Queued by the SLU

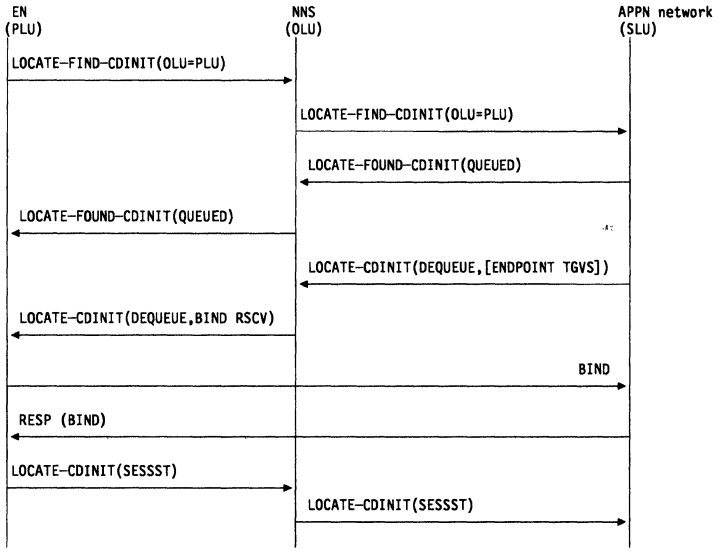


Figure 208. EN (PLU)—NNS...APPN Network, PLU-Initiated, Queued by the SLU

Note: The BIND does not have to take the same path as the LOCATE flow.

EN (SLU)--NNS...APPN Network, SLU-Initiated, with No Queueing

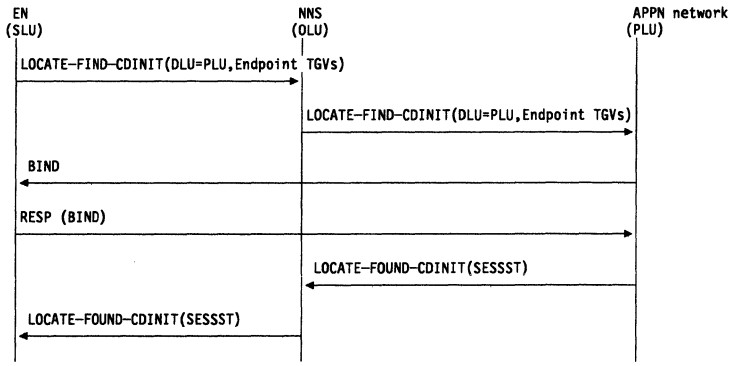


Figure 209. EN (SLU)—NNS...APPN Network, SLU-Initiated, with No Queueing

Note: The BIND does not have to take the same path as the LOCATE flow.

EN (SLU)--NNS...APPN Network, SLU-Initiated, Queued by the PLU

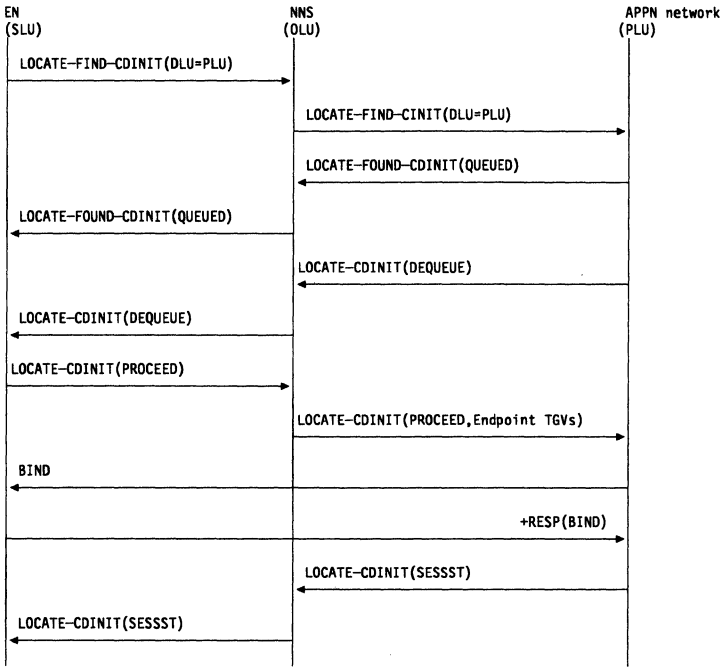


Figure 210. EN (SLU)—NNS...APPN Network, SLU-Initiated, Queued by the PLU

Note: The BIND does not have to take the same path as the LOCATE flows.

APPN Network...NNS--EN (SLU), PLU-Initiated, No Queueing

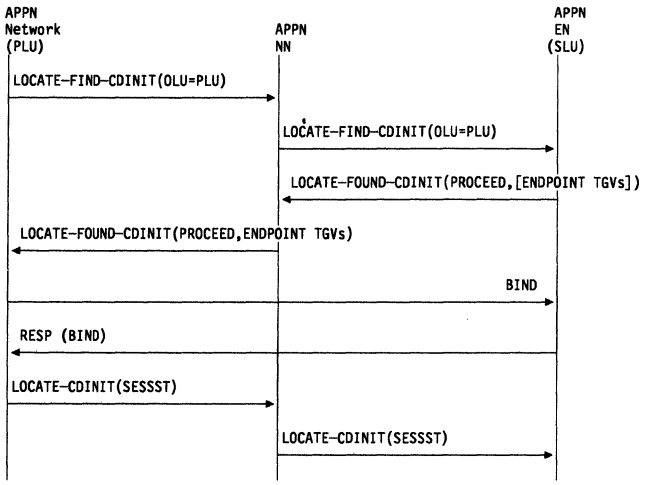


Figure 211. APPN Network...NNS—EN (SLU), PLU-Initiated, No Queueing

Note: The BIND does not have to follow the same path as the LOCATE flow.

APPN Network...NNS--EN (SLU), PLU-Initiated, Queued by the SLU

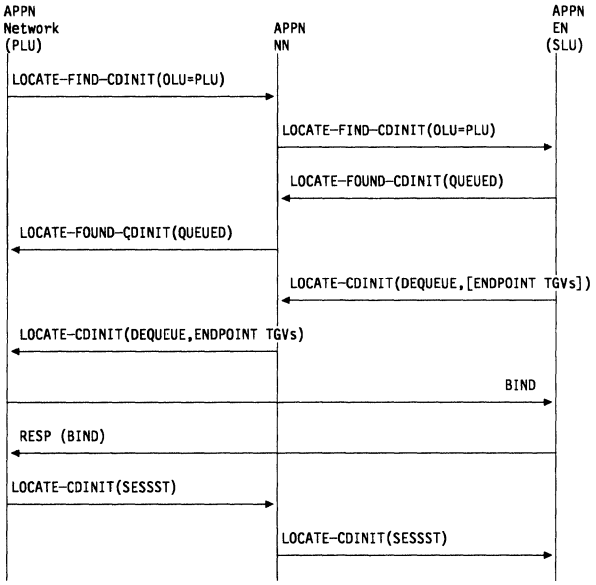


Figure 212. APPN Network...NNS—EN (SLU), PLU-Initiated, Queued by the SLU

Note: The BIND does not have to follow the same path as the LOCATE flow.

APPN Network...NNS--EN (PLU), SLU-Initiated, No Queueing

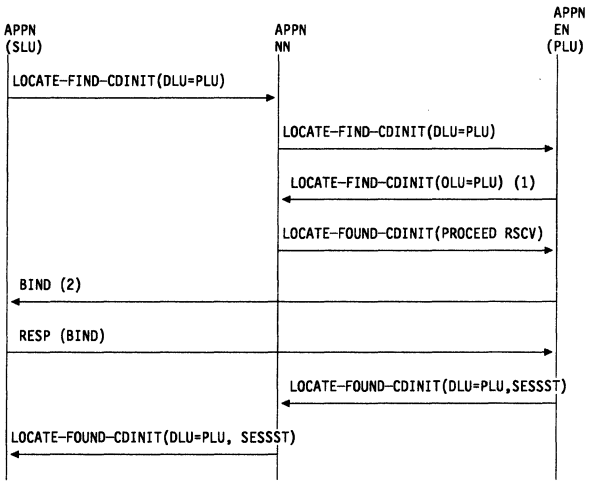


Figure 213. APPN Network...NNS—EN (PLU), SLU-Initiated, No Queueing

1. The purpose of this search is to get the NN to compute the session RSCV for the EN.
2. The BIND does not have to follow the same path as LOCATE flows.

APPN Network...NNS--EN (PLU), SLU-Initiated, Queued by the PLU

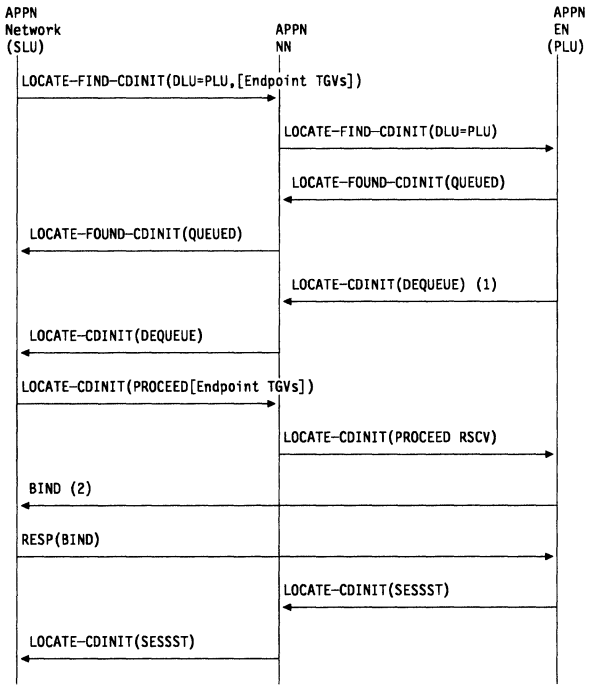


Figure 214. APPN Network...NNS—EN (PLU), SLU-Initiated, Queued by the PLU

1. The PLU has become available.
2. The BIND does not have to follow the same path as the LOCATE flows.

SA (PLU)==ICN...APPN Network (SLU), DSRLIST Transforming into PLU-Initiated, Search-Only

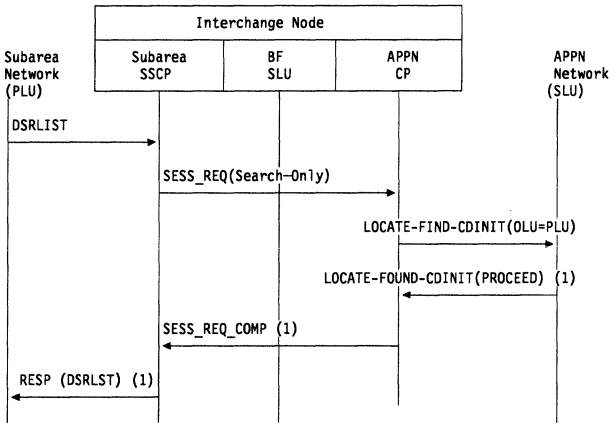


Figure 215. SA(PLU)==ICN...APPN Network (SLU), DSRLIST Transforming into PLU-Initiated, Search-Only

1. Target LU location information.

SA (PLU)==ICN...APPN Network (SLU), PLU-Initiated, with No Queueing

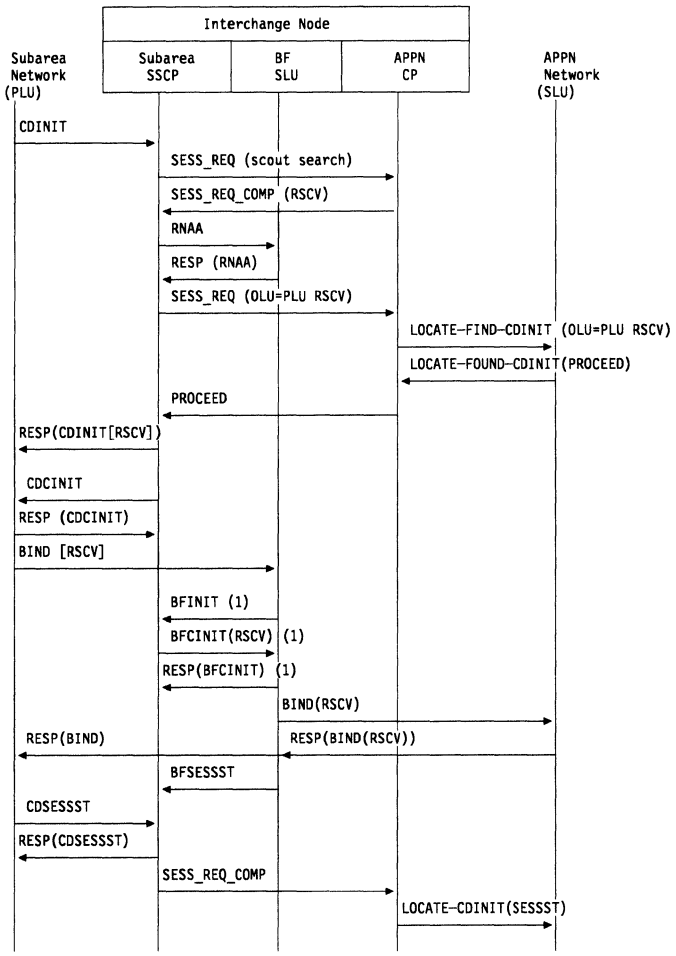


Figure 216. SA (PLU)==ICN...APPN Network (SLU), PLU-Initiated, with No Queueing

1. These BFINIT/BFCINIT flows will not occur if the RSCV is passed on the BIND.

SA (PLU)==ICN...APPN Network (SLU), PLU-Initiated, USERVAR Resolution Required

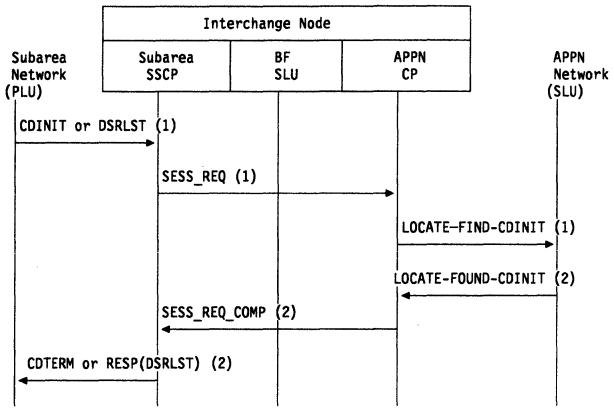


Figure 217. SA (PLU)==ICN...APPN Network (SLU), PLU-Initiated, USERVAR Resolution Required

1. Generic USERVAR name.
2. Resolved USERVAR name.

Note: For remaining session setup flows, see Figure 216 on page 1069.

SA (PLU)==ICN...APPN Network (SLU), PLU-Initiated, Queued by the SLU

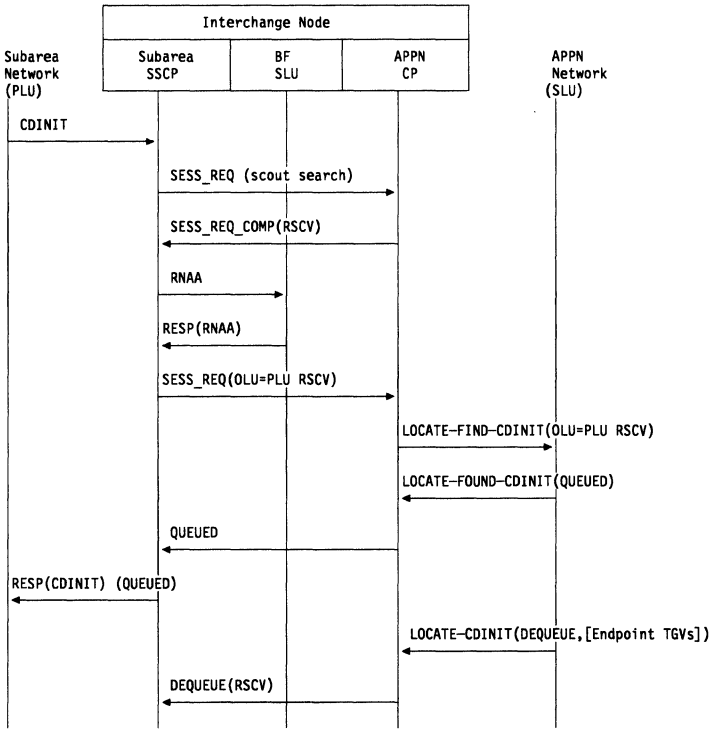


Figure 218 (Part 1 of 2). SA (PLU)==ICN...APPN Network (SLU), PLU-Initiated, Queued by the SLU

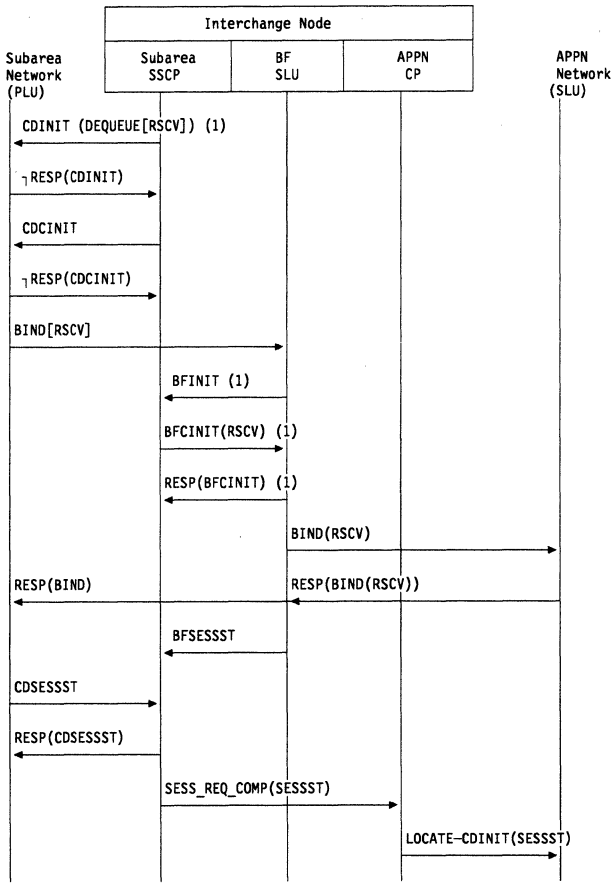


Figure 218 (Part 2 of 2). SA (PLU)==ICN...APPN Network (SLU), PLU-Initiated, Queued by the SLU

1. These BFINIT/BFCINIT flows will not occur if the RSCV is passed on the BIND.

SA (SLU)==ICN...APPN Network (PLU), SLU-Initiated, No Queueing

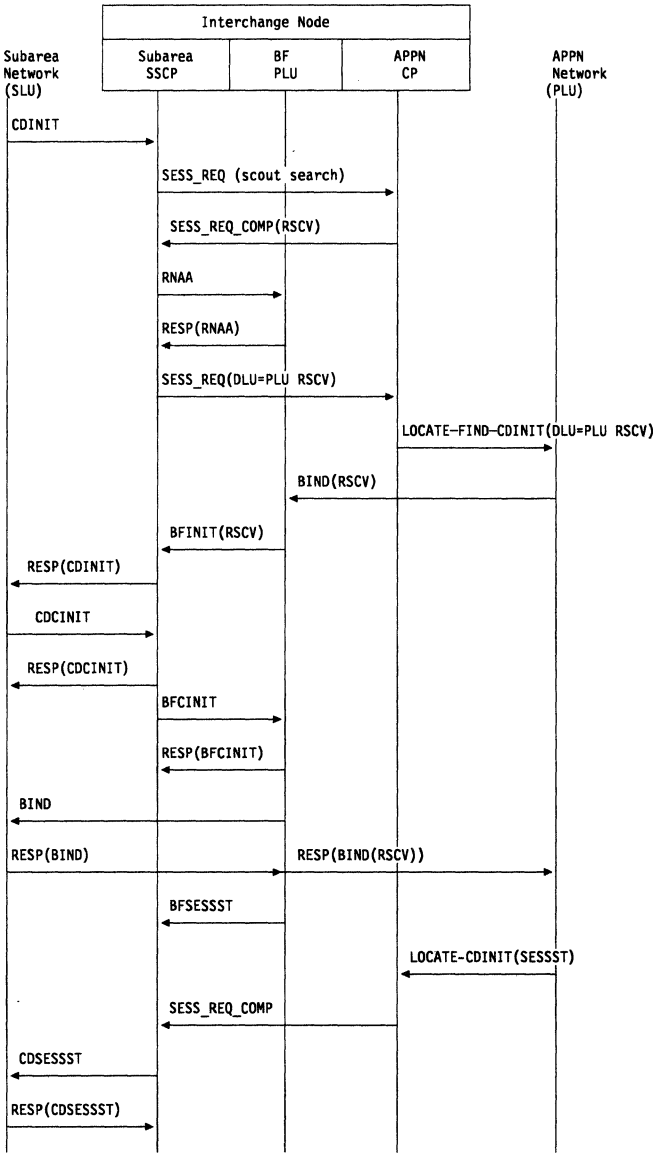


Figure 219. SA (SLU)==ICN...APPN Network (PLU), SLU-Initiated, No Queueing

SA (SLU)==ICN...APPN Network (PLU), SLU-Initiated, Queued by the PLU

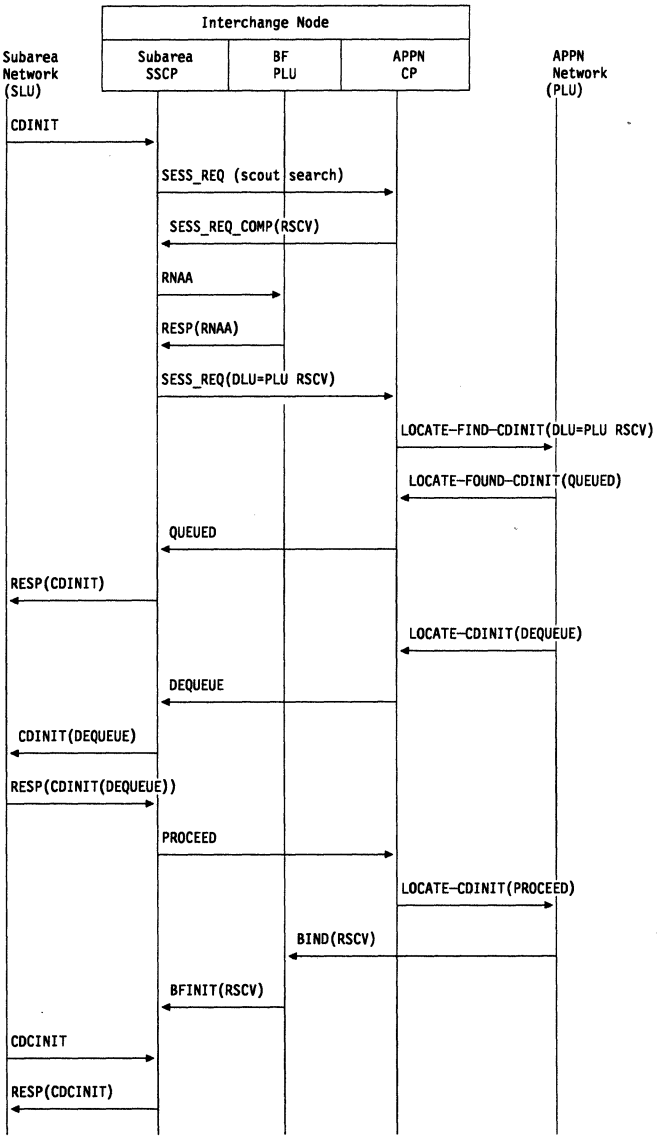


Figure 220 (Part 1 of 2). SA (SLU)==ICN...APPN Network (PLU), SLU-Initiated, Queued by the PLU

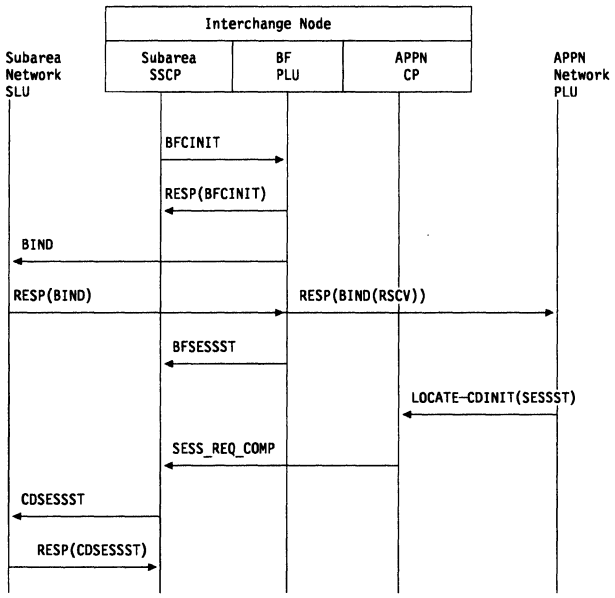


Figure 220 (Part 2 of 2). SA (SLU)==ICN...APPN Network (PLU), SLU-Initiated, Queued by the PLU

SA (SLU)==ICN...APPN network (PLU), Autologon, PLU Not Available Initially

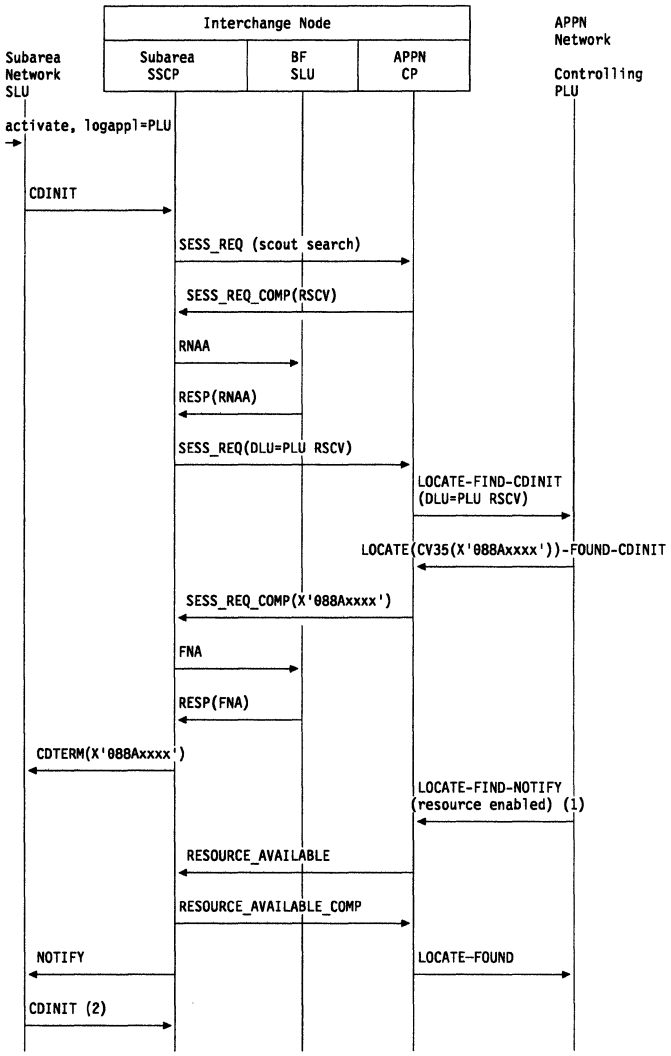


Figure 221. SA (SLU)==ICN...APPN network (PLU), Autologon, PLU not available initially

1. The controlling PLU becomes available.
2. Normal SLU-initiated flows continue from here.

SA(PLU)==ICN...APPN Network(SLU), Orderly Termination of Active Session

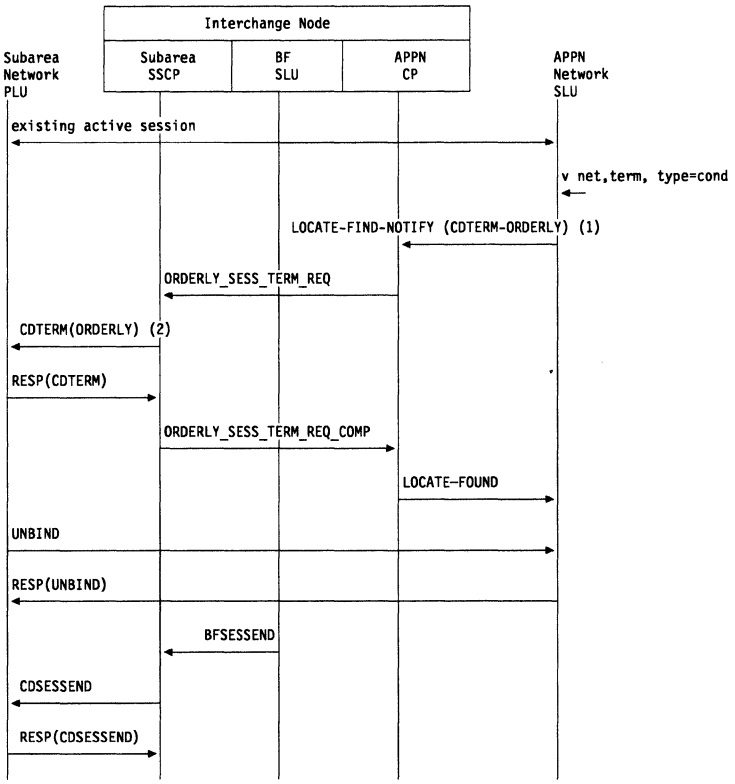


Figure 222. SA(PLU)==ICN...APPN Network(SLU), Orderly Termination of Active Session

1. FQCPID of the session to be terminated.
2. The CDTERM type depends on the V NET,TERM type.

V NET,TERM type	CDTERM type
COND	ORDERLY
UNCOND	FORCED
FORCE	CLEANUP

APPN Network(PLU)...ICN==(SA)SLU, Orderly Termination of Active Session

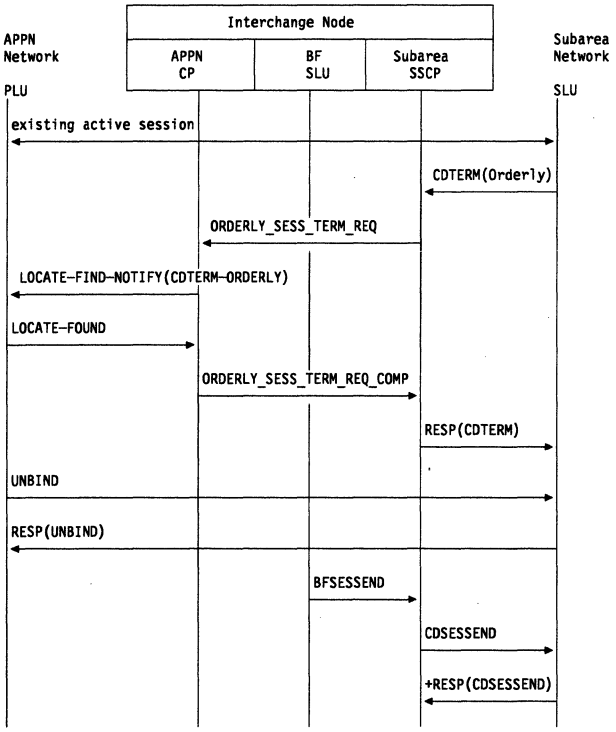


Figure 223. APPN Network(PLU)...ICN==(SA)SLU, Orderly Termination of Active Session

SA(SLU)==ICN...APPN Network(PLU), Forced Termination of Pending Active Session

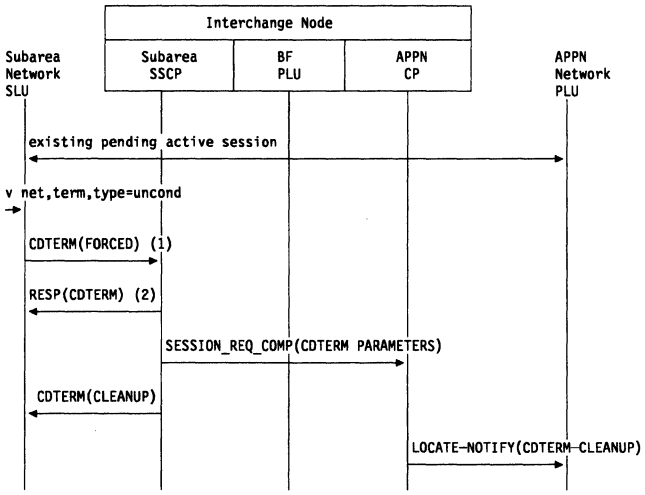


Figure 224. SA(SLU)==ICN...APPN Network(PLU), Forced Termination of Pending Active Session

1. The CDTERM type depends on the V NET,TERM type.

V NET,TERM type	CDTERM type
COND	ORDERLY
UNCOND	FORCED
FORCE	CLEANUP

2. APPN only has orderly and cleanup termination. Therefore, the forced termination is promoted to cleanup when it crosses from subarea into APPN.

SA(PLU)==ICN...APPN Network(SLU), Forced Termination of Pending Active Session (PLU Accessible without Going into APPN)

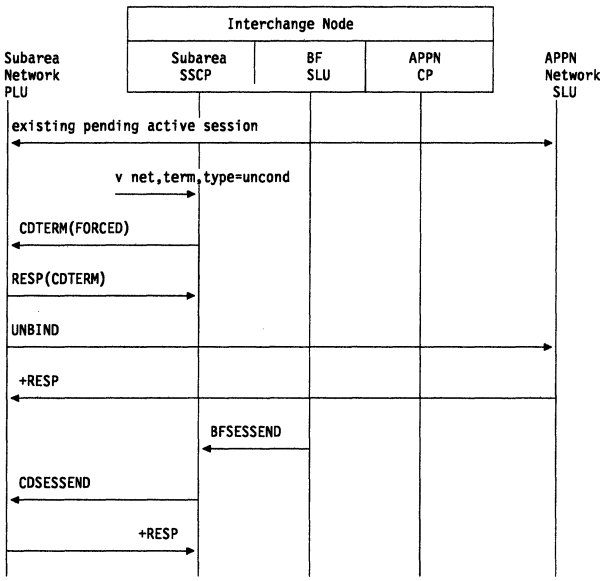


Figure 225. SA(PLU)==ICN...APPN Network(SLU), Forced Termination of Pending Active Session (PLU Accessible without Going into APPN)

Note: Whenever a forced termination crosses the boundary from subarea into APPN, it is promoted to cleanup. In this case because the PLU is accessible without going into the APPN network, promotion does not occur.

SA(PLU)==ICN...APPN Network(SLU), Forced Termination of Queued Session

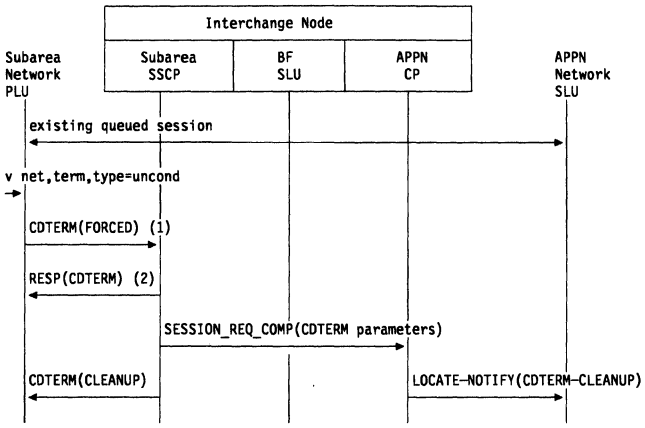


Figure 226. SA(PLU)==ICN...APPN Network(SLU), Forced Termination of Queued Session

1. The CDTERM type depends on the V NET,TERM type.

V NET,TERM type	CDTERM type
COND	ORDERLY
UNCOND	FORCED
FORCE	CLEANUP

2. APPN only has orderly and cleanup termination. Therefore, the forced termination is promoted to cleanup when it crosses from subarea into APPN.

SA(PLU)==ICN...APPN Network(SLU), Forced Termination of Queued Session (PLU Accessible without Going into APPN)

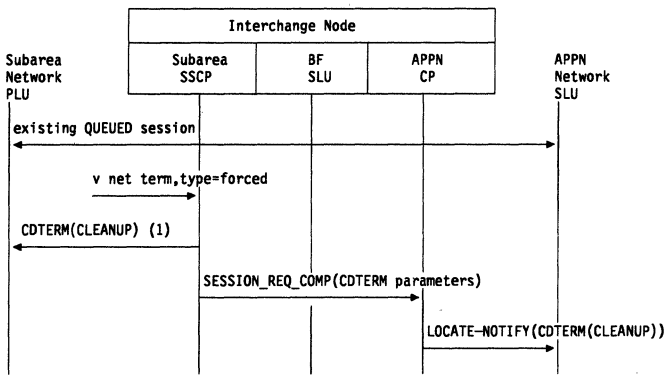


Figure 227. SA(PLU)==ICN...APPN Network(SLU), Forced Termination of Queued Session (PLU Accessible without Going into APPN)

1. Because the session is queued (instead of pending active) and the forced termination is not issued in the primary LU's domain, a CLEANUP is sent.

SA (PLU)==ICN...APPN Network (SLU), Session Release Request

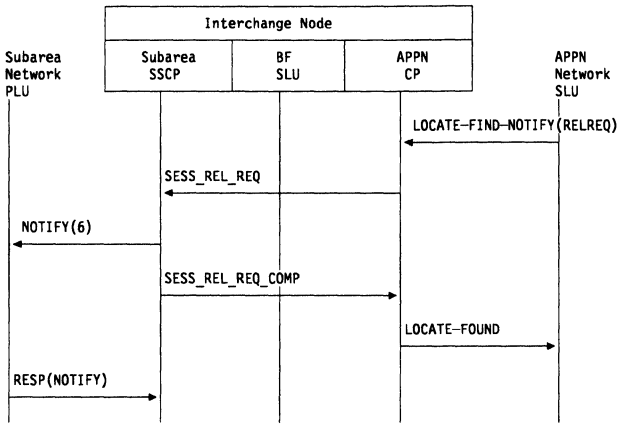


Figure 228. SA (PLU)==ICN...APPN Network (SLU), Session Release Request

SA (SLU)==ICN...APPN Network (PLU), Session Release Request

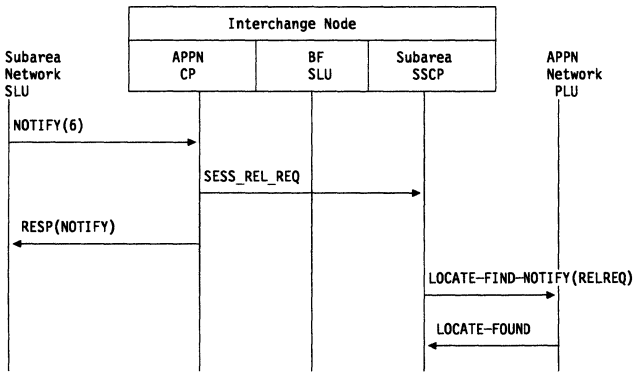


Figure 229. SA (SLU)==ICN...APPN Network (PLU), Session Release Request

CLSDST PASS through APPN

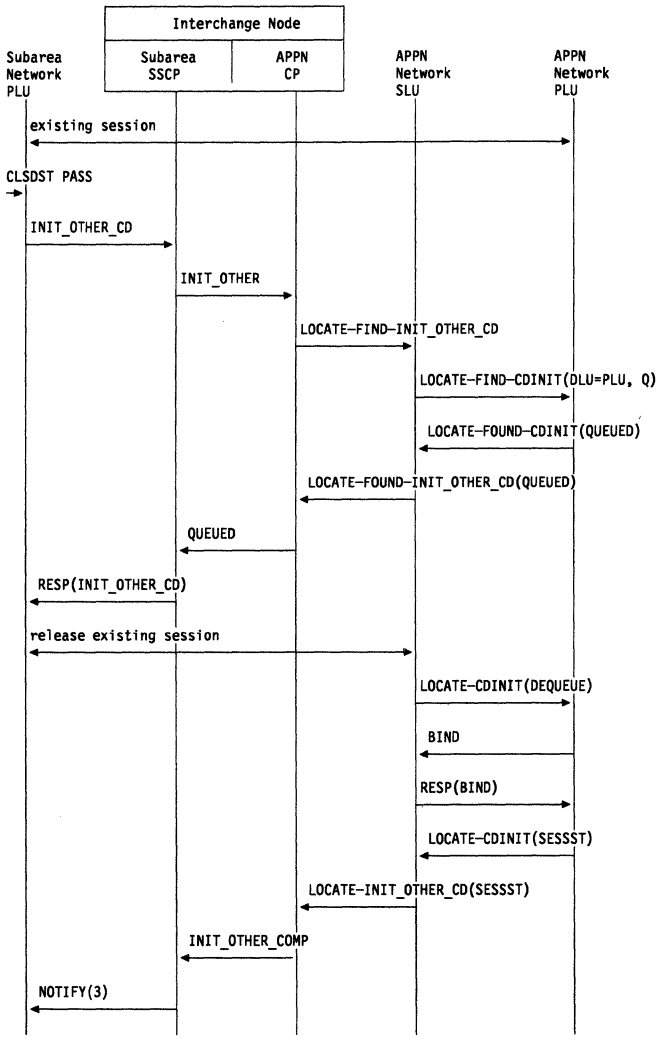


Figure 230. CLSDST PASS through APPN. The SLU is Single-Session Capable.

CLSDST PASS from APPN to Subarea

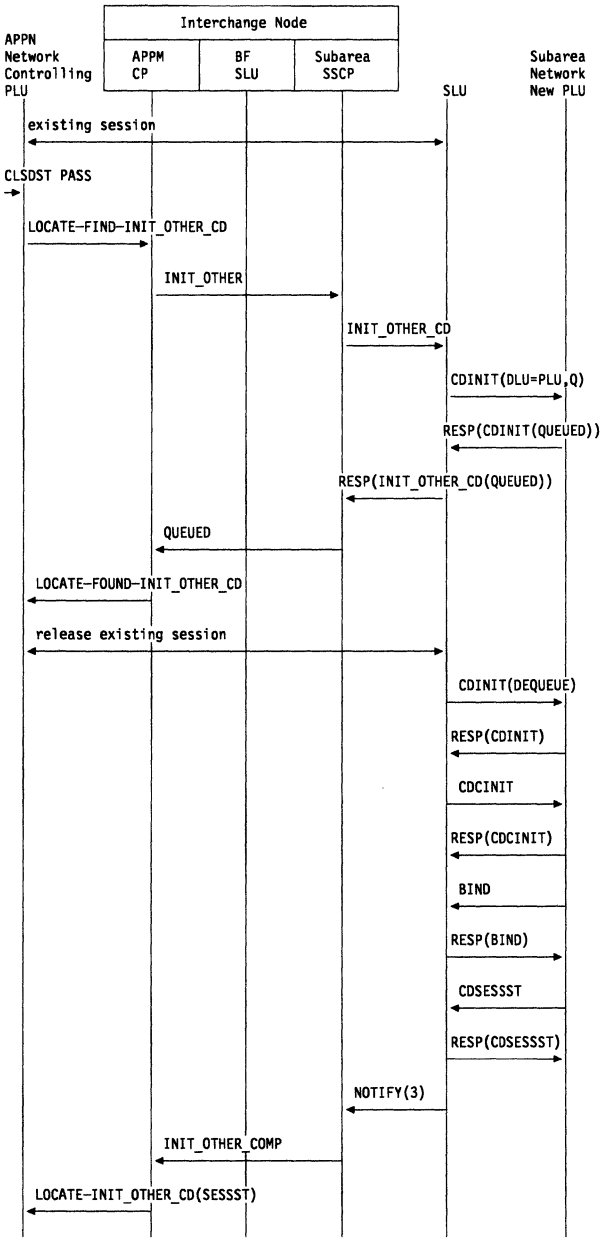


Figure 231. CLSDST PASS from APPN to Subarea. The SLU is single-session capable.

EN-NN-EN, PLU-Initiated, No Queueing

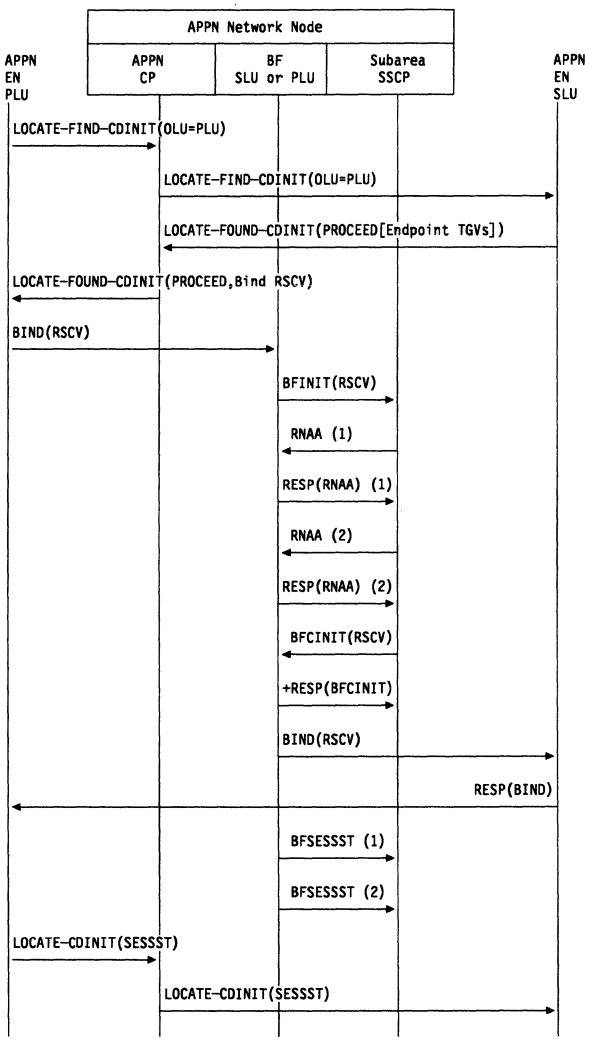


Figure 232. EN-NN-EN, PLU-Initiated, No Queueing (Including BIND Flows for Intermediate Network Node)

1. For the PLU side of the session.
2. For the SLU side of the session.

Intermediate Network Node (INN) BIND

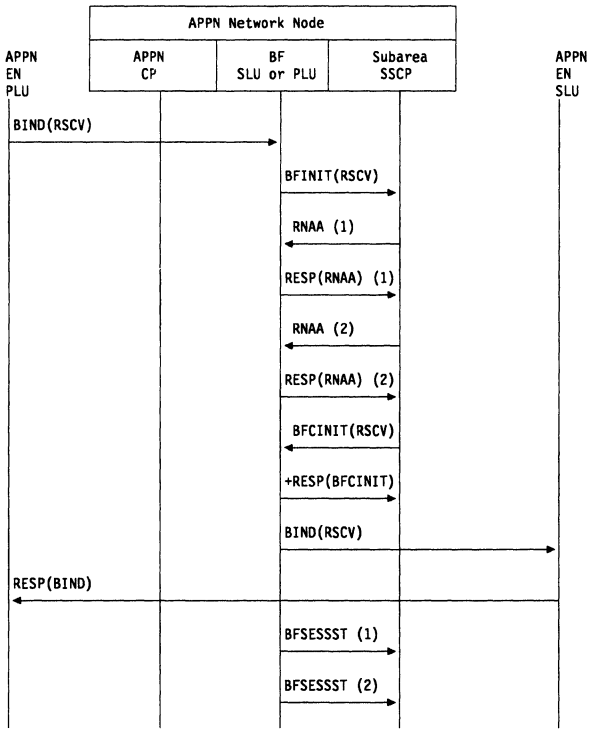


Figure 233. Intermediate Network Node (INN) BIND. The LOCATE did not go through this node.

1. For the PLU side of the session.
2. For the SLU side of the session.

APPN Network (PLU)...ICN==SA(SLU), PLU-Initiated, Search-Only Flow Transformed into a DSRLST

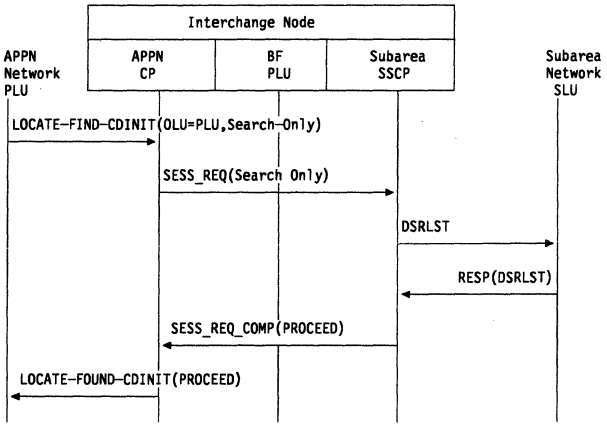


Figure 234. APPN Network (PLU)...ICN==SA(SLU), PLU-Initiated, Search-Only Flow Transformed into a DSRLST

APPN Network (PLU)...ICN==SA(SLU), PLU-Initiated, No Queueing

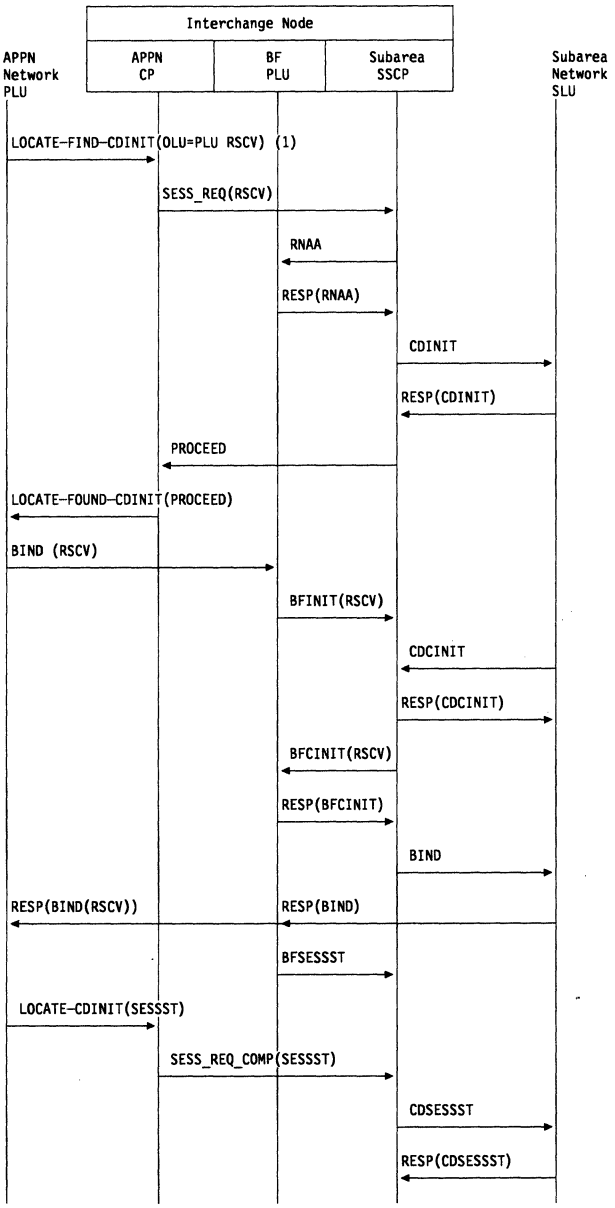


Figure 235. APPN Network (PLU)...ICN==SA(SLU), PLU-Initiated, No Queueing

1. Because the DLU is in subarea, the NNS(OLU) precomputed the RSCV.

APPN Network (PLU)...ICN==SA(SLU), PLU-Initiated, Directed Search without Required Precomputed RSCV

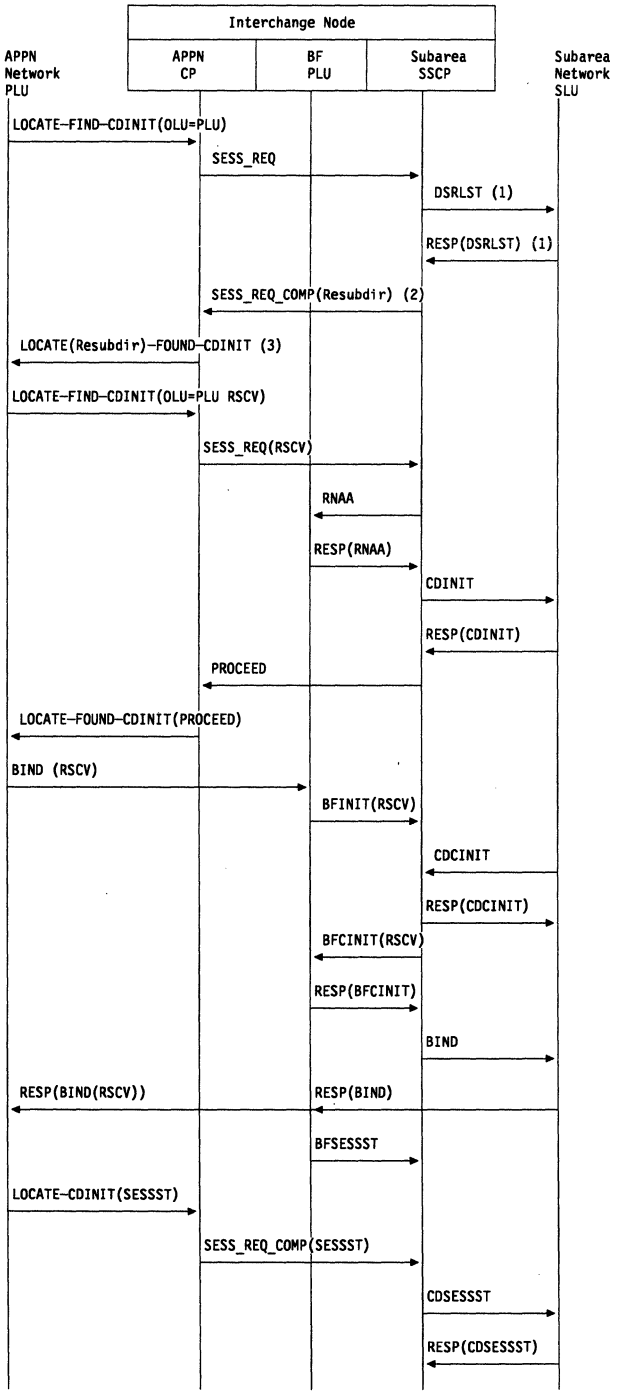


Figure 236. APPN Network (PLU)...ICN==SA(SLU), PLU-Initiated, Directed Search without Required Precomputed RSCV

1. Optional; sent if owning CP is not known.
2. Resubdir means "Resubmit Request on a Directed Search."
3. The interchange node returns the fact that the SLU is in a subarea network and requires a precomputed RSCV.

APPN Network (PLU)...ICN==SA(SLU), PLU-Initiated, USERVAR Resolution Required

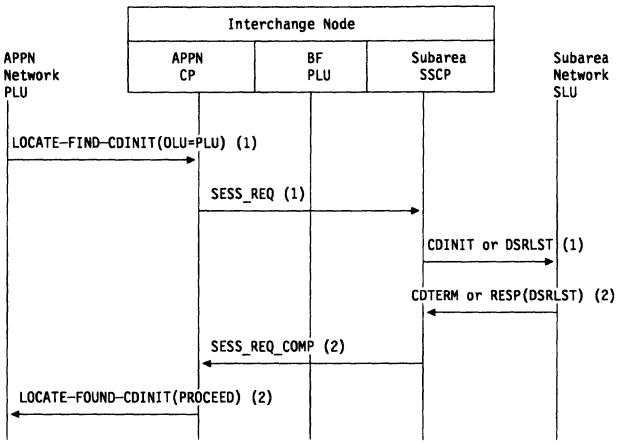


Figure 237. APPN Network (PLU)...ICN==SA(SLU), PLU-Initiated, USERVAR Resolution Required

1. Generic USERVAR name
2. Resolved USERVAR name

For remaining session setup flows, see Figure 235 on page 1089.

APPN Network (PLU)...ICN==SA(SLU), PLU-Initiated, Queued by the SLU

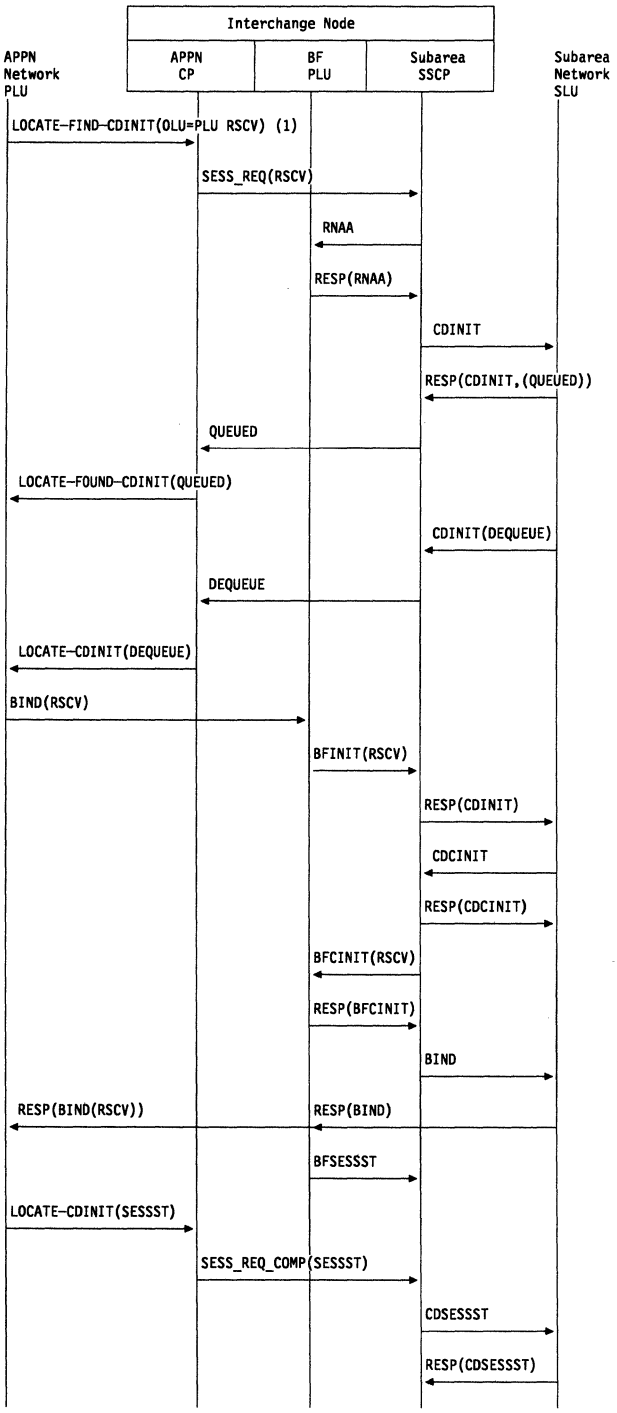


Figure 238. APPN Network (PLU)...ICN==SA(SLU), PLU-Initiated, Queued by the SLU

1. Because the DLU is in subarea, the NNS(OLU) precomputed the RSCV.

APPN Network (SLU)...ICN==SA(PLU), SLU-Initiated, No Queueing

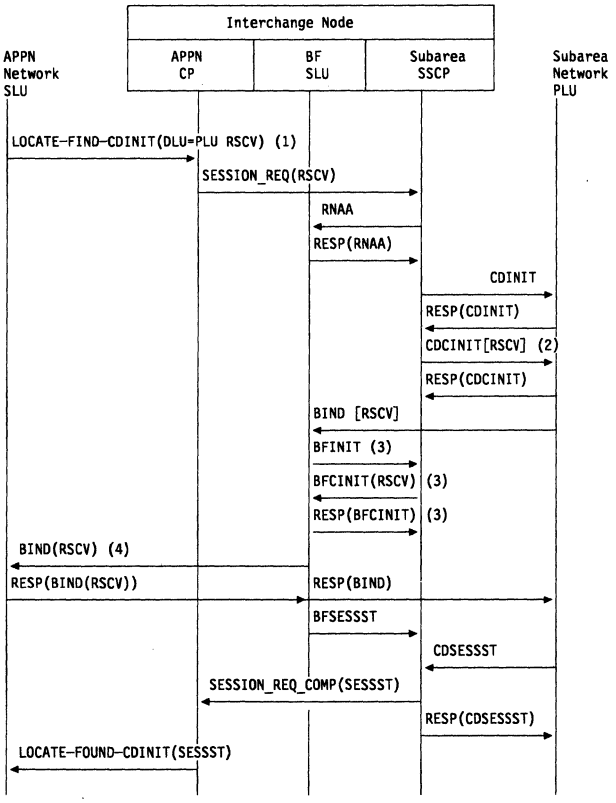


Figure 239. APPN Network (SLU)...ICN==SA(PLU), SLU-Initiated, No Queueing

1. Because the DLU is in subarea, the NNS(OLU) precomputes the RSCV.
2. If the adjacent SSCP toward the PLU is VTAM V4R1 or higher and has the same network identifier, the RSCV is passed on the CDCINIT.
3. If the RSCV is passed on the CDCINIT, these flows will not occur.
4. The BIND does not have to follow the same path as the LOCATE flows.

APPN Network (SLU)...ICN==SA(PLU), SLU-Initiated, Queued by the PLU

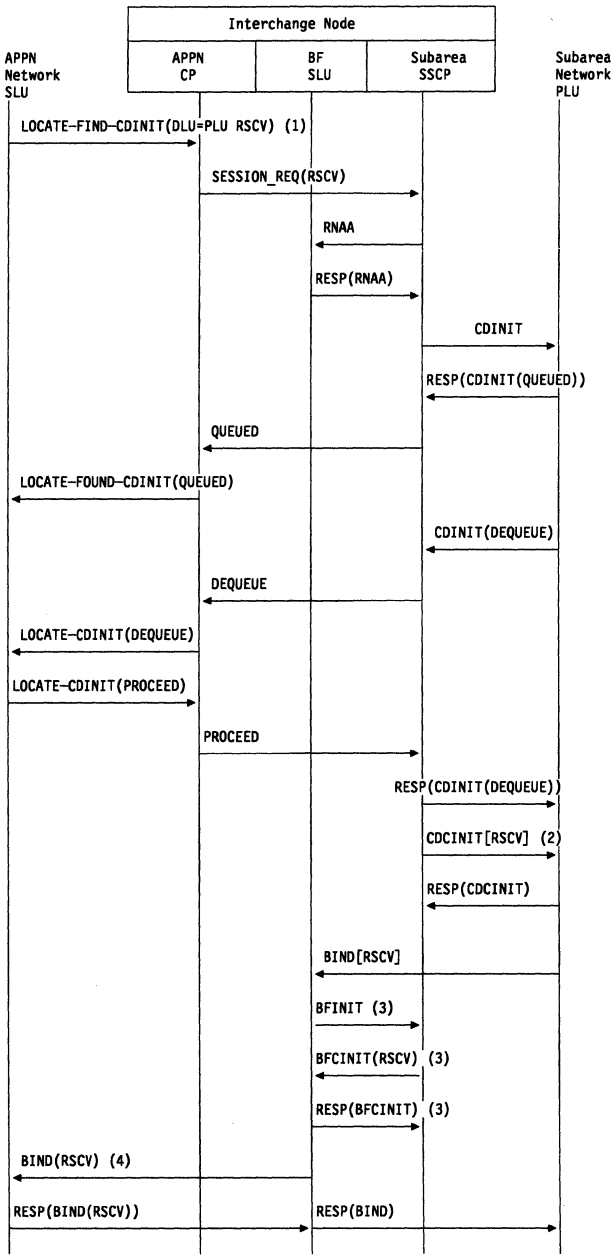


Figure 240 (Part 1 of 2). APPN Network (SLU)...ICN==SA(PLU), SLU-Initiated, Queued by the PLU

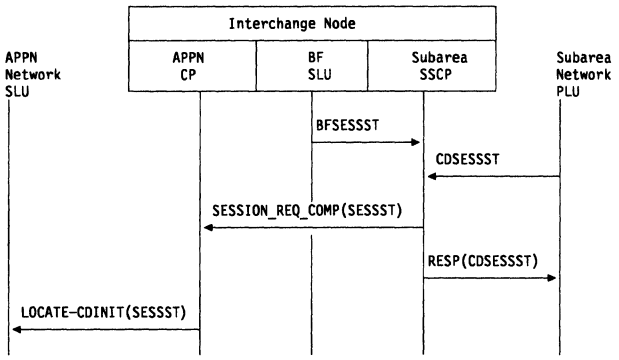


Figure 240 (Part 2 of 2). APPN Network (SLU)...ICN==SA(PLU), SLU-Initiated, Queued by the PLU

1. Because the DLU is in subarea, the NNS(OLU) precomputed the RSCV.
2. If the adjacent SSCP into the subarea is VTAM V4R1 or higher and has the same network identifier, the RSCV is passed on the CDCINIT.
3. If the RSCV is passed on the CDCINIT, these flows will not occur.
4. The BIND does not have to follow the same path as the LOCATE flows.

APPN Network (SLU)...ICN==SA(PLU), Autologon (PLU Not Available Initially)

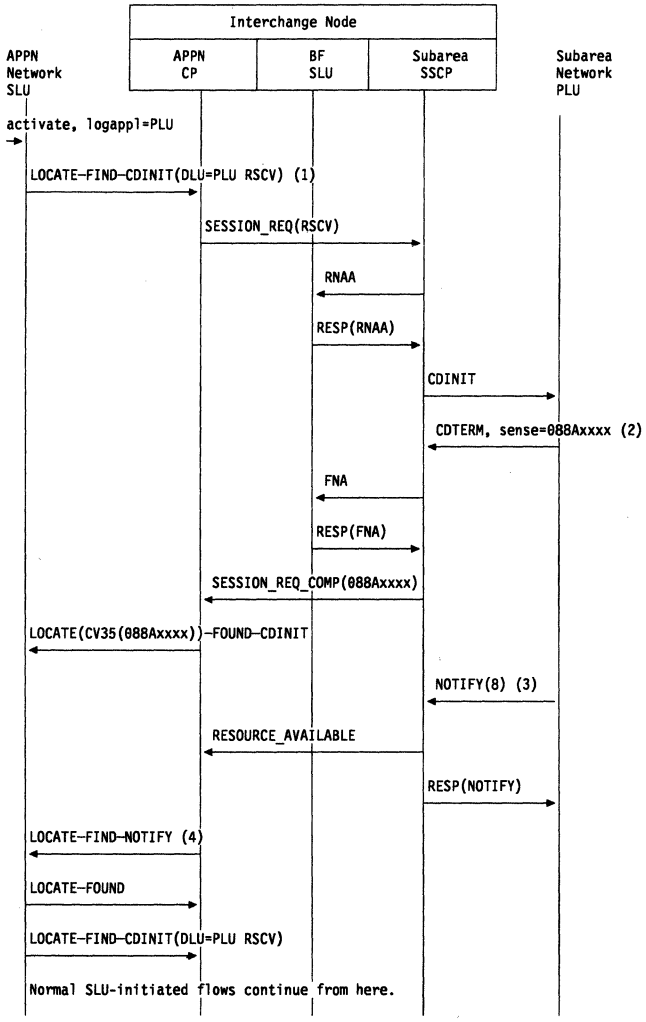


Figure 241. APPN Network (SLU)...ICN==SA(PLU), Autologon (PLU Not Available Initially)

1. Because the DLU is in subarea, the NNS(OLU) precomputed the session RSCV.
2. The PLU is not currently available
3. Some time later, the PLU becomes available.
4. Resource enabled.

APPN Network (PLU)...ICN==VR-based TG==ICN...APPN Network (SLU)

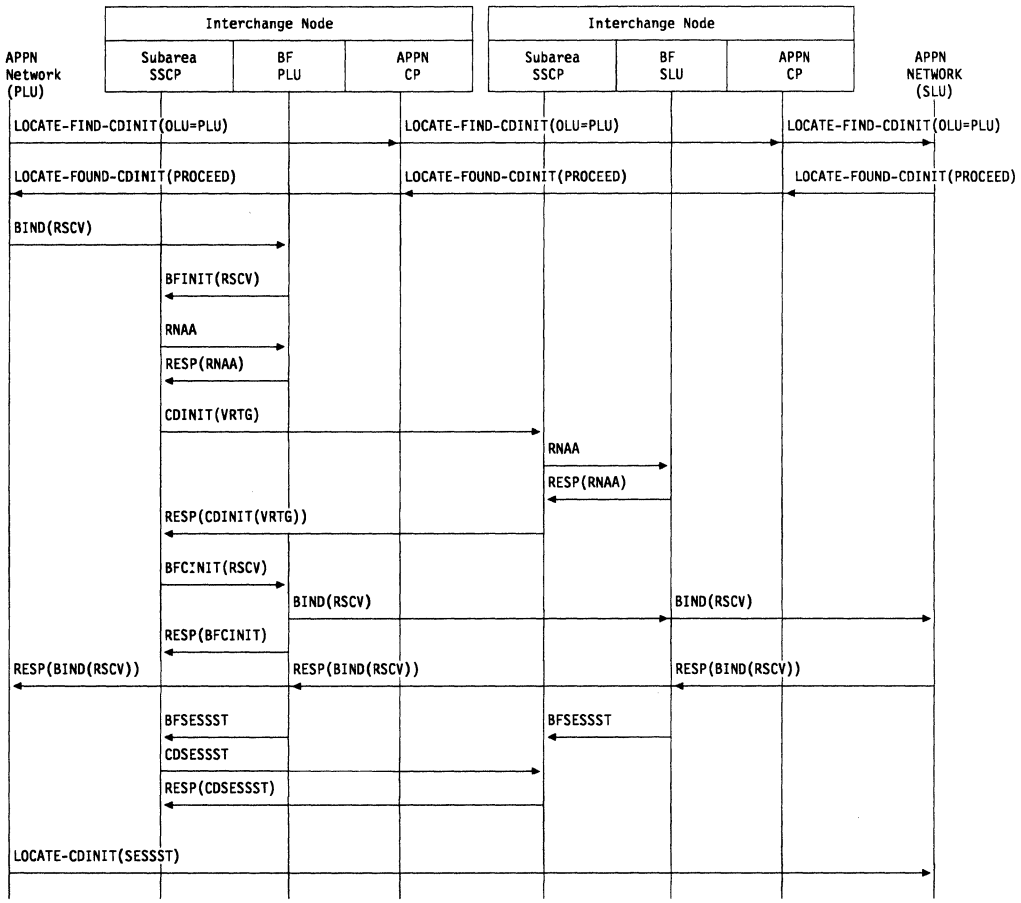


Figure 242. APPN Network (PLU)...ICN==VR-Based TG==ICN...APPN Network (SLU), PLU-Initiated

Dependent LU Server Flows

Figure 243 on page 1099 through Figure 263 on page 1123 illustrate the flow of requests and responses between dependent logical unit requestors and servers.

Index of Dependent LU Server Flows

Table 75 lists the dependent LU server flows illustrated here.

Table 75. Index of Dependent LU Server Flows

Single Subnetwork Flow	Page
Activating resources	
CPSVRMGR pipe activation, DLUR-Initiated	Figure 243 on page 1099
CPSVRMGR pipe activation, DLUS-initiated	Figure 244 on page 1100
Dependent LUs, dynamic registration and activation of	Figure 246 on page 1102
Dependent LUs, activation of pre-defined	Figure 247 on page 1103
Physical Unit (PU), dynamic activation of	Figure 245 on page 1101
SSCP-PU session activation race	Figure 248 on page 1104
Deactivating resources	
CPSVRMGR pipe deactivation	Figure 249 on page 1105
Downstream PU outage	Figure 250 on page 1106
REQDISCONT (immediate) received from downstream PU	Figure 252 on page 1108
REQDISCONT (normal) received from downstream PU	Figure 251 on page 1107
LU-LU sessions	
APPN PLU-initiated to a dependent SLU	Figure 257 on page 1113
Session termination, USS flows for	Figure 260 on page 1116
USS SLU-initiated to APPN PLU	Figure 258 on page 1114
USS SLU-initiated to subarea PLU	Figure 259 on page 1115
SSCP-PU, SSCP-LU session deactivation	
Forced	Figure 254 on page 1110
Normal	Figure 253 on page 1109
With Giveback (ANS=CONT)	Figure 256 on page 1112
With Giveback (ANS=STOP)	Figure 255 on page 1111
Cross Subnetwork Flow	**Page**
PLU-Initiated Session with DLUS and DLUR within Different Subnetworks	Figure 261 on page 1118
PLU-Initiated Session with DLUS and PLU in one Subnetwork and DLUR in Another	Figure 262 on page 1121
SLU-Initiated Session with DLUS and DLUR within Different Subnetworks	Figure 263 on page 1123

Single Subnetwork Flows

Figure 243 through Figure 260 on page 1116 show the flow of requests and responses between dependent logical unit requestors and servers within a single subnetwork.

DLUR-Initiated CPSVRMGR Pipe Activation

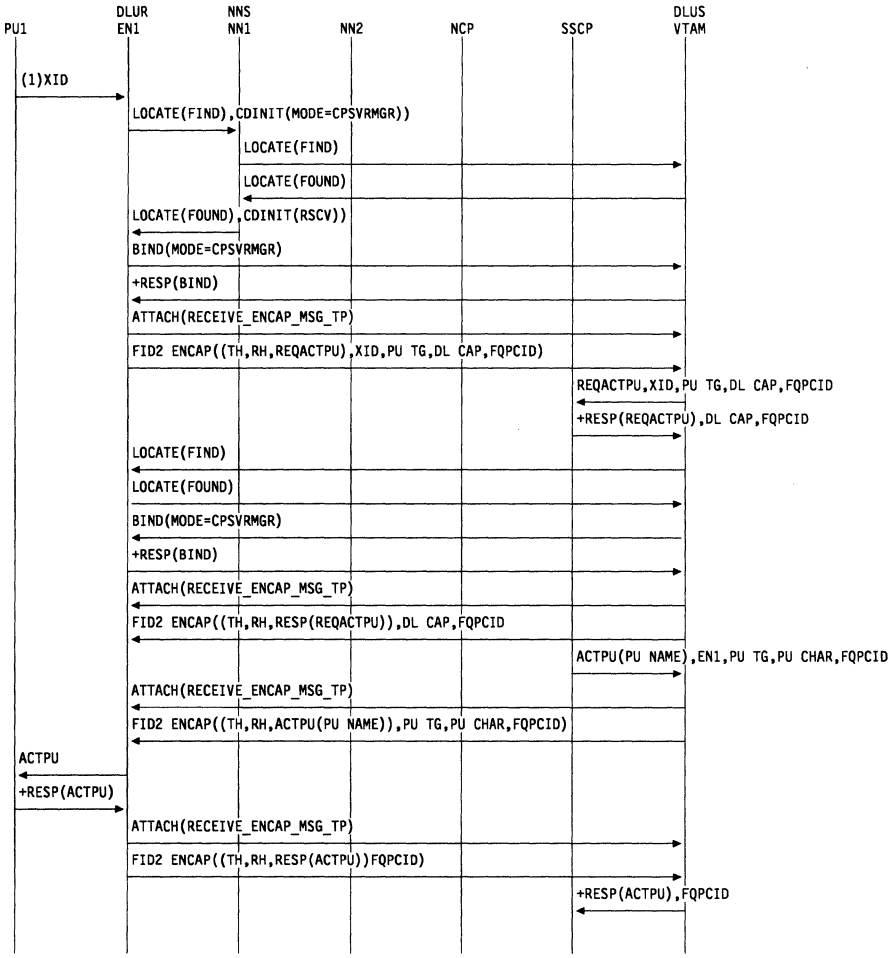


Figure 243. DLUR-Initiated CPSVRMGR Pipe Activation

1. XID flows as a result of a set normal response mode (SNRM) RU, an external command, or an internal activation signal.

DLUS-Initiated CPSVRMGR Pipe Activation

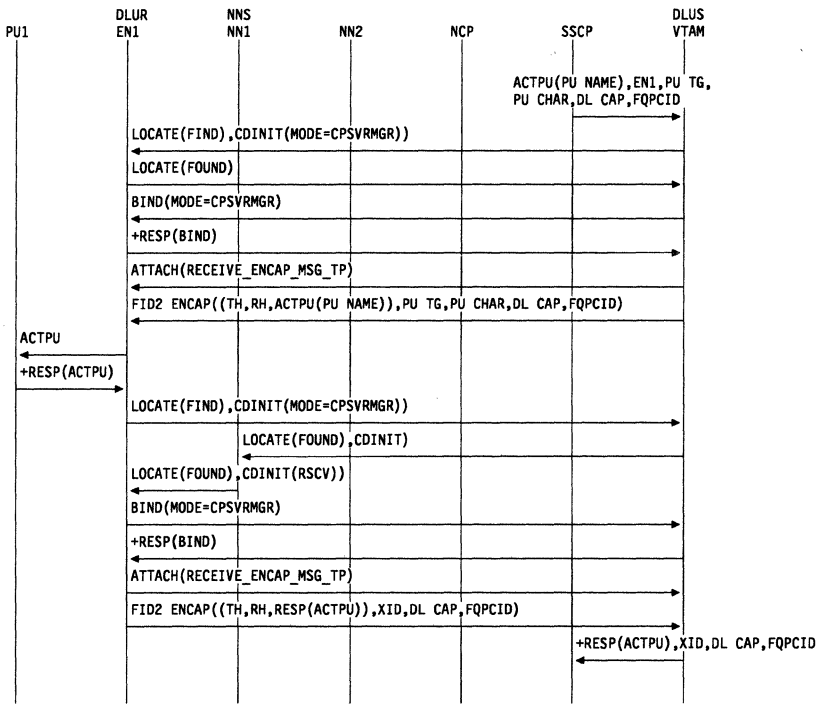


Figure 244. DLUS-Initiated CPSVRMGR Pipe Activation

Dynamic PU Activation

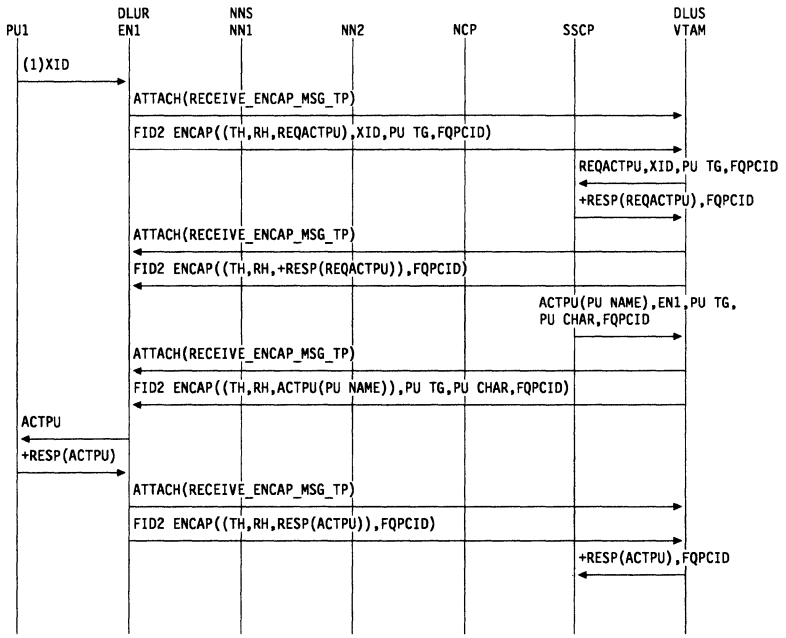


Figure 245. Dynamic PU Activation

1. XID flows as a result of a set normal response mode (SNRM) RU, an external command, or an internal activation signal.

Dynamic Registration and Activation of Dependent LUs

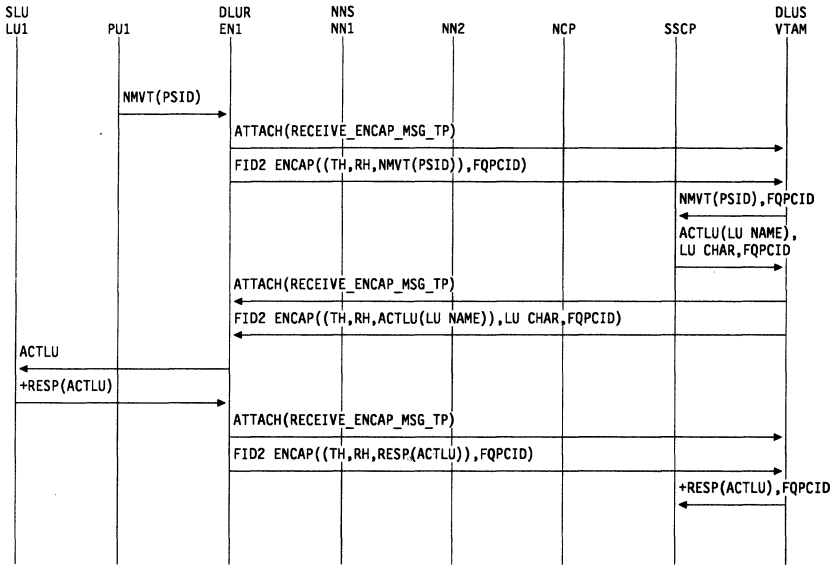


Figure 246. Dynamic Registration and Activation of Dependent LUs

Activation of Pre-Defined Dependent LUs

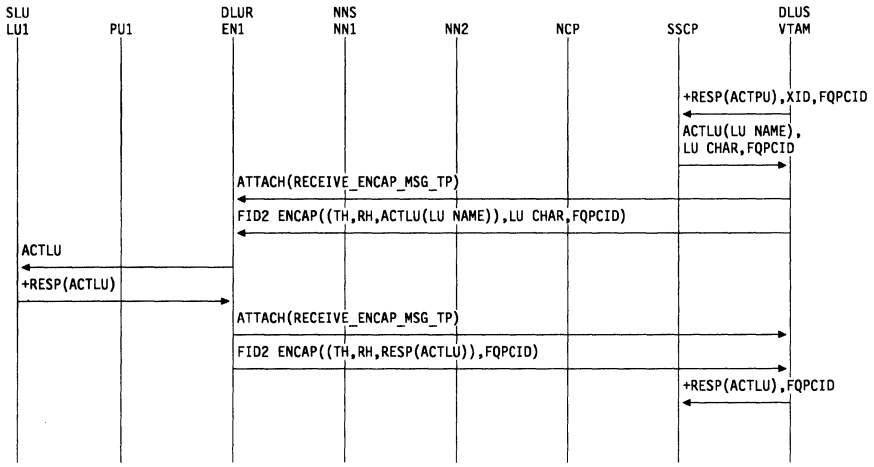


Figure 247. Activation of Pre-Defined Dependent LUs

SSCP-PU Session Activation Race

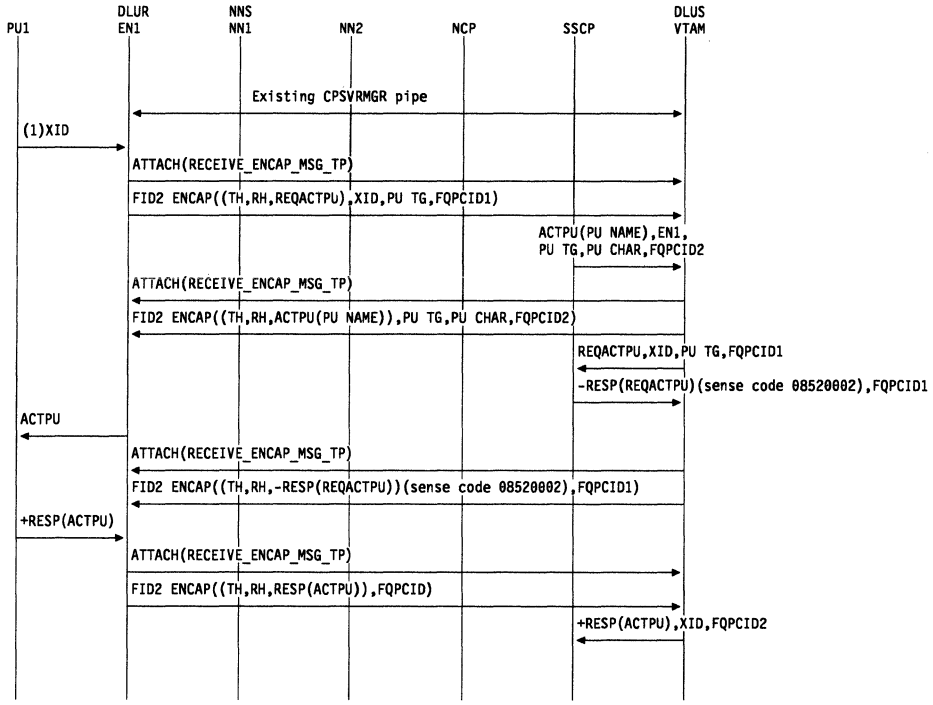


Figure 248. SSCP-PU Session Activation Race

1. XID flows as a result of a set normal response mode (SNRM) RU, an external command, or an internal activation signal.

CPSVRMGR Pipe Deactivation

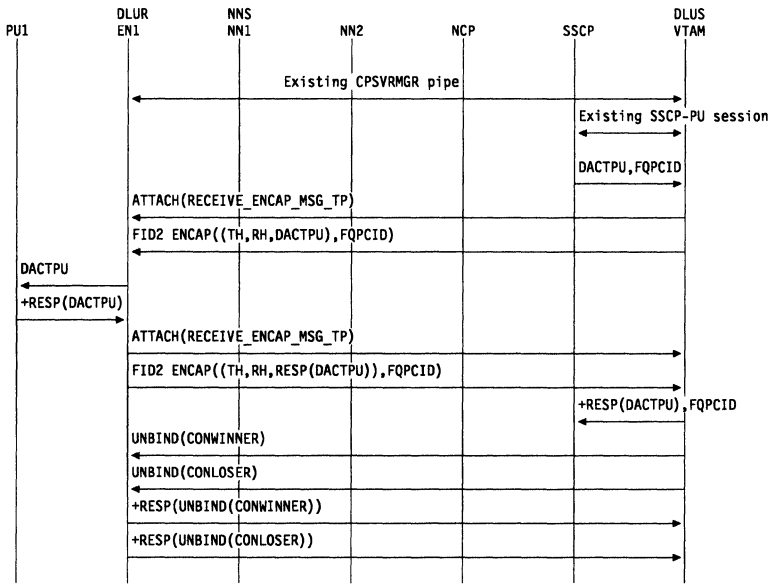


Figure 249. CPSVRMGR Pipe Deactivation

Downstream PU Outage

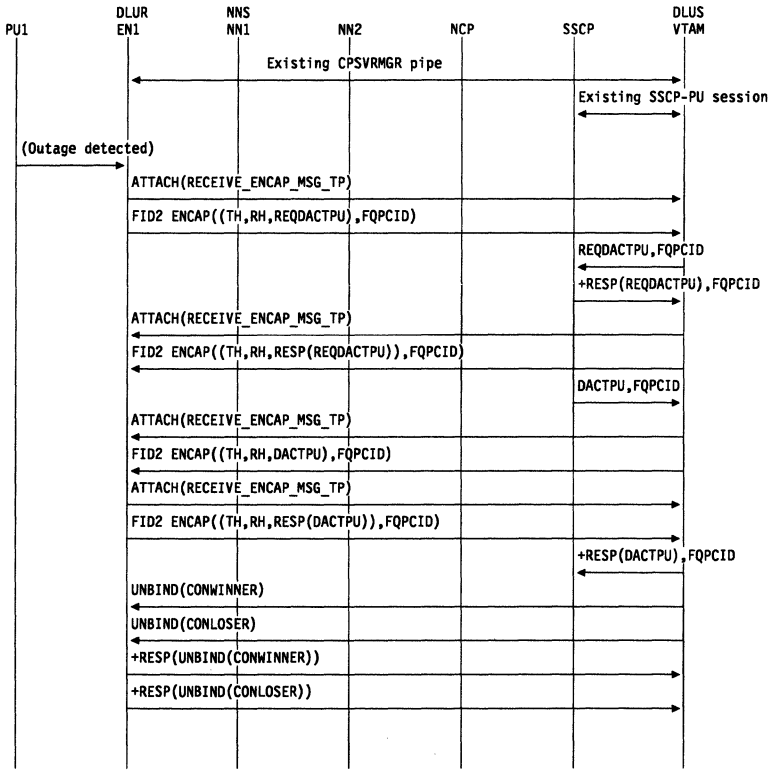


Figure 250. Downstream PU Outage

Receipt of REQDISCONT (Normal) from Downstream PU

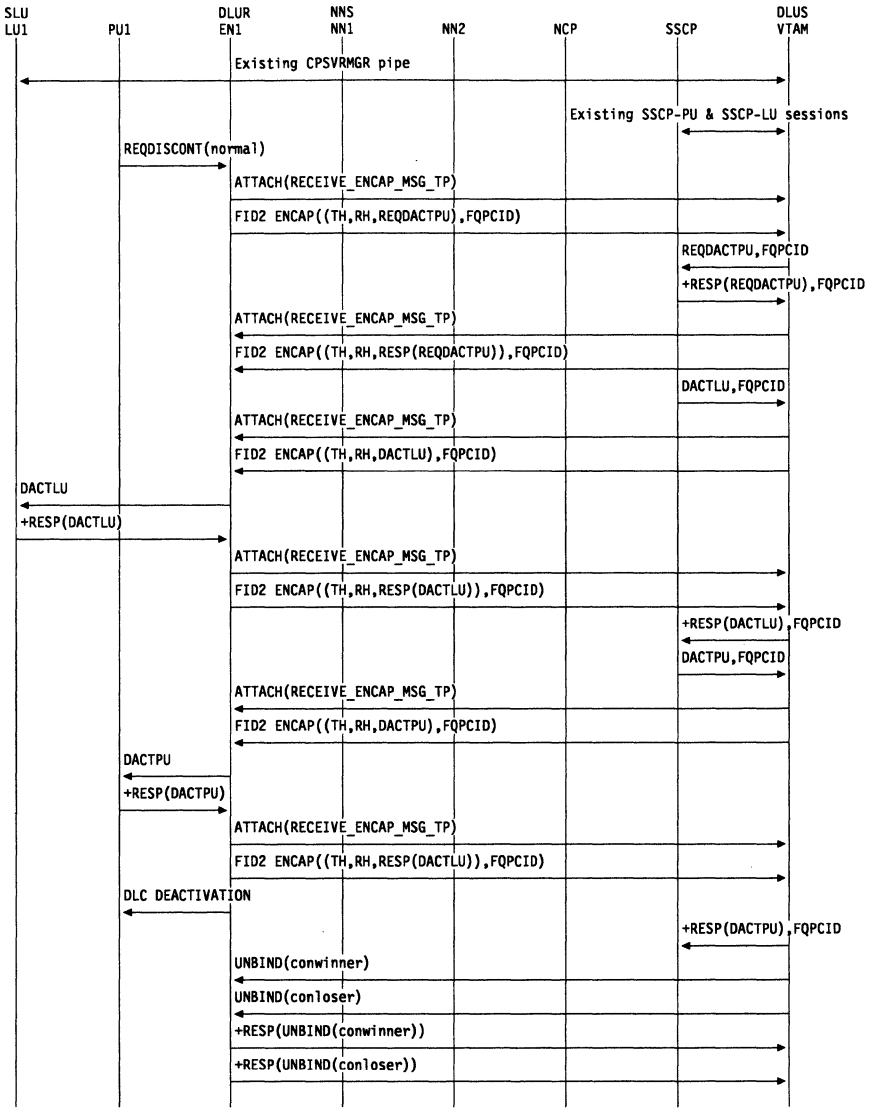


Figure 251. Receipt of REQDISCONT (Normal) from Downstream PU

Receipt of REQDISCONT (Immediate) from Downstream PU

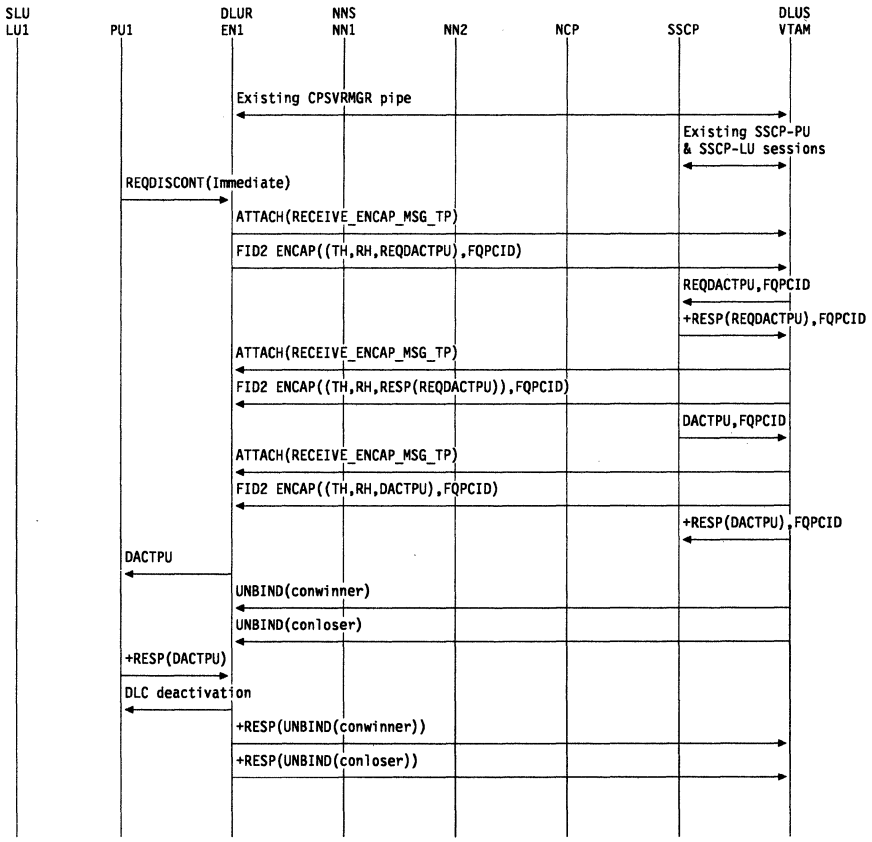


Figure 252. Receipt of REQDISCONT (immediate) from Downstream PU

Normal SSCP-PU/SSCP-LU Session Deactivation

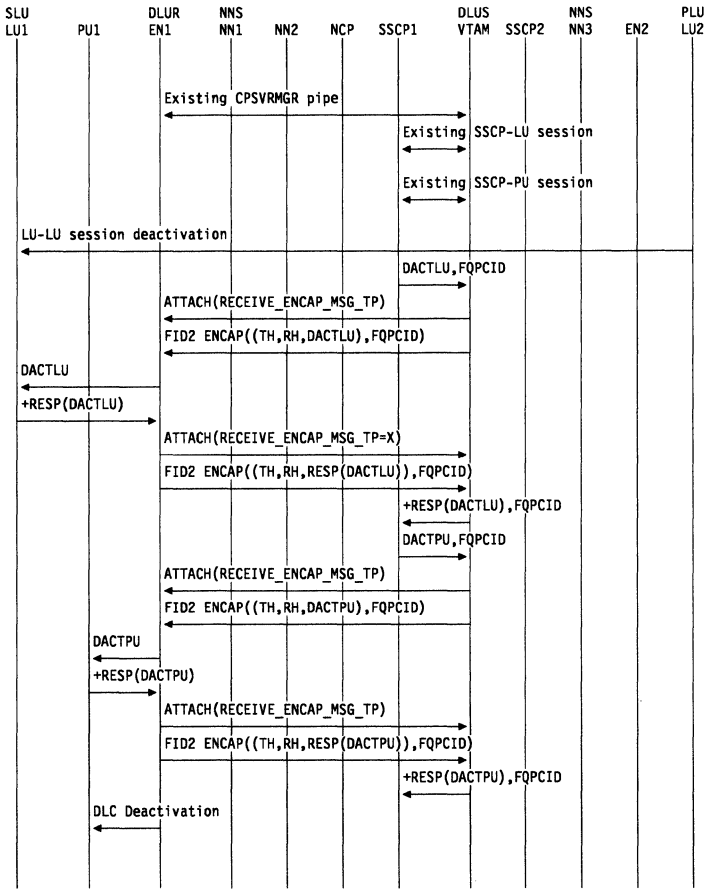


Figure 253. Normal SSCP-PU/SSCP-LU Session Deactivation

Forced SSCP-PU/SSCP-LU Session Deactivation

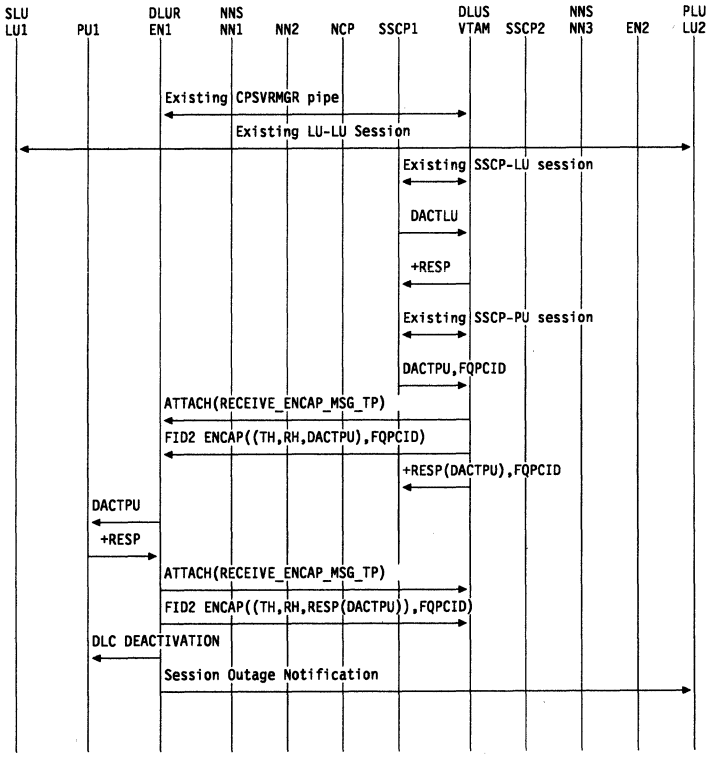


Figure 254. Forced SSCP-PU/SSCP-LU Session Deactivation

Giveback SSCP-PU/SSCP-LU Session Deactivation (ANS=STOP)

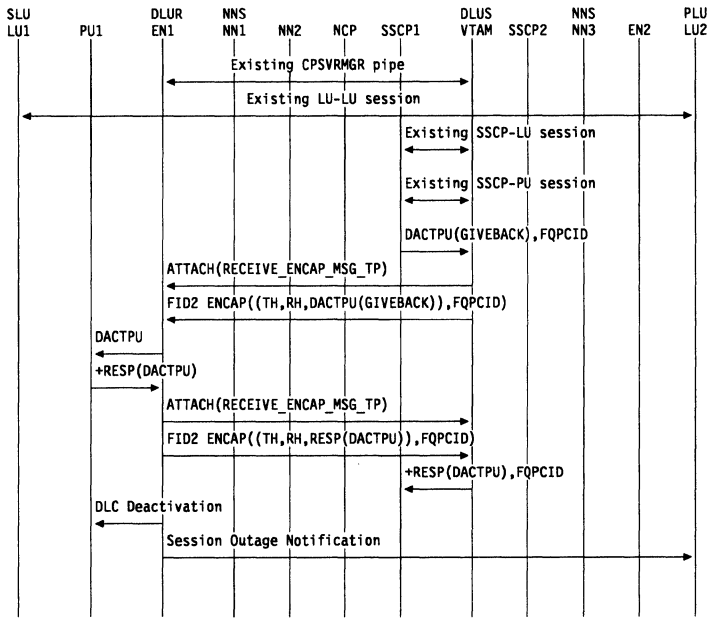


Figure 255. Giveback SSCP-PU/SSCP-LU Session Deactivation (ANS=STOP)

Giveback SSCP-PU/SSCP-LU Session Deactivation (ANS=CONT)

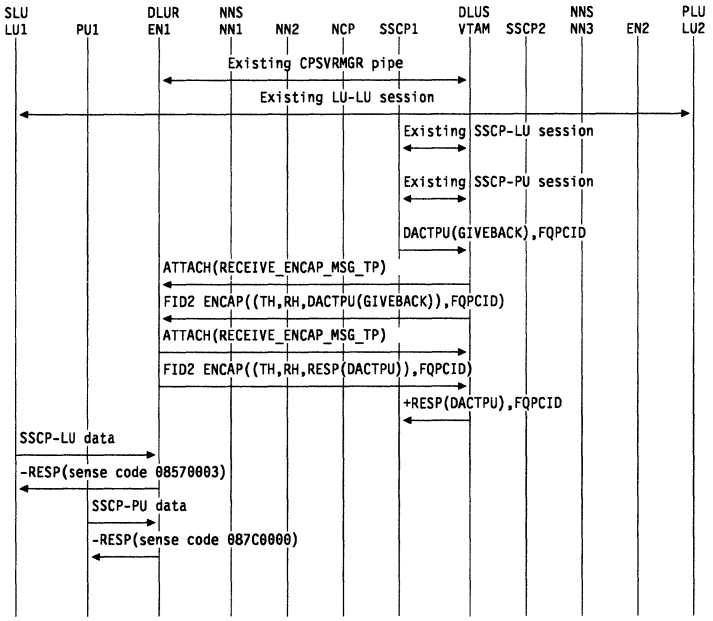


Figure 256. Giveback SSCP-PU/SSCP-LU Session Deactivation (ANS=CONT)

APPN PLU-Initiated LU-LU Session to a Dependent SLU

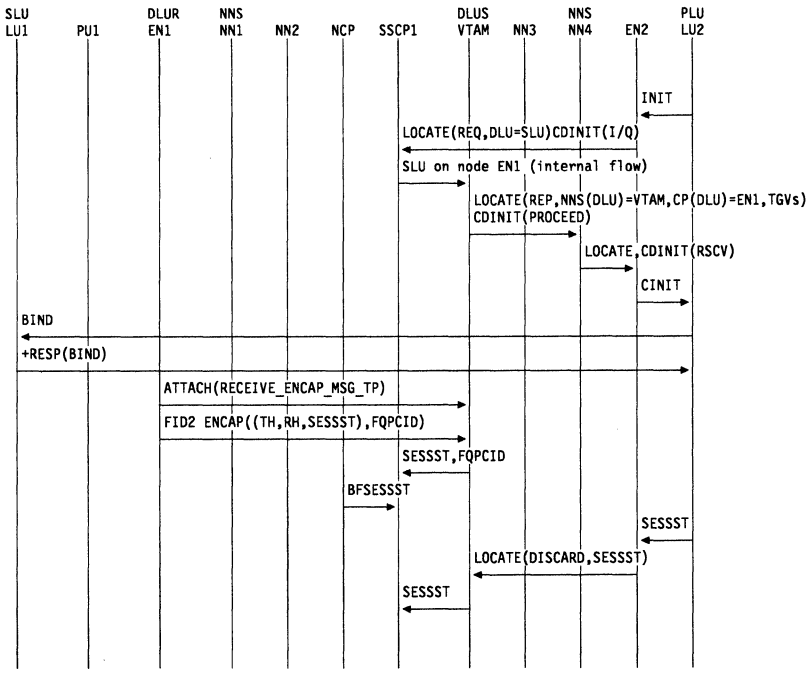


Figure 257. APPN PLU-Initiated LU-LU Session to a Dependent SLU

Note: The transmission group (TG) vectors of the end node dependent LU requestor are provided by previous TG vector registration over the CPSVRMGR pipe.

USS SLU-Initiated LU-LU Session to APPN PLU

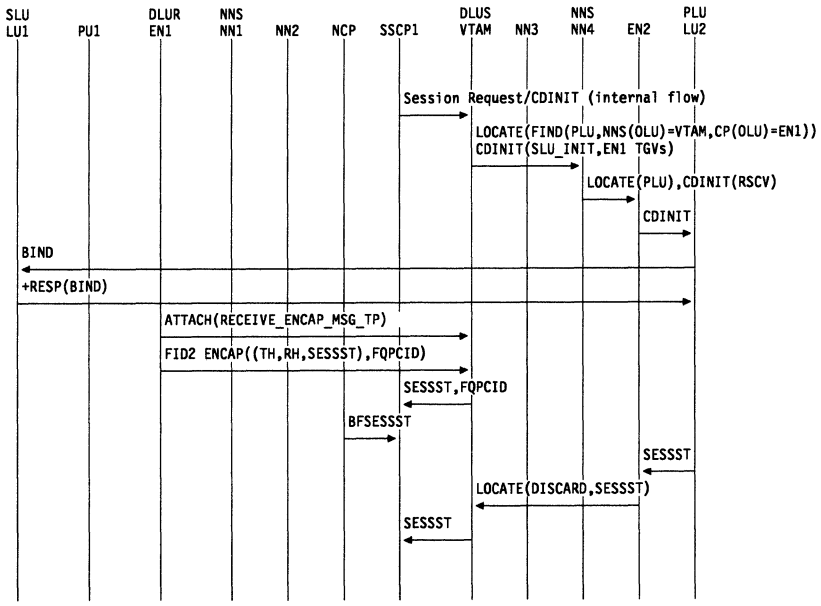


Figure 258. USS SLU-Initiated LU-LU Session to APPN PLU

Note: The transmission group (TG) vectors of the end node dependent LU requestor are provided by previous TG vector registration over the CPSVRMGR pipe.

USS SLU-Initiated LU-LU Session to Subarea PLU

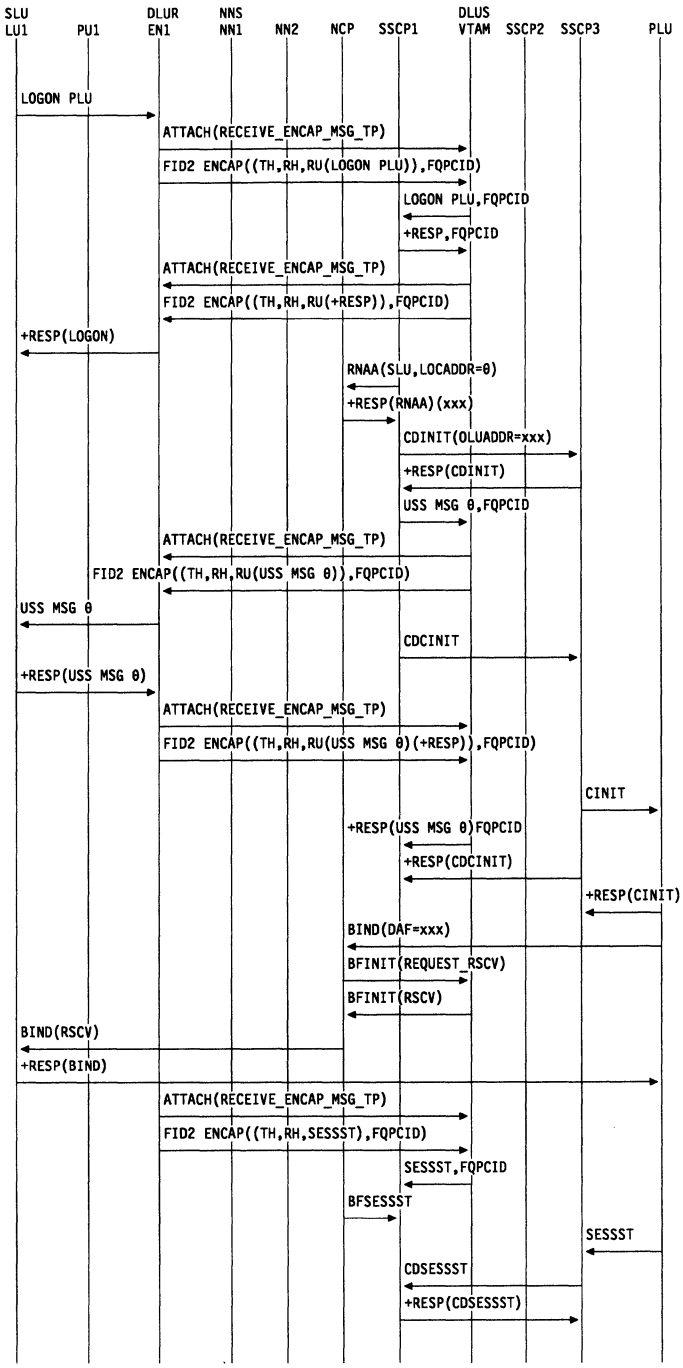


Figure 259. USS SLU-Initiated LU-LU Session to Subarea PLU

USS Flows for LU-LU Session Termination

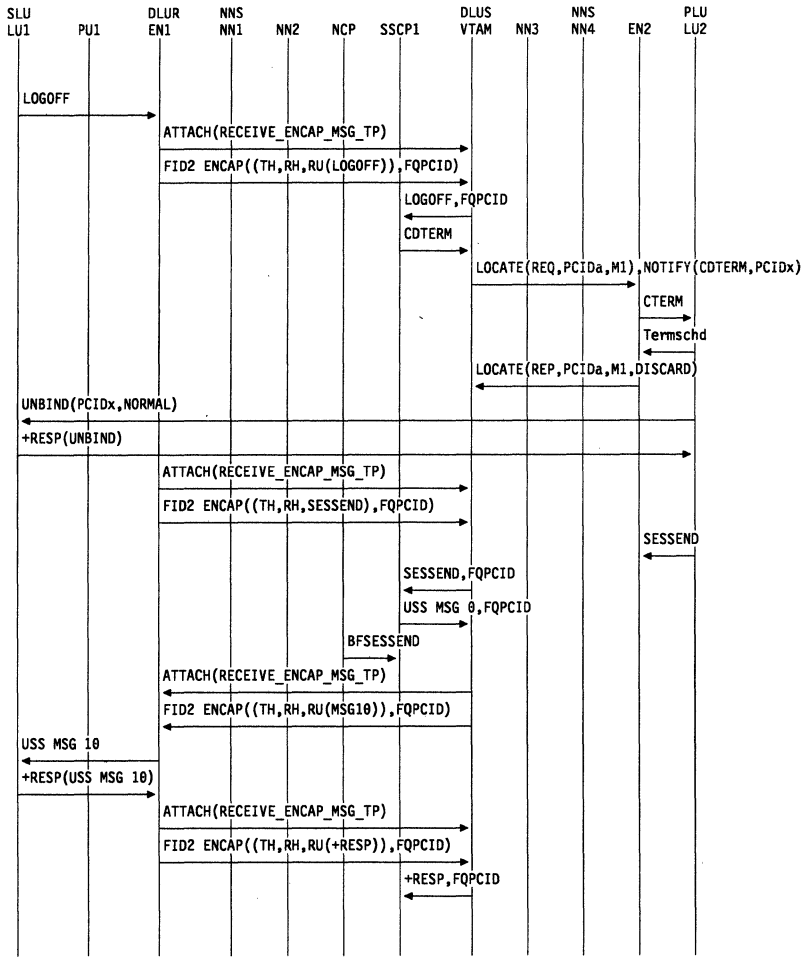


Figure 260. USS Flows for LU-LU Session Termination

Note: The transmission group (TG) vectors of the end node dependent LU requestor are provided by previous TG vector registration over the CPSVRMGR pipe.

Cross Subnetwork Flows

Figure 261 on page 1118 through Figure 263 on page 1123 show the flow of requests and responses between dependent logical unit requestors and servers across subnetworks.

Several abbreviations are used in these flow diagrams.

DSL DLUS-served LU
DSR DLUR search required
ISB Internet search bit
OCR Owning CP respond

PLU-Initiated Session with DLUS and DLUR within Different Subnetworks, PLU Is Through the Subarea

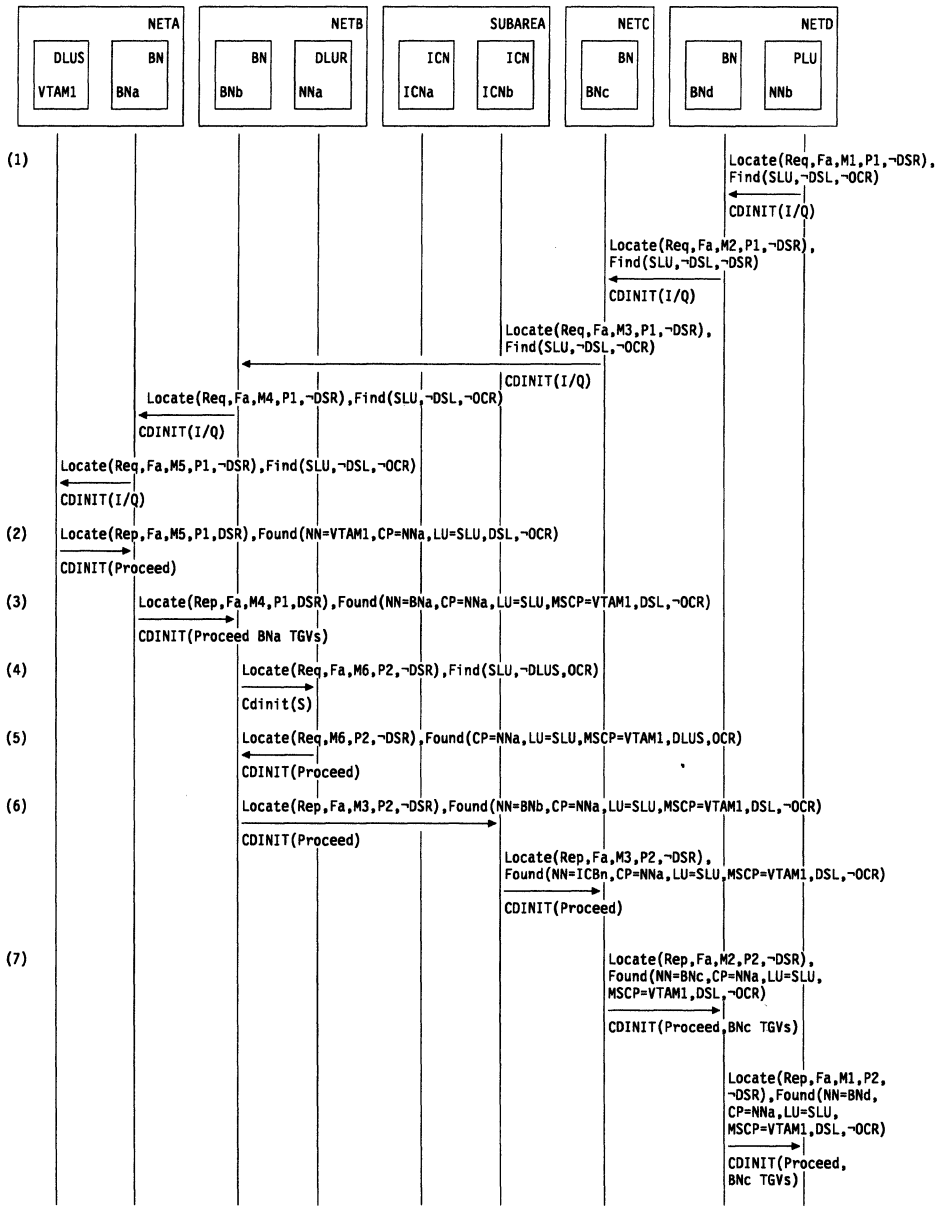


Figure 261 (Part 1 of 2). PLU-Initiated Search with DLUS and DLUR within Different Subnetworks, PLU Through the Subarea

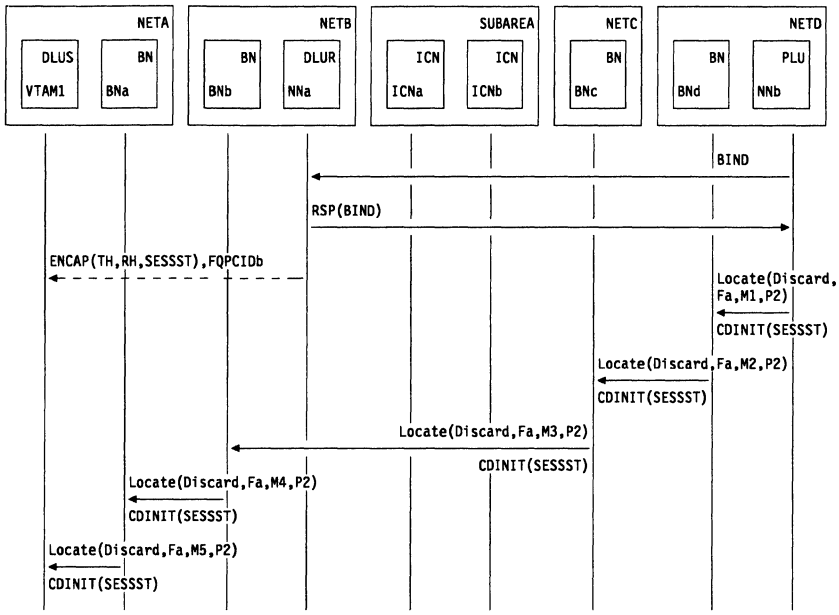


Figure 261 (Part 2 of 2). PLU-Initiated Search with DLUS and DLUR within Different Subnetworks, PLU Through the Subarea

1. NNb initiates a search to locate the SLU. NNb has the location of the SLU cached, and the hierarchy indicates that BNd is the NNS(SLU). NNb sends a directed Locate to BNd to verify the location of the SLU and to obtain the SLU TGVs.

   BNd initiates a search to locate the SLU. The search ultimately reaches VTAM1.

   Neither the DSL nor DSR indicators will be set within the Find GDS variable on the search sent by NNb.

2. The OCR indicator is not set, so VTAM1 responds to the Locate. Since NNa is within a different subnet than VTAM1, and the PLU is also non-native, VTAM1 will set the DSR indicator on the Locate reply. VTAM1 will also set the DSL indicator on the Locate reply since the SLU is a DLUS-served LU.

3. BNa caches the location of the SLU as being on NNa with VTAM1 as the NNS. Since the OCR indicator is not set on the reply, BNa does not set the OCR indicator within the cache entry. BNa then modifies the Found resource hierarchy to indicate itself as the NNS(DLU) and VTAM1 as the Management Services Control Point (MSCP). BNa also adds its own TGVs to the Locate reply.

4. BNb caches the location of the SLU as being on NNa with BNa as the NNS. Since the OCR indicator is not set on the reply, BNb does not set the OCR indicator within the cache entry.

   Since the DSR and DSL indicators are set on the reply and BNb is returning a reply to a non-Border Node, BNb must obtain the SLU TGs to be included on the Locate reply which will be returned to the NNS(PLU).

   To obtain the TGVs, BNb initiates a Locate search to find the SLU. This search will be a PLU-init Search Only. The OCR indicator is set, requesting that the DLUR node respond to the Locate request. The DSR indicator will not be set, since BNb is the node which is performing the extra Locate search.

5. The OCR indicator is set, so NNa responds to the Locate. Since the SLU is a DLUS-served LU, NNa sets the DSL indicator on the reply. When building the reply, NNa will include a CV X'40' which includes the DLUS node's CP name.
6. BNb caches the location of the SLU as being on NNa. Since both the DSL and OCR indicators are set on the Locate reply, BNb sets the OCR indicator within the cache entry.

   BNb modifies the Found resource hierarchy to indicate itself as the NNS(DLU). BNb removes BNa's TGVs from the reply. Since NNa is a network node, there are no TGVs to add to the Locate reply which is forwarded to ICNa.
7. Since the DSR indicator is not set on the reply, neither BNc nor BNd will submit an extra Locate search to obtain the SLU TGVs even though each is closer to the PLU than BNb.

   When NNb receives the Locate reply, NNb calculates an RSCV and sends the BIND to NNa. This establishes the session between the PLU and the SLU.

PLU-Initiated Session with DLUS and PLU in one Subnetwork and DLUR in Another

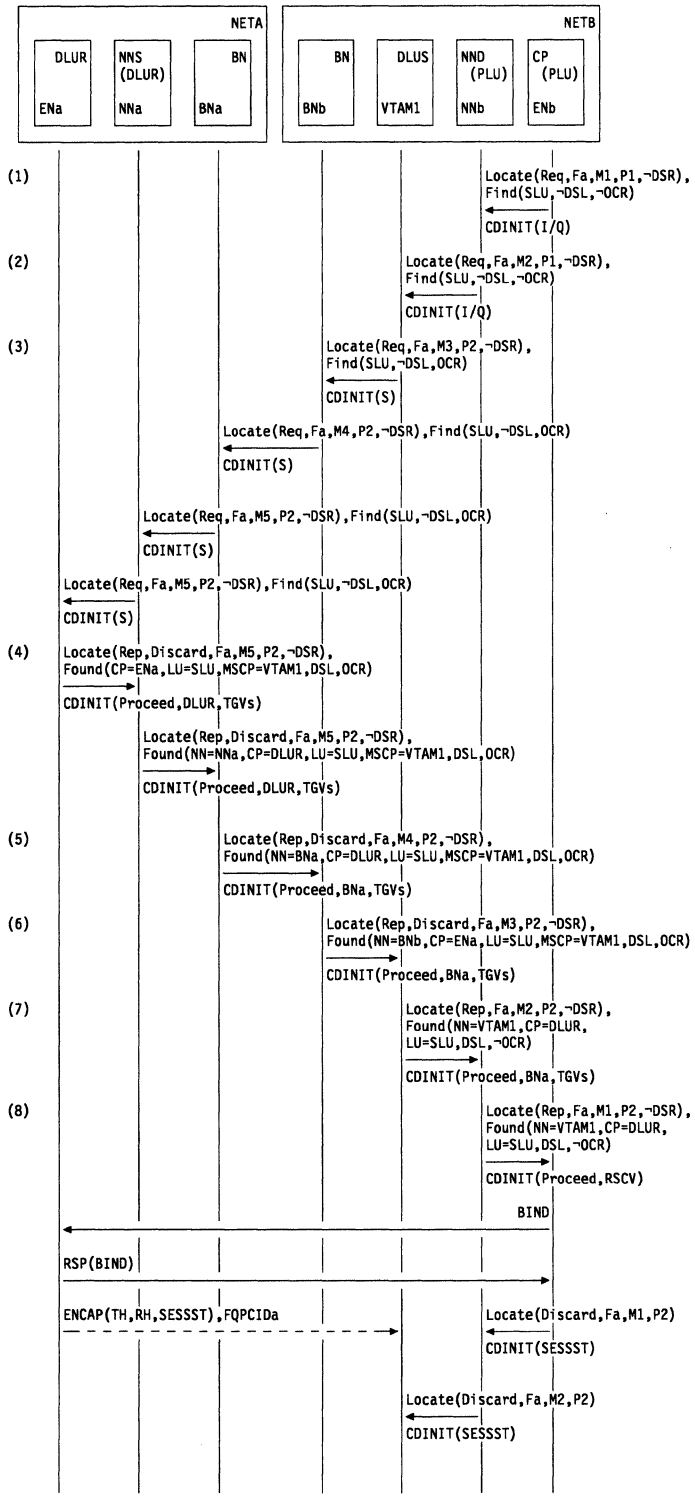


Figure 262. PLU-Initiated Session with DLUS and PLU in Same Subnetwork and DLUR in Another

1. ENb initiates a search to locate the SLU. The search is forwarded to ENb's Network Node Server. Neither the DSL, DSR, nor OCR indicators will be set within the Find GDS variable.
2. NNb initiates a search to locate the SLU. NNb has the location of the SLU cached, and the hierarchy indicates that VTAM1 is the NNS(SLU). NNb sends a directed Locate to VTAM1 to verify the location of the SLU and to obtain the SLU TGVs.
3. The OCR indicator is not set, so VTAM1 responds to the locate. Since ENa is in a different subnet than VTAM1 no endpoint TGVs were reported over the CPSVRMGR pipe. Since NNb is within the same APPN subnet as VTAM1, VTAM1 must obtain the endpoint TGVs to be included in the Locate reply returned to NNb.

   To obtain the endpoint TGVs, VTAM1 initiates a new Locate search to find the SLU. This search will be a PLU-init Search-Only. The OCR indicator will be set, requesting that the DLUR node respond to the Locate request. The DSR indicator will not be set, since VTAM1 is the node which is performing the extra Locate search.

   When initiating the Locate search, a new PCID modifier slot will be allocated and the PRN will be incremented. This will allow the Locate search to appear as a new search within both the APPN and subarea networks.

   When searching their caches for the SLU, VTAM1, BNb, and BNa all look for entries where the OCR indicator is set. These entries will allow the nodes to route the Locate search to the DLUR node instead of the DLUS node.
4. The OCR indicator is set, so ENa responds to the Locate. Since the SLU is a DLUS-served LU, ENa sets the DSL indicator. When building the reply, ENa will include a CV X'40' which includes the DLUS node's CP name.
5. BNa caches the location of the SLU as being on the DLUR with NNa as the Network Node Server. Since both the DSL and OCR indicators are set on the search reply, BNa sets a OCR indicator within the cache entry. BNa then modifies the Found resource hierarchy to indicate itself as the NNS(DLU). BNa also replaces the DLUR TGVs with its own TGVs before forwarding the Found to BNb.
6. BNb caches the location of the SLU, with BNa as the NNS. As with BNa, a OCR indicator is saved with the cache entry. BNb then modifies the resource hierarchy so that it appears as the NNS(DLU). The Locate reply is then forwarded to VTAM1.
7. VTAM1 caches the location of the SLU. As with BNa and BNb, VTAM1 sets the OCR indicator within the cache entry. VTAM1 then replies to the Locate search request received from NNb. Since the SLU is a DLUS-served LU, VTAM1 alters the Found hierarchy to indicate that VTAM1 is the NNS(DLU) and ENa is the CP(DLU). VTAM1 then removes the TGVs returned on the Locate reply received from BNb and places the TGVs on the Locate reply which it is constructing. Since the SLU is a DLUS-served resource, the DSL indicator is set. However, the DSR indicator is not set since VTAM1 has already obtained the correct DLUR TGVs.
8. NNb calculates an RSCV using the endpoint TGVs returned by VTAM1. NNb returns the RSCV to ENb on the Locate reply. ENb then BINDs the session between the PLU and the SLU.

SLU-Initiated Session with DLUS and DLUR within Different Subnetworks

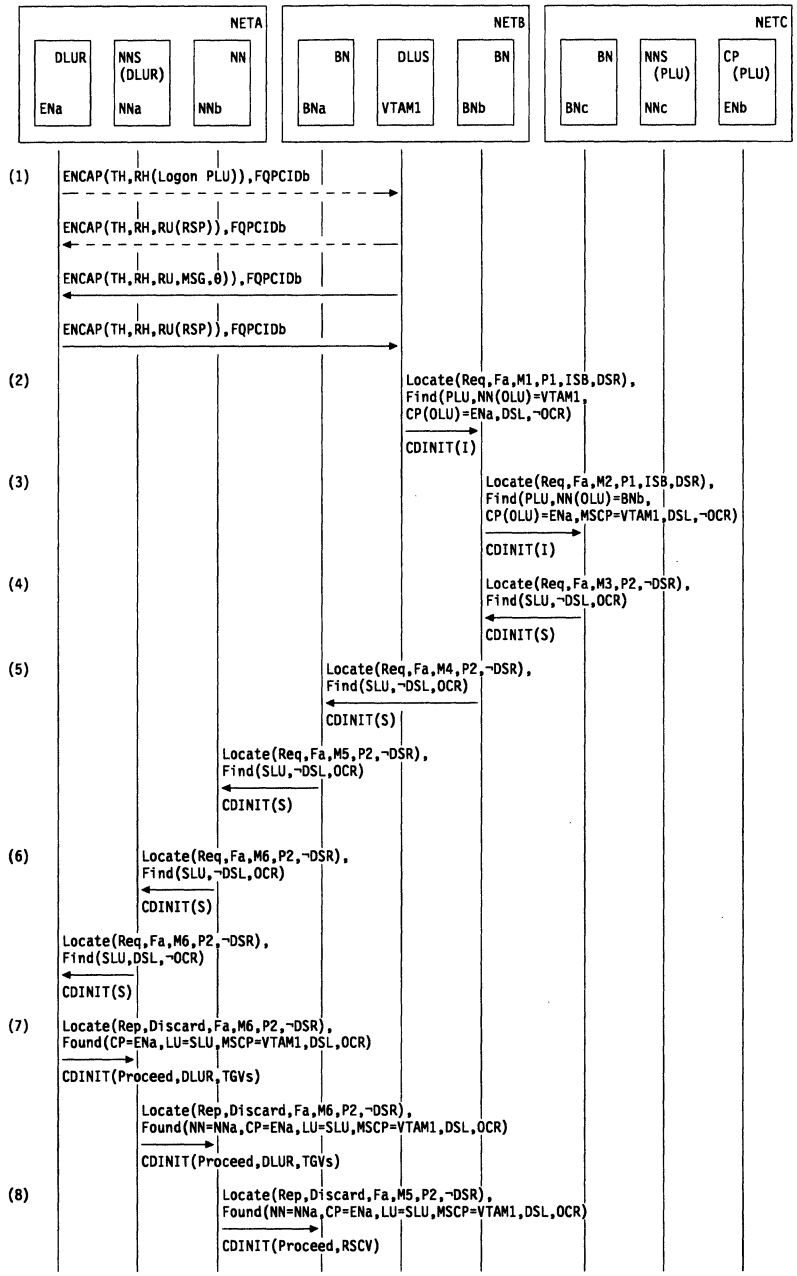


Figure 263 (Part 1 of 2). SLU-Initiated Session with DLUS and DLUR within Different Subnetworks

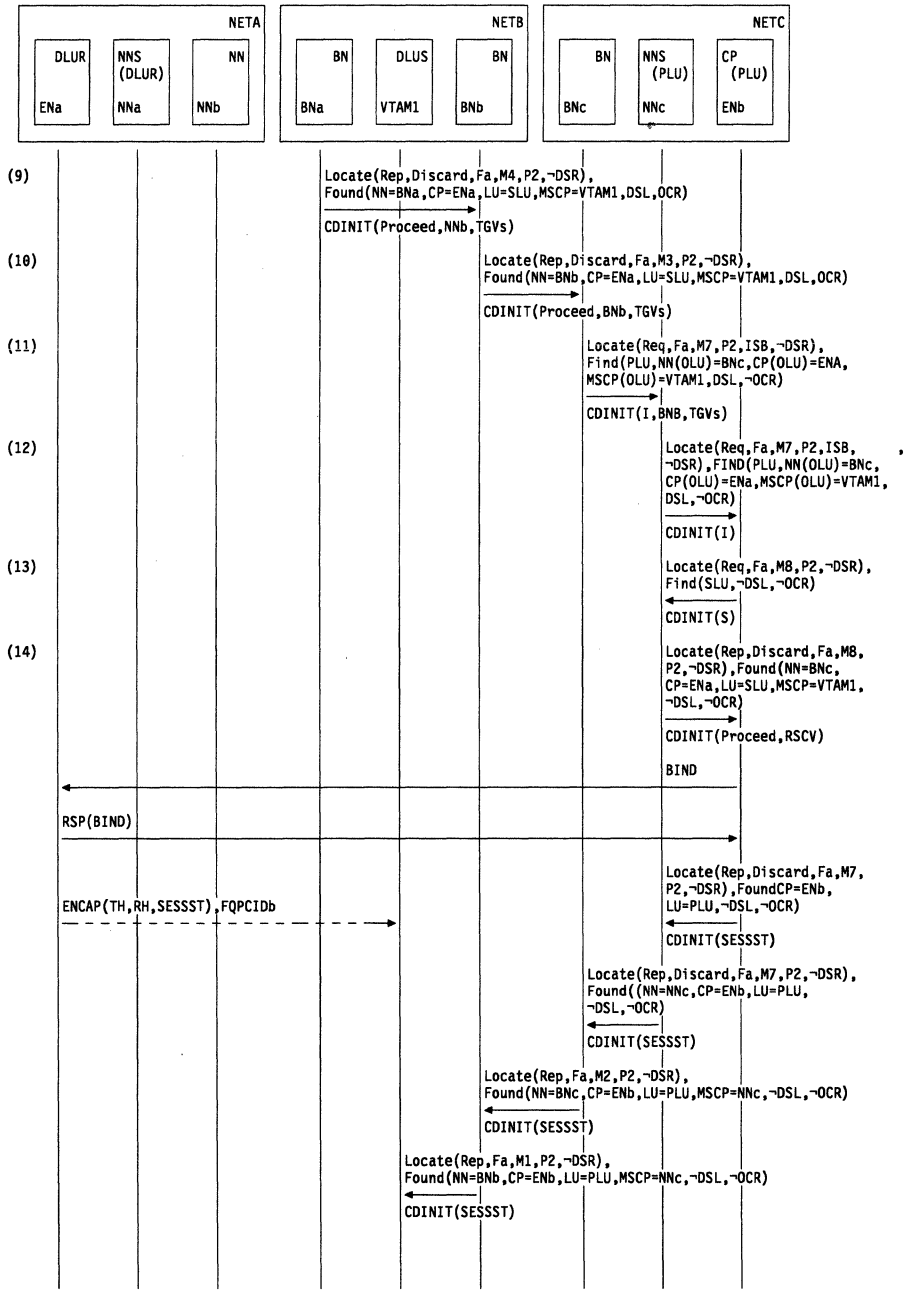


Figure 263 (Part 2 of 2). SLU-Initiated Session with DLUS and DLUR within Different Subnetworks

1. The SLU initiates a Logon, which ENa encapsulates on the CPSVRMGR pipe and sends to VTAM1. VTAM1 sends a response to the Logon request followed by USS MSG0. Ena sends a response to USS MSG0 to VTAM1.
2. Prior to initiating a search to locate the PLU, VTAM1 examines its cache. The PLU is located and is cached as being accessible through BNb. Since ENa is in a different subnet than VTAM1, no endpoint TGVs were reported over the CPSVRMGR pipe. Since the PLU is also within a different APPN subnet, VTAM1 will set the DSR indicator on the Locate request. VTAM1 will also set the DSL indicator on the Locate request since the SLU is a DLUS-served LU.

3. BNb caches the location of the SLU as being on ENa with VTAM1 as the NNS. The OCR indicator will not be set in the cache entry. BNb has the PLU cached as being accessible through BNc.

   BNb modifies the Find resource hierarchy to indicate itself as the NNS(OLU) and adds a CV X'40' with VTAM1 as the MSCP. BNb also adds its endpoint TGs to the Locate search request and forwards the request to BNc.

4. BNc caches the location of the SLU as being on BNb. The OCR indicator is not set within the cache entry.

   BNc has the location of the PLU cached as being within the native subnet. Since both the DSL indicator and the DSR indicator are set on the Locate request, BNc must obtain the SLU TGVs to be included on the Locate search request.

   To obtain the endpoint TGVs, BNc initiates a Locate search to find the SLU. This search will be a PLU-init Search Only. The OCR indicator is set, requesting that the DLUR node respond to the Locate request. The DSR indicator will not be set, since BNc is the node which is performing the extra Locate search.

5. BNb receives the new Locate search from BNc. BNb finds a cache entry for the SLU with the OCR indicator set. The entry indicates that BNa is the NNS(SLU), so BNb forwards the Locate search to BNa.

   BNa also finds a cache entry for the SLU with the OCR indicator set. The cache entry for BNa indicates that the search should be forwarded to NNb.

6. NNb receives the Locate search from BNa. As part of its search logic, NNb will send either a directed Locate search to NNa or will perform a network broadcast which will ultimately reach NNa. Either way, the Locate search will be forwarded to NNa and, ultimately, ENa.

7. The OCR indicator is set, so ENa responds to the Locate. Since the SLU is a DLUS-served LU, ENa sets the DSL indicator on the reply. When building the reply, ENa will include a CV X'40' which includes the DLUS node's CP name.

8. NNb caches the location of the SLU as being on ENa, with NNa as the NNS. NNb then calculates an RSCV for the Bind route between BNa and ENa and returns the RSCV to BNa.

9. BNa caches the location of the SLU as being on ENa, with NNb as the NNS. Since both the DSL and OCR indicators are set on the Locate reply, BNa sets the OCR indicator within the cache entry.

   BNa modifies the Found resource hierarchy to indicate itself as the NNS(DLU). BNa then removes the RSCV from the Locate reply and places NNb's TGVs on the reply. The reply is then returned to BNb.

10. BNb caches the location of the SLU as being on ENa with BNa as the NNS. Both the DSL and OCR indicators are on the Locate reply, BNb sets the OCR indicator on the reply. BNb then modifies the resource hierarchy in the reply to indicate itself as the NNS(DLU). BNb also replaces NNb's TGs with its own TGs and then forwards the reply to BNc.

11. BNc caches the location of the SLU as being on ENa with BNb as the NNS. Since both the DSL and OCR indicators are set on the Locate reply, BNc sets the OCR indicator within the cache entry.

BNc then takes the endpoint TGVs which were returned on the Locate reply which was just received from BNb and places the TGVs on the Locate request which was received from BNb. BNc then modifies the resource hierarchy on the request to indicate itself as the NNS(OLU). The search request is then sent to NNc.

12. NNc forwards the search request to ENb.
13. Because this is a SLU-init search request and an RSCV was not present on the Locate request, ENb initiates a PLU-init search with the SLU as the target.
14. NNc correlates the PLU-init search request with the outstanding SLU-init search request. Using the information which was provided on the original SLU-init search, NNc calculates a RSCV and returns it to ENb on the Locate reply. ENb then BINDs the session between the PLU and the SLU.

    Since NNc creates the Locate reply instead of VTAM1, neither the DSL indicator nor the DSR indicator will be set on the reply. This can be contrasted to the original SLU-init request in which the DSL indicator was set.

Appendix G. Control Point/Control Block (CPCB) Operation Codes

Many processes of the VTAM program are represented by RUPE, NCSPL, DLRPL, CPCB, PPL, NOSPL, MFT, FILTR, or TQE work elements. Each of these work elements contains a prefix called a CPCB at the beginning of the control block. The CPCB prefix contains a field called the CPCB operation code (CPCBOPC), which provides an indication of the type of VTAM process represented by the work element.

The CPCBOPCs are mapped by the ISTCPKCB with reference to the data area. The CPCBOPC field is 4 bytes long and contains a category byte (CPCBCAT), followed by a 3-byte specific operation code field (CPCBFMH).

The contents of the CPCB operation code category are summarized below:

Category (hex)	Meaning
00	Operator command
01	Dump/load/restart request
02	I/O purge request
03	Timer management request
04	Unformatted RU
08	Function management data RU
09	Network control RU
0A	Data flow control RU
0B	Session control RU
0C	Function management data access method RU
0D	Network control access method RU
0E	Data flow control access method RU
0F	Session control access method RU
FC	Function management data access method interprocess signal
FD	Network interprocess signal

Table 76 on page 1128 summarizes the possible values of the CPCB operation code field.

Notes:

1. For each CPCB operation code listed, the character string representation that may appear in VTAM operator messages is given, along with the function of the associated work element.
2. Internal codes that are used only by the product support organization to assist in internal flow diagnosis are not included in this list.

Table 76 (Page 1 of 9). Control Point/Control Block Operation Codes (CPCBOPC)

CPCBOPC	Message Display	Function
00010000	VARY	VARY Command
00010001	VARY ACT	VARY Activate
00010002	VARY INACT	VARY Deactivate
00010004	VARY LOGON	VARY LOGON
00010005	VARY ACT	VARY Activate, LOGON Parameter Specified
00010008	VARY DRDS	VARY DRDS
00010012	VARY INACT	VARY Deactivate Immediate
00010032	VARY INACT	VARY Deactivate Immediate (Internal)
00010040	VARY ANS	VARY ANS
00010041	VARY ACT	VARY Activate, ANS Parameter Specified
00010045	VARY ACT	VARY Activate, ANS and LOGON Parameters Specified
00010080	VARY HANGUP	VARY HANGUP
00010100	VARY PATH	VARY PATH=USE
00010200	VARY PATH	VARY PATH=NOUSE
00010400	VARY INOP	VARY INOP
00010802	FORCE DEAC	Force Deactivate
00011002	FORCE REAC	Force Reactivate
00011802	VARY INACT	VARY Deactivate Giveback
00012000	VARY ACQ	VARY Acquire
00012001	VARY ACQ	VARY Activate, ACQ Parameter Specified
00012002	INACT SON	Deactivate (Session Outage Notification)
00012005	VARY ACQ	VARY Activate, ACQ and LOGON Parameters Specified
00014000	VARY REL	VARY Release
00014010	VARY REL	VARY Release Immediate
00014012	VARY REL	VARY Release Giveback
00018000	VARY DIAL	VARY DIAL
00018004	VARY NOLOGON	VARY NOLOGON
00020000	F EXIT CMD	MODIFY EXIT CMD
00020001	F DUMP	MODIFY DUMP
00020002	F ENCR	MODIFY ENCR
00020003	F TOP	MODIFY TOP
00020004	F CHANGE	MODIFY CHANGE
00020010	F TABLE	MODIFY TABLE
000200A0	F TRACE	MODIFY TRACE, NETCNTR

Table 76 (Page 2 of 9). Control Point/Control Block Operation Codes (CPCBOPC)

CPCBOPC	Message Display	Function
000200C0	F NOTRACE	MODIFY NOTRACE, NETCNTR
00020120	F AGPTRACE	MODIFY Activate generalized PIU trace
00020140	F DGPTRACE	MODIFY Deactivate generalized PIU trace
00020180	F LOAD ADD	MODIFY LOAD ADD
00020240	SETTIME	SETTIME Cancel
00020200	F CDRM	MODIFY CDRM
00020280	F LOAD REP	MODIFY LOAD REPLACE
00020401	F DUMP TRN	Transfer DUMP (NCP)
00020408	F DR MOVE	MODIFY DR MOVE
00020480	F LOAD PRG	MODIFY LOAD PURGE
00020801	PRG NCP DUMP	PURGE DUMP (NCP)
00020808	F DR DELETE	MODIFY DR DELETE
00020820	F ALTRACE	MODIFY Activate Line Trace
00020840	F DLTRACE	MODIFY Deactivate Line Trace
00020880	F LOAD CNC	MODIFY LOAD CANCEL
00021001	F DUMP CSP	MODIFY DUMP (CSP)
00021080	F LOAD	MODIFY LOAD ACTION=SETTIME
00021801	PRG DUMP CSP	PURGE DUMP (CSP)
00022001	F DUMP MOS	MODIFY DUMP (MOSS)
00022080	F LOAD	MODIFY LOAD ACTION=RENAME
00022801	PRG DUMP MOS	PURGE DUMP (MOSS)
00024001	F DUMP DYN	MODIFY Dump (Dynamic)
00024020	F ACT SIT	MODIFY Activate SIT Trace
00024040	F DACT SIT	MODIFY Deactivate SIT Trace
00028001	F DUMP DYN CH	MODIFY Dump (Dynamic-CH)
00028820	F ATGTRACE	MODIFY Activate TG Trace
00028840	F DTGTRACE	MODIFY Deactivate TG Trace
0002C000	F ALSLIST	MODIFY Adjacent Link Station List
0002D000	F DEFAULTS	MODIFY DEFAULTS
0002E001	F DIR	MODIFY DIR DELETE
0002E002	F DIR	MODIFY DIR UPDATE
00030001	SOFT INOP	SOFT INOP
00030002	SSCP TKOVR	SSCP TAKEOVER
00030004	HARD INOP	HARD INOP
00040000	DISP INT	Display Internal Commands
00060001	INT SYNCH	Internal Synchronization Function
00060002	IPL INIT	IPL Initial Request
00060004	IPL TEXT	IPL Text Request

Table 76 (Page 3 of 9). Control Point/Control Block Operation Codes (CPCBOPC)

CPCBOPC	Message Display	Function
00060008	IPL FINAL	IPL Final Request
00060010	DUMP INIT	Dump Initial Request
00060020	DUMP TEXT	Dump Text Request
00060040	DUMP FINAL	Dump Final Request
00060100	HALT CDLNK	Process Cross-Domain Links During HALT
00061001	REQDMP CSP	Request CSP Dump
00062000	RESET LU	Reset LU
00062001	REQDMP MOS	Request MOSS Dump
00063001	INT TRF DH	Internal transfer of dump header
00063002	INT TRF DMS	Internal transfer of dump main storage
00064000	REQLOAD	Request Load
00064001	REQDUMP DY	Request Dynamic Dump Data
00068000	REQDUMP	Request Dump
00080001	DIAL START	Dial Start Request
012B0000	CKPTN	Checkpoint Node Status Function
014B0000	CHKPT	Checkpoint Resource Status Function
01BD0000	CPMSG	Internal WTOR Function
01DD0000	DLR PURGE	Dump/Load/Restart Purge
01EA0000	CPCRYPT	Cryptography Management Function
01ED0000	SELECT VR	Virtual Route Select
02510000	CDRM CLEAR	Clear CDRM-CDRM Session
02520000	CDRM ERP	CDRM ERP Internal Clear
03000000	TIMER REQ	Set Timer Request
04000000	CHAR CODED	Unformatted Request Unit
08010001	CHG TLIMIT	Change Transmission Limit
08010002	CHG NRSPOL	Change Negative Response to Poll Limit
08010003	CHG SESSLM	Change Session Limit
08010004	CHG POLLIM	Change Poll Limit
08010201	CONTACT	Contact
08010202	DISCONTACT	Discontact
08010203	IPL INIT	NC IPL Initial
08010204	IPL TEXT	NC IPL Text
08010205	IPL FINAL	NC IPL Final
08010206	DUMP INIT	Dump Initial
08010207	DUMP TEXT	Dump Text
08010208	DUMP FINAL	Dump Final
08010209	RMPO	Remote Power Off
0801020A	ACTLINK	Activate Link

Table 76 (Page 4 of 9). Control Point/Control Block Operation Codes (CPCBOPC)

CPCBOPC	Message Display	Function
0801020B	DACTLINK	Deactivate Link
0801020E	CONNOUT	Connect Out
0801020F	ABCONN	Abandon Connection
08010211	SCV	Set Control Vector
08010214	ENT SLOWDN	Enter Slowdown
08010215	EXT SLOWDN	Exit Slowdown
08010216	ACTCONNIN	Activate Connect In
08010217	DACTCONNIN	Deactivate Connect In
08010218	ABCONNOUT	Abandon Connect Out
08010219	ANA	Assign Network Address
0801021A	FNA	Free Network Address
0801021B	REQDISCONT	Request Discontact
08010280	CONTACTED	Contacted
08010281	INOP	Inoperative
08010284	REQCONT	Request Contact
08010285	NSLSA	Network Services Lost Subarea
08010302	ACTTRACE	Activate Trace
08010303	DACTTRACE	Deactivate Trace
08010331	DISP STOR	Display Storage
08010334	RECSTOR	Record Storage
08010381	RECMS	Record Maintenance Statistics
08010382	REC TEST	Record Test Data
08010383	REC TRACE	Record Line Trace Data
08010604	NSPE	Network Services Procedure Error
08010681	INIT SELF	Initiate(Self) Format 0
08010683	TERM SELF	Terminate(Self) Format 0
0812C100	CP	CP Capabilities
0812C200	TDU	TDU
0812C300	REGISTER	REGISTER Resource
0812C400	LOCATE	LOCATE Resource
0812C500	CDINIT	CDINIT
0812C900	DELETE	DELETE
0812CA00	FIND	FIND
0812CB00	FOUND	FOUND
0812CC00	NOTIFY	NOTIFY
0812CD00	IOCD	IOCD
083F0233	INIT LOAD	NS Init Load
083F0234	LOAD STAT	NS Load Status

Table 76 (Page 5 of 9). Control Point/Control Block Operation Codes (CPCBOPC)

CPCBOPC	Message Display	Function
083F0814	TR-INQUIRY	Translate Inquiry
083F0816	TR-REPLY	Translate Reply
08410210	RNAA	Request Network Address Assignment
08410220	NFY SESSEND	Notify Sessend
08410237	LOAD REQRD	NS Load Required
08410240	ADDNR	Add Network Resource
08410243	IPL INIT	NS IPL Init
08410244	IPL TEXT	NS IPL Text
08410245	IPL FINAL	NS IPL Final
08410246	IPL ABORT	NS IPL Abort
08410286	RDELETENR	Request Delete Network Resource
08410287	LOST CTLPT	Lost Control Point
08410289	ROUTE-INOP	Network Services Route Inoperative
0841028A	REQACTCDRM	Request ACTCDRM
08410304	REQ MNT ST	Request Maintenance Statistics
08410305	LINKLVL2	Enter Test Mode (LL2)
08410307	REQ RTTEST	Request Route Test
08410311	MS SCV	Maintenance Services Set Control Vector
08410384	RECFMS	Record Formatted Maintenance Statistics
08410385	RECTR	Record Test Results
08410386	ER TESTED	Explicit Route Tested
0841038D	NMVT	Network Manager Vector Transport
08810387	REQ ECHO	Request Echo Test
08810389	ECHO TEST	Echo Test
08810601	CINIT	Control Initiate
08810602	CTERM	Control Terminate
08810620	NOTIFY	Notify
08810629	CLEANUP	Cleanup
08810680	INIT OTHER	Initiate(Other)
08810681	INIT SELF	Initiate(Self) Format 1
08810682	TERM OTHER	Terminate(Other)
08810683	TERM SELF	Terminate(Self)
08810685	BIND FAIL	Bind Failure
08810686	SESS START	Session Started
08810687	UBIND FAIL	Unbind Failure
08810688	SESS ENDED	Session Ended
08810810	FORWARD	Forward Request
08810812	DELIVER	Deliver Request

Table 76 (Page 6 of 9). Control Point/Control Block Operation Codes (CPCBOPC)

CPCBOPC	Message Display	Function
08810814	CNM REQ	CNM Request
08812601	BF CNTL INIT	BF Control Initiate
08812629	BF CLEANUP	BF Cleanup
08812681	BF INIT	BF Initiate
08812683	BF TERM	BF Terminate
08812686	BF SESS STRT	BF Session Start
08812688	BF SESS END	BF Session End
0881268C	BF SESS INFO	BF Session Information Request
08818620	CD NOTIFY	Cross-Domain Notify
08818627	CD DSEARCH	Cross-Domain Direct Search List
08818640	CDINIT OTH	Cross-Domain Initiate (Other)
08818641	CDINIT	Cross-Domain Initiate
08818643	CDTERM	Cross-Domain Terminate
08818645	CDSSF	Cross-Domain Session Setup Failure
08818646	CDSESSST	Cross-Domain Session Started
08818647	CDSTF	Cross-Domain Session Takedown Failure
08818648	CDSESSEND	Cross-Domain Session Ended
08818649	CDTAKEDOWN	Cross-Domain Takedown
0881864A	CDTD COMP	Cross-Domain Takedown Complete
0881864B	CDCINIT	Cross-Domain Control Initiate
09050000	NCLSA	Network Control Lost Subarea
09060000	ER INOP	Explicit Route Inoperative
09060000	ANSS	Auto Network Shutdown Started
09070000	ANSC	Auto Network Shutdown Complete
09080000	LOST PATH	Lost Path
09090000	ER TEST	Explicit Route Test
090A0000	ER TST RPY	Explicit Route Test Reply
090B0000	ER ACT	Explicit Route Activate
090C0000	ER ACT RPY	Explicit Route Activate Reply
090D0000	ACTVR	Activate Virtual Route
090E0000	DACTVR	Deactivate Virtual Route
090F0000	ER OP	Explicit Route Operative
09510000	SW TO NCP	Switch Line to NCP Mode
09520000	SW TO EP	Switch Line to EP Mode
0A040000	LUSTAT	LU Status
0A050000	RTR	Ready to Receive
0A700000	BIS	Bracket Initiation Stopped
0A710000	SBI	Stop Bracket Initiation

Table 76 (Page 7 of 9). Control Point/Control Block Operation Codes (CPCBOPC)

CPCBOPC	Message Display	Function
0A800000	QEC	Quiesce at End of Chain
0A810000	QC	Quiesce Complete
0A820000	RELQ	Release Quiesce
0A830000	CANCEL	Cancel
0A840000	CHASE	Chase
0AC00000	SHUTDOWN	Shutdown
0AC10000	SHUTC	Shutdown Complete
0AC20000	RSHUTD	Request Shutdown
0AC80000	BID	Bid
0AC90000	SIGNAL	Signal
0B0D0000	ACTLU	Activate LU
0B0E0000	DACTLU	Deactivate LU
0B110000	ACTPU	Activate PU
0B120000	DACTPU	Deactivate PU
0B140000	ACTCDRM	Activate CDRM
0B150000	DACTCDRM	Deactivate CDRM
0B310000	BIND	Bind Session
0B320000	UNBIND	Unbind Session
0BA00000	SDT	Start Data Traffic
0BA10000	CLEAR	Clear Session
0BA20000	STSN	Set and Test Sequence Numbers
0BA30000	RQR	Request Recovery
0BC00000	CRV	Cryptography Verify
0C0102A0	AM ALLORSC	Allocate Resource
0C0102A1	AM FREERSC	Free Resource
0C0102A2	AM SETRT	Set Routable State
0C0102A3	AM RESETRT	Reset Routable State
0C0102A5	SDDLU EXIT AMRU	Secondary LU exit AMRU
0C0102A4	AM SC EXIT	Switched connection exit
0C010480	AM RCRD MS DATA	Record Measurement Data
0C410201	AM CONFIG SVCS	Config SVCS
0C410206	AM GAINED GWN	Gained GWN
0C410207	AM LOST GWN	Lost GWN
0C410208	AM DACT TRF	Deactivate Transforms
0C410210	AM RNAA	Request Network Address Assignment
0C410212	AM CONN	Connect
0C410213	AM DISC	Disconnect
0C410214	AM INIT PU	Initiate PU

Table 76 (Page 8 of 9). Control Point/Control Block Operation Codes (CPCBOPC)

CPCBOPC	Message Display	Function
0C410266	XID	AM Exchange ID
0C4102BD	AM ADDLINK	Add Link
0C4102BE	AM ADDLSTA	Add Link Station
0C4102BF	DELETENR	Delete Network Resource
0C4102CD	AM REQDUMP	Request Dump
0C4102CE	AM CONDLOD	Request Conditional Load
0C4102CF	AM UNCDLOD	Request Unconditional Load
0C410601	AM OPNACB	Open ACB
0C410602	AM CLSACB	Close ACB
0C800700	AM VCNS LREQ	VCNS Logon Request
0C800701	AM VCNS LRSP	VCNS Logon Response
0C810619	AM ARCA	Address Request Complete
0C810620	AM RESUME	Resume
0C810629	AM CLEANUP	Cleanup
0C810643	Generic TERM	Termination Placeholder
0C810680	AM REALLOC	Reallocate
0C810681	GEN SESS INIT	Generic Session Initiation
0C810801	AM NOTIFY	Notify
0C810A00	API SETLST	SETLOGON(START)
0C810A01	API SETLSP	SETLOGON(STOP)
0C810A02	API SETLQS	SETLOGON(QUIESCE)
0C810A03	API SETPER	SETLOGON(PERSIST)
0C810A04	API SETNPE	SETLOGON(NPERSIST)
0C810A10	API SIMLOG	SIMLOGON
0C810A20	API OPNACQ	OPNDST(ACQUIRE)
0C810A21	API OPNACC	OPNDST(ACCEPT)
0C810A22	API OPNRES	OPNDST(RESTORE)
0C810A30	API INQLOG	INQUIRE(LOGONMSG)
0C810A31	API INQDVC	INQUIRE(DEVCHAR)
0C810A32	API INQCNT	INQUIRE(COUNTS)
0C810A33	API INQTOP	INQUIRE(TOPLOGON)
0C810A34	API INQCID	INQUIRE(CIDXLATE)
0C810A35	API INQTRM	INQUIRE(TERMS)
0C810A36	API INQAPS	INQUIRE(APPSTAT)
0C810A37	API INQSPM	INQUIRE(SESSPARMS)
0C810A38	API INQSKY	INQUIRE(SESSKEY)
0C810A39	API INQDSP	INQUIRE(DISPLAY)
0C810A3A	API INQPER	INQUIRE(PERSESS)

Table 76 (Page 9 of 9). Control Point/Control Block Operation Codes (CPCBOPC)

CPCBOPC	Message Display	Function
0C810A3B	API INQNQN	INQUIRE(NQN)
0C810A40	API INTERP	INTRPRET
0C810A50	API CLSPAS	CLSDST(PASS)
0C810A51	API CLSRLS	CLSDST(RELEASE)
0C810A60	API SESONC	SESSIONC
0C810A70	API SNDCMD	SENDCMD
0C810A75	API SEND	SEND
0C810A80	API RCVCMD	RCVCMD
0C810A85	API RECEIVE	RECEIVE
0C810A90	API REQSES	REQSESS
0C810AA0	API OPNSEC	OPNSEC
0C810AB0	API TRMSES	TERMSESS
0C810AC0	API RESETSR	RESETSR
0D010000	AM VR INOP	Virtual Route Inoperative
0D0B0000	AM REQ ERA	Request Explicit Route Activate
0D0E0000	AM REQ VRD	Request Virtual Route Deactivate
0DFF0000	AM VR STAT	Virtual Route Status
0E010000	AM PCE	Purge Chain Element
0F010000	AM NFY SLT	Notify (Schedule LOSTERM Exit)
0F020000	AM SSA	Set Session Address
0F030000	AM SSADISC	Set Session Address and Disconnect
0F040000	AM OSA	Override Session Address
0F050000	AM PWQ	Purge Wait Queue
0F060000	AM FLUSH	Flush Virtual Route
0F310000	AM GBIND	Generic BIND
0F320000	AM GUNBIND	Generic UNBIND
FF000000	VECTOR	VECTOR FMD request units

Appendix H. Storage and Control Block ID Codes

This appendix lists the control block ID codes for VTAM and VSCS control blocks. It also includes storage ID codes for VSCS.

This appendix contains the following sections:

- "VTAM Control Block ID Codes"
- "VSCS Storage and Control Block ID Codes (VM)" on page 1139.

VTAM Control Block ID Codes

You can identify certain VTAM control block types in a storage dump by examining an identification code in the first byte of the control block (offset 0). The control block identification codes are shown in the following table. In case of duplicate codes, use other means (such as the operating environment or the control block's context) to determine the type of control block.

Note: Internal codes that are used only by the product support organization to assist in internal flow diagnosis are not included in Table 77.

Table 77 (Page 1 of 2). Control Block ID Codes

ID	Control Block
00	SONCB
00	RPL
01	RPH
03	FMCB
05	VRBLK
06	ICNCB
07	LDNCB
09	HALCB
0A	BSCLB
0B	VLNCB
0C	PCLCB
0D	PRWCB
0E	TRGCB
0F	ACDEB
10	UECB
11	DYPAB
13	TRAC
14	ERT
19	PDVT
1A	CHAIN
1B	TGCB

Table 77 (Page 1 of 2). Control Block ID Codes

ID	Control Block
1E	RCE
24	OCW
26	PICB
29	LMPCB
2B	RAQ
2C	PAQ
2D	SAT
2E	AHNCB
41	PUSCB
43	PLSCB
45	POIA
46	POCB
47	POMCB
48	PORCB
49	POWE
4C	ERCT
4D	TGE
4E	VRWSE
50	DLRPL
52	LUCB
54	RUPE

Table 77 (Page 2 of 2). Control Block ID Codes

ID	Control Block
54	TUNB
58	TQE
5A	PRQAB
5B	PRBLK
5C	CPCB
60	NCSPL
61	PST
62	AMU
62	NSSCB
62	NSICB
63	SMP
64	OCB
65	NACP
66	CAB
67	CANT
68	RSQE
69	RDTPL
6A	NOSPL
6B	SSIB
6B	CAR
6C	IOSIB
6D	ASRIT
6D	RANT
6E	GWIT
6F	RARB
75	WRE
77	ADJSR
78	ADJSS
7B	RIB
81	EXLST
96	SIBXN
97	SIBIX
98	SIB
99	TSCB
9A	TSPL
9B	LSCB
9C	CNCB
A0	ACB

Table 77 (Page 2 of 2). Control Block ID Codes

ID	Control Block
A2	RNCA
A3	ALCA
BD	UDT
BE	INT1
BF	COS
C0	LOGMD
C1	RPL6X
CA	GRPCB
D0	NIB
EB	AUTOE
FE	XCNCB
FF	OCA
ART	ART
LSP	LSPL
PRT	PRTCB
SAP	SAPCB

VSCS Storage and Control Block ID Codes (VM)

You can identify certain VSCS control block types in a storage dump by examining either an identification code in the two bytes of the control block for the control block ID or the beginning of storage for the seven byte save area identifier.

Storage Identifiers

ID	Storage Type
SS	Static storage
SD	Dynamic storage
ZZ	Dummy storage prefix

Control Block Identifiers

ID	Control Block
CG	DTICGB
CI	DTICIA
CL	DTICLB
CM	DTICMD
DTICXWB	DTISWBs for Communication Services
DTIPSWB	DTISWBs for Presentation Services
DTISPSB	DTISWBs for Presentation Services Abends
DTISUTB	DTISWBs for Utility Services Abends
DTISVSB	DTISWBs for VTAM Services Abends
DTITXWB	DTISWBs for Timer Services
DTIUSWB	DTISWBs for Utility Services
DTIVSWB	DTISWBs for VTAM Services
DTIVXWB	DTISWBs for VTAM Exit Services
IN	DTIWEB internal
PB	DTIPDB
PD	DTIDSB for Presentation Services
PG	DTIPGB
PL	DTIPLB
SA	DTISAB
SC	DTISCB
SM	DTISDB
TA	DTITAB
TD	DTIGTD
TR	DTITHDR
UG	DTIUGB
VA	DTIVAB
VD	DTIDSB for VTAM Services
VE	DTIVEIB
VG	DTIVGB
VL	DTIVLB
4I	DTIWEB IREAD internal read buffer
1R	DTIWEB IRECEIVE one-way reply
2R	DTIWEB IRECEIVE one-way send
1S	DTIWEB ISEND one-way reply
2S	DTIWEB ISEND one-way send
3R	DTIWEB IRECEIVE two-way reply
4R	DTIWEB IRECEIVE two-way send
3S	DTIWEB ISEND two-way reply
4S	DTIWEB ISEND two-way send

Appendix I. Path Information Unit (PIU) Discard Reason Codes

This appendix includes the following sections:

The VTAM program path information unit (PIU) discard reason codes can be found in session monitor trace panels and in the VTAM internal trace (VIT) DSCD entry. When VTAM's transmission subsystem component (TSC) cannot forward a PIU toward its final destination, it generates a discard trace entry, consisting of a discard reason code and the discarded PIU. The TSC sends the discard trace entry to the NetView program, or to the VIT, or to both.

Note: The NetView program must be active to VTAM in order to accept discard trace entries. The VIT must also be active.

To reference all discard entries sent to the NetView program, issue the following NetView command:

SESS *DISCARD

The NetView program puts discarded PIUs in this file and, when possible, also puts them in the associated session trace file.

Because PIUs are often discarded without causing problems in the network, most discarded PIUs are not indicative of error conditions and do not need to be investigated. A discarded PIU should be investigated only if an error symptom, such as a "hang" condition, occurs.

See Table 80 on page 1165 to determine what book describes session monitor trace panels.

See "DSCD Entry for Discarded PIU Buffer (Part 1)" on page 662 for a full description of the VIT DSCD entry.

PIU Discard Reason Code Descriptions

Table 78 lists the VTAM discard reason codes for each BSC, HS, and TSC module that discards PIUs. BSC, HS, and TSC modules, with their associated reason codes, are listed in alphanumeric order. For each reason code, a specific reason is given for the discard, followed by a probable cause. Some probable causes are described in more detail in Table 79 on page 1154.

Table 78 (Page 1 of 12). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause (See Table 79)
ISTBSCBC (receive BFCLEANUP)	0001	The link station is not connected.	The link has become inoperative.
ISTBSCBP (boundary function pacing utility)	0001	The link station is not connected.	The link has become inoperative.
ISTBSCBQ (receive BIND request)	0001	The link station is not connected.	The link has become inoperative.
	0002	Storage could not be obtained for a BSB.	Internal VTAM logic error.
ISTBSCBR (receive negative BIND/GBIND response)	0001	The BSB could not be found from the network addresses in the transmission header.	The session control block has already been freed.
	0002	The BSB is not in a valid state.	Internal VTAM logic error.
ISTBSCIA (receive inoperative access method RU)	0001	The BPB for the PU element address in the inoperative AMRU could not be found.	Session control blocks have already been freed.
MVS ISTBSCMQ (receive MPTN Connect)	0001	The link station is not connected.	The link has become inoperative.
ISTBSCRT (boundary function session services router)	0001	An unrecognized response TSCB was received and freed.	Internal VTAM logic error.
	0002	An unrecognized request AMRU was received and freed.	Internal VTAM logic error.
	0003	This work element is being discarded because the NCB is being freed.	The link is becoming inoperative.
ISTBSCUA (receive UNBIND access method RU)	0001	The BSB for the session control block specified in the UNBIND cleanup AMRU could not be found.	The session control block has already been freed.
ISTBSCUB (receive UNBIND request)	0001	The BSB could not be found when the UNBIND came from the PLU.	The session control block has already been freed.
	0002	The link station is not connected.	The link has become inoperative.
	0003	The BSB is in the PLU initial state and the BSB has not been added to the subarea HASH table.	The session is being terminated by an UNBIND RU to the route extension PLU.

Table 78 (Page 2 of 12). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause (See Table 79)
ISTBSCUR (receive UNBIND/GUNBIND response)	0001	The BSB could not be found from the network addresses in the transmission header.	The session control block has already been freed.
ISTBSCVR (boundary function session services VTAM recovery routine)	0001	An abend occurred while processing a response.	Internal VTAM logic error.
	0002	An abend occurred while processing an RU.	Internal VTAM logic error.
	0003	An abend occurred while processing an AMRU.	Internal VTAM logic error.
	0004	A link inoperative condition has occurred. The BFSS PAB is being freed.	Internal VTAM logic error.
ISTHSCEP	0001	The session is being deactivated. Unsent PIUs must be discarded.	Internal VTAM logic error.
	0002	The session is being deactivated. PIUs on the pacing queue must be discarded.	Internal VTAM logic error.
	0003	The session is being deactivated. PIUs on the HS PAB must be discarded.	Internal VTAM logic error.
	0004	The session is being deactivated. Expedited flow PIUs must be discarded.	Internal VTAM logic error.
	0005	The session is being deactivated. PIUs on the response queue are discarded.	Internal VTAM logic error
ISTHSCGI	0001	Either insufficient resources or LRM encounters an error while processing.	Internal VTAM logic error.
ISTHSCRT	0001	A PIU that is not a BIND or UNBIND was received from another VTAM component on the expedited work queue.	Internal VTAM logic error.
	0002	An SC PIU that is not a BIND or UNBIND was received from the network.	Internal VTAM logic error.
ISTHSCRU	0001	Insufficient resources.	Internal VTAM logic error.
ISTHSCRV	0001	Format error on received request PIU.	Internal VTAM logic error.
	0002	Format error on received response PIU.	Internal VTAM logic error.
	0003	PIU received in incorrect state.	Internal VTAM logic error.
	0004	Insufficient resources.	Internal VTAM logic error.
ISTHSCSN	0001	Chaining state error.	Internal VTAM logic error.
ISTHSCSR	0001	Chaining state error.	Internal VTAM logic error.
ISTHSCTP	0001	Insufficient resources.	Internal VTAM logic error.
	0002	Pacing response received when pacing is not supported.	Internal VTAM logic error.
	0003	Insufficient resources for IPM acknowledgement.	Internal VTAM logic error.
ISTHSCTR	0001	PIU received caused transmission control protocol violation.	Internal VTAM logic error.

Table 78 (Page 3 of 12). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause (See Table 79)
ISTLSC6I	0001	TG segment out of sequence; connection will be terminated.	Network error.
	0002	Discard the request or response, due to connection termination.	ISTTSCPD (PIU Deblocker) determined that PIU is not valid.
ISTLSC6Z	0001	Previous PIU was lost.	Network error.
	0003	Segmenting length error.	Network error.
	0004	Accumulated PIU was discarded due to error.	Network error.
ISTTSCBP (boundary pacing utility)	0001	An unexpected pacing response has been received.	The half-session partner is pacing the session after "No Pacing" specified on BIND.
ISTTSCBR	0001	BDU command code is not valid.	
	0002	No response to the command in progress, or the command in progress has been sent again.	
	0003	Control command is not valid.	
	0004	Not an error response, but SET MODE failed in processing completion of select BHSET command.	
	0005	CLEAR command in progress failed.	
ISTTSCCB	0001	Boundary half-session control block could not be found.	Session has been deactivated.
ISTTSCCN	0001	Device was reset.	
ISTTSCCR (connection point manager receive)	0001	The request TSCB is a PRI.DT.SEND or a SEC.DT.RCV in pending reset state.	Timing on LU-LU session traffic.
	0002	Inbound FMD request when SSCP-LU session is not active.	Timing on SSCP-LU session traffic.
	0003	HOT IO Condition	
	0004	An IPM was received that was of invalid length.	
ISTTSCDR (data receive)	0001	Data was received but the session initiation macro (OPNDST or OPNSEC) is incomplete.	Session is terminated during session initiation.
ISTTSCFC	0001	A partial PIU on the FMCB discarded because FMCB is being cleaned up.	Application program or session abended.
	0002	A queued session control RU discarded because FMCB is being cleaned up.	Application program or session abended.
ISTTSCGR (general response routine)	0001	Response could not be generated because the PIU was FID0, was not first in the segment, was already a response, or a response was not allowed by RH.	Internal VTAM logic error.

Table 78 (Page 4 of 12). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause (See Table 79)
ISTTSCHT	0001	PIU is discarded from the PUT pending traffic queue.	
	0002	PIU is discarded from the NCB pending traffic queue.	
	0003	PIU is discarded from the NCB work queue.	
	0004	PIU is discarded from the NCB inbound router PAB.	
ISTTSCIP (host IRN processor)	0001	The route is not available to the destination.	Post-failure traffic purge.
	0002	A response cannot be generated.	Post-failure traffic purge.
ISTTSCIR	0001	A partial PIU on the normal flow wait queue is discarded because PLUSC AMRU is received.	A partial PIU is received when an entire PIU is expected.
ISTTSCIS (inbound session control function interpreter)	0001	The response is discarded when PRI.SESS.SEND is not in a pending active state.	Timing on LU-LU session traffic.
	0002	The response is discarded when PRI.DT.SEND is not in a pending active state.	Timing on LU-LU session traffic.
	0003	The response is discarded when PRI.DT.SEND is not in a pending reset state.	Timing on LU-LU session traffic.
	0004	The response is discarded when RQR.SEND is not in a pending state.	Timing on LU-LU traffic.
	0005	The response is discarded when PRI.SESS.SEND or SEC.SESS.RCV is not in an active state.	Timing on LU-LU session traffic.
	0006	The session control response is not valid.	Logical unit error.
	0007	Last in segment not received.	Logical unit error.
	0008	Discard the request or response.	An UNBIND has freed the PIUs on the half-session send queue.
	0009	Discard the request or response.	An UNBIND has freed the PIUs on the half-session pacing queue.
	0010	PIUs on the response queue are discarded because CLEAR request received.	Element on the response queue for CP-CP session.
	0011	A queued session control RU is discarded because CLEAR request received.	Session control RU purged.
ISTTSCIU (host IRN utility processor)	0001	The link is not contacted and active.	Post-failure traffic purge.
	0002	No IRN extension exists on the NCB.	Post-failure traffic purge.

Table 78 (Page 5 of 12). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause (See Table 79)
ISTTSCJS	0001	PIUs discarded from BCT pending traffic queue for logical unit session control request.	
	0002	PIUs discarded from BSCLB waiting printer end queue for logical unit session control request.	
	0003	PIUs discarded from BCT pending traffic queue for PU session control request.	
	0004	PIUs discarded from BSCLB waiting printer end queue for PU session control request.	
ISTTSCJT	0001	PIU is discarded from the BCT pending traffic queue.	
	0002	PIU is discarded from the BSCLB waiting printer end queue.	
	0003	PIU is discarded from the PCPAB work queue.	
	0004	PIU is discarded from the BSCLB inbound router PAB.	
ISTTSCKF	0001	Data cannot be encrypted or decrypted on LU 6.2 session.	Internal VTAM logic error.
	0002	Data length error during encryption on LU 6.2 session.	Received RU data is not a multiple of 8 bytes.
	0003	Pad count error during encryption on LU 6.2 session.	Pad count on received data is less than 1 or greater than 7.
	0004	PIU discarded because of storage allocation failure.	Storage allocation failure.
ISTTSCKS (session control function interpreter for cryptography)	0001	The request or response was discarded because ENCRYPTN=31 was specified as a start option but the cryptographic facility is not running in a 31-bit storage environment.	
ISTTSCLS (local 3270 write scheduler)	0001	The link is not yet active.	Post-failure traffic purge.
	0002	The request was purged because of the session control request that followed.	Timing on LU-LU session traffic.
VM ISTTSCMB (migration BIND, UNBIND, ACTCDRM, DACTCDRM processor)	0001	This is not a migration route.	Internal VTAM logic error or route has been redefined.
	0002	The ICNCB was not found.	Post-failure traffic purge.
	0003	The virtual route (VR) number and the transmission priority (TP) number are not 0, as they must be for migration.	VR and TP numbers. must be 0 for migration.
ISTTSCNC (network control function interpreter)	0001	The initialization complete request is the only inbound request allowed from network control.	Error in network.

Table 78 (Page 6 of 12). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause (See Table 79)
ISTTSCNS (no session PAB processor)	0001	No processing is required for this FID0 PIU.	Timing consideration with BSC 3270.
ISTTSCOS (outbound session control function interpreter)	0001	Discard the request or response.	An UNBIND has freed the PIUs on the segment assembler queue.
	0002	Discard the request or response.	An UNBIND has freed the PIUs on the half-session send queue.
	0003	Discard the request or response.	An UNBIND has freed the PIUs on the half-session pacing queue
	0004	Discard the request or response.	An UNBIND has freed the PIUs on the normal flow data queue.
ISTTSCPA (PLU activation router)	0001	An unrecognized response RU was received.	Internal VTAM logic error.
ISTTSCPH (purge hot I/O processor)	0001	The RU size is exceeded, and the PIU is not for session control.	Logical unit problem — logical unit sent a PIU larger than the BIND-specified maximum.
ISTTSCPI (path control inbound processor)	0001	The RU size is exceeded, and the PIU is not for session control.	Logical unit problem — logical unit sent a PIU larger than the BIND-specified maximum.
ISTTSCPL (purge LUCB PABs)	0001	The RU is not for session activation or session deactivation.	An application has abended, and VTAM is freeing work for the application.
ISTTSCPR (path control router)	0001	A response cannot be generated for an IRN PIU.	Post-failure traffic purge.
ISTTSCQA (session activation request processor)	0001	A GBIND has been received for a link or station that is becoming inoperative.	A GBIND has arrived for an LU that is subordinate to a PU or a link that is currently becoming inoperative.
ISTTSCQD (session deactivation request processor)	0001	A contention ACTCDRM was discarded because a DACTCDRM arrived from the same subarea.	Operation — operator deactivated CDRM from one host after activation.
	0002	A contention DACTCDRM(SON) was discarded because an ACTCDRM for the same subarea had been previously received.	The route over which the pending CDRM session is being activated has become inoperable.
ISTTSCQF	0001	Abend recovery PIU discard.	
	0002	A queued session control RU is discarded.	FMCB in recovery from ABEND
ISTTSCQP (session deactivation processor)	0001	An ACTCDRM was discarded due to ACTCDRM contention session restart.	Operation — CDRM activated from both hosts simultaneously.

Table 78 (Page 7 of 12). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause (See Table 79)
ISTTSCRA (abend recovery)	0001	This is not a "first-in-segment" PIU.	Post-failure traffic purge after application program abend.
ISTTSCRI (virtual route path control inbound)	0001	The origin subarea is not valid.	Adjacent NCP error.
	0002	The route is not usable.	Post-failure traffic purge — ER not operative.
	0003	The sequence number is not valid.	Lost data or logical unit error.
ISTTSCRO (virtual route path control outbound)	0001	The route is not active.	Post-failure traffic purge.
	0002	The activation number does not match the VR activation number.	Post-failure traffic purge.
ISTTSCSA (segment assembler)	0001	The beginning of the PIU segment was lost.	Network error.
	0002	Two "First-in-Segment" PIUs were received. Discard the partial PIU segment and retransmit the PIU.	Check logical unit.
	0003	Hot I/O. Discard all PIU segments for this session.	Check logical unit.
	0004	Response PIU received that is not valid.	Check logical unit.
ISTTSCSC (session serialization control)	0001	The routing block code does not exist.	A request has arrived for a resource attached to a link that is not currently active.

Table 78 (Page 8 of 12). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause (See Table 79)
ISTTSCSR (LUCB session router)	0001	Incorrect virtual route ID.	PIU originator error.
	0002	A complete DACTCDRM request caused by contention resolution, session override, or other session outage notification.	Network operator error or transmission group failure.
	0003	Session deactivation response and state is not pending reset.	Out of synchronization. Possibly misrouted.
	0004	Negative session activation response and flow direction does not match the direction indicator.	PIU originator error.
	0005	Negative session activation response and sequence number that is not valid.	Out of synchronization or lost traffic.
	0006	Negative session activation response and state is not pending active.	Out of synchronization — multiple session activation and deactivation running concurrently.
	0007	The session control response PIU is not flagged as expedited flow.	Protocol violation by the response originator.
	0008	The session deactivation response is not valid for the session type.	The application specified a user RH that is not valid — not indicating definite response.
	0009	An unsolicited session deactivation response was received and discarded.	Session deactivation responses are being forwarded when a session deactivation request has not been sent.
	0010	Protocol violation — negative response indicator not set in the RH of a negative BIND response.	Protocol violation by the response originator.
	0011	An UNBIND request discarded because session is not LU-LU.	Internal VTAM logic error.
MVS ISTTSCTD (TCP/IP Inbound Boundary Function)	0001	Non-session control RU received and the session is not active.	Trailing session traffic exists on a session that is no longer active.
	0002	The session is reset and an UNBIND request was received.	Trailing session traffic exists on a session that is no longer active.
	0003	A BIND response was received and the session is not pending active.	The session has been deactivated.
	0004	Invalid Session Control request	VTAM has received an invalid session control RU indicating an error in the network.

Table 78 (Page 9 of 12). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause (See Table 79)
MVS ISTTSCTO (TCP/IP Outbound Boundary Function)	0001	Bind response was received and the session is not pending active.	The session has been deactivated.
	0002	Unbind response was received and the session is not pending reset.	Trailing session traffic exists on a session that is no longer active.
ISTTSCTR (boundary function transmission control outbound)	0001	(PLU,SLU).BF.SESS.RCV is not pending active for the BIND response.	A stray BIND response for a previous session arrived before the expected BIND response, or the session partner changed his PCID between the BIND request and the BIND response.
	0002	(PLU,SLU).BF.CPMGR.RCV detected a middle or last segment PIU for a BIU sent to a route extension that does not accept segments.	The BIND request or the BIND response received from the route extension during session setup had the whole-bius required indicator set. The originator of the discarded PIU violated the protocol.

Table 78 (Page 10 of 12). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause (See Table 79)
ISTTSCTS (boundary function transmission control inbound)	0001	(SSCP,PU).BF.SESS.RCV is not pending active for the ACTPU response. VTAM was not waiting for an ACTPU response.	Out of synchronization on SSCP-PU session.
	0002	(SSCP,PU).BF.SESS.RCV is not pending reset for the DACTPU response. VTAM was not waiting for a DACTPU response.	Out of synchronization on SSCP-PU session.
	0003	(SSCP,SLU).BF.SESS.RCV is not pending active for the ACTLU response. VTAM was not waiting for an ACTLU response.	Out of synchronization on SSCP-LU session.
	0004	(SSCP,LU).BF.SESS.RCV is not pending reset for the DACTLU response. VTAM was not waiting for a DACTLU response.	Out of synchronization on SSCP-LU session.
	0005	(PLU,SLU).BF.SESS.RCV is not pending active for the BIND response. VTAM was not waiting for a BIND response.	Timing between SSCP sessions and LU-LU session — SSCP session reset during logical unit session activation.
	0006	(PLU,SLU).BF.SESS.RCV is not pending active for the UNBIND response. VTAM was not waiting for an UNBIND response.	Timing between SSCP sessions and LU-LU session — SSCP session reset during logical unit session deactivation.
	0007	There is an error in control vector hex 60.	A stray BIND response for a previous session arrived before the expected BIND response, or the session partner changed his PCID between the BIND request and the BIND response.
	0008	The indicator to accept segments into the subarea is off.	The half-session partner is incapable of receiving PIU segments, but the other partner is sending one.
	0009	VTAM received an unexpected request.	Error in network. VTAM has received a session control request that is not valid.
	0010	(PLU,SLU).BF.SESS.RCV is reset upon receipt of an UNBIND.	Timing on simultaneous session deactivation between VTAM and the session partner.
MVS ISTTSCTT (TCP/IP Write Scheduler)	0001	The link station is not connected.	The link has become inoperative.
ISTTSCWS (communication controller/cluster controller write scheduler)	0001	The channel link has not been contacted.	Post-failure traffic purge.
ISTTSCYA	0001	A GBIND is being discarded.	Boundary PU has INOP'd.

Table 78 (Page 11 of 12). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause (See Table 79)
ISTTSCYP	0002	A trailing UNBIND response was received and discarded.	Session resources have already been freed and reused for a new session.
ISTTSCXS (channel-to-channel attachment write scheduler)	0001	The channel link is not in contacted or active state.	Post-failure traffic purge.
ISTTSC3R (3270 virtual PU/LU transmission control outbound)	0001	DT.RCV is not active.	Timing on LU-LU session traffic.
	0002	The session is being reset.	Timing on LU-LU session traffic.
ISTTSC3S (3270 virtual PU/LU transmission control inbound)	0001	The session was not active, and the PIU received did not contain a session control RU.	Timing on LU-LU session traffic.
	0002	(SSCP,PU).SEC.SESS.RCV is not pending active for the ACTPU response. VTAM was not waiting for an ACTPU response.	Out of synchronization on SSCP-PU session.
	0003	(SSCP,PU).SEC.SESS.RCV is not pending reset for the DACTPU response. VTAM was not waiting for a DACTPU response.	Out of synchronization on SSCP-PU session.
	0004	(SSCP,SLU).SEC.SESS.RCV is not pending active for the ACTLU response. VTAM was not waiting for an ACTLU response.	Timing between SSCP-PU and SSCP-LU sessions.
	0005	(SSCP,SLU).SEC.SESS.RCV is not pending reset for the DACTLU response. VTAM was not waiting for a DACTLU response.	Timing between SSCP-PU and SSCP-LU sessions.
	0006	(PLU,SLU).SEC.SESS.RCV is not pending active for a BIND response. VTAM was not waiting for a BIND response.	Timing between SSCP sessions and LU-LU session — SSCP session reset during LU-LU session activation.
	0007	(PLU,SLU).SEC.SESS.RCV is not pending reset for the UNBIND response. VTAM was not waiting for an UNBIND response.	Timing between SSCP sessions and LU-LU session — SSCP session reset during LU-LU session deactivation.
	0008	For an LU-LU session, the DATATRAFFIC_RCV is not active.	Timing on LU-LU session traffic.
ISTTSC6F	No discard codes from this module. Using TSALERT with reason code 0 for normal PIU discard.		
ISTTSC6I (LAN virtual link inbound data router)	0001	Discard the request or response, due to connection termination.	ISTTSCPD (PIU deblocker) determined that the PIU is not valid.

Table 78 (Page 12 of 12). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause (See Table 79)
ISTTSC6T (LAN virtual link connection terminator)	0001	Discard the request or response, due to connection termination.	The PIU was on the NCB work queue.
	0002	Discard the request or response, due to connection termination.	The PIU was on the NCB pending queue.
	0003	Discard the request or response, due to connection termination.	The PIU was on the inbound segmenting queue.
ISTTSC6Z	0001	End of PIU lost.	Error in network.
	0002	Retransmission of PIU, discard the partial PIU.	Error in network.
	0003	Segmenting length error.	Error in network.
	0004	Queue freed — segmenting error.	Error in network.
ISTTSC8X	0001	A PIU on the outbound work queue is discarded.	Multipath channel station (PU) is being deactivated.

Probable Causes of Discards

Table 79. Probable Causes of Discards

Probable Cause	Explanation of Cause
Timing on LU-LU session traffic	This discard can occur on the session between the application program and the SLU. It is usually caused by a reset request (for example, a Clear or a BIND) from the application program. This type of discard does not usually cause a problem in the network.
Timing between SSCP sessions and LU-LU session	This discard usually occurs when a reset of the SSCP-PU or SSCP-LU session affects a pending initiation or termination of the LU-LU session. For example, if the application program issues a CLSDST request to the logical unit, VTAM sends an UNBIND request to the logical unit. If the SSCP-LU session fails before the UNBIND response comes back, the UNBIND response is discarded. This type of discard does not usually cause a problem in the network.
Timing between SSCP-PU and SSCP-LU sessions	This discard usually occurs when a reset of the SSCP-PU session affects a pending initiation or termination of the SSCP-LU session. For example, if the operator deactivates the logical unit, VTAM sends a DACTLU to the logical unit. If the SSCP-PU session fails before the DACTLU response comes back, the DACTLU response is discarded. This type of discard does not cause a problem in the network.
Post-failure traffic purge	This discard may be caused by a network failure. Any PIU traffic arriving after the network failure but before the session is terminated can be purged using the discard function. For example, after a channel I/O error, PIUs that continue to be queued outbound over the channel can be discarded. This type of discard does not usually cause a problem in the network.
Internal VTAM logic error	This discard indicates a hardware or software error in the network; it usually occurs after a major SNA protocol violation. For example, if VTAM is waiting for a sequence number of 0008 but receives a PIU with sequence number 0010, PIUs with sequence numbers 0008 and 0009 are missing. The missing PIUs could have been lost anywhere in the session path, or perhaps they were not sent. The logical unit, physical unit, line, NCP, channel, or VTAM might be responsible for the problem. Further investigation is usually required to determine where in the network the problem is. The session type and reason code determine what type of documentation is needed to resolve a problem identified by this discard code.

Glossary, Bibliography, and Index

Glossary

For definitions of the terms and abbreviations used in this book, refer to the *VTAM Glossary*.

Bibliography

VTAM V4R2 Publications

Following are descriptions of the books in the VTAM V4R2 library for the MVS/ESA, VM/ESA, and VSE/ESA operating systems. The books are arranged here according to the tasks they describe:

- Planning
- Installation, Resource Definition, and Tuning
- Customization
- Operation
- Writing Application Programs
- Diagnosis
- Quick Reference.

The complete set of unlicensed books in this section can be ordered using a single order number, SBOF-4394 for MVS, SBOF-7001 for VM, and SBOF-7002 for VSE.

Softcopy Information

IBM Networking Softcopy Collection Kit CD-ROM (SK2T-6012):

The softcopy library contains softcopy versions of the licensed and unlicensed books for VTAM V4R2 and the VTAM AnyNet Feature for V4R2 for MVS/ESA.

All of the unlicensed and licensed VTAM books described in this section are available in softcopy on this CD-ROM. These softcopy files can be read using any of the IBM BookManager READ programs. They can also be read with the IBM Library Reader program shipped on this CD.

In addition, this CD contains the Online Message Facility. The Online Message Facility is an OS/2 program that provides online access to information from *VTAM Messages and Codes* and other BookManager softcopy books. The facility helps network operators and system programmers operate and diagnose problems without interrupting those tasks.

The CD also contains softcopy of the unlicensed books of many other products.

Marketing Information

A Networking Overview and the following IBM Networking Previews are available:

- AnyNet
- VTAM.

Ask your IBM marketing representative for more information.

Planning

Planning for NetView, NCP, and VTAM (SC31-7122):

This book helps you plan for new products or for migrating to new releases of networking products. It describes product functions, explains benefits you can gain from using them in different situations, and address cross-product implications. The book contains cross-task reference information and storage estimates.

Planning for Integrated Networks (SC31-7123):

This book helps you plan for SNA (subarea and APPN) and TCP/IP networks. It includes discussion of protocol strategies, migration scenarios, processing goals, and management considerations.

Planning Aids: Pre-Installation Planning Checklist for NetView, NCP, and VTAM (SX75-0092):

This checklist identifies important tasks to consider and complete before you begin to install these product. The document can be reproduced and folded to fit easily in a pocket or folder for quick reference and easy portability.

VTAM Licensed Program Specifications (GC31-6490):

This flyer is the warranty for VTAM and includes:

- A list of new functions
- Descriptions of VTAM features
- Machine requirements
- Programming requirements.

VTAM Release Guide for MVS/ESA (GC31-6492):

This guide provides an overview of the new functions in VTAM V4R2 and includes:

- Advantages of new functions
- Planning considerations for new functions
- Effect of new functions on existing functions
- Changes to commands, definition statements, and messages
- Programming requirements, such as the release of NCP required.

VTAM Release Guide for VM/ESA (GC31-8089):

This guide provides an overview of the new functions in VTAM V4R2 and includes:

- Advantages of new functions
- Planning considerations for new functions
- Effect of new functions on existing functions
- Changes to commands, definition statements, and messages
- Programming requirements, such as the release of NCP required.

VTAM Release Guide for VSE/ESA (GC31-8090):

This guide provides an overview of the new functions in VTAM V4R2 and includes:

- Advantages of new functions
- Planning considerations for new functions
- Effect of new functions on existing functions
- Changes to commands, definition statements, and messages
- Programming requirements, such as the release of NCP required.

VTAM Migration Guide for MVS/ESA (GC31-6491):

This guide helps you upgrade VTAM V4R1, V3R4.2, V3R4.1, V3R4, or V3R3 to VTAM V4R2. It includes:

- Planning to upgrade to VTAM V4R2
  - Upward and downward compatibility
  - Software and hardware requirements
  - Storage requirements
  - Impacts of new functions and enhancements performed without changes to user interfaces
  - Changes to installation process
- Upgrading user interfaces to VTAM V4R2
  - Changes to start options
  - Changes to buffer pools
  - Changes to definition statements
  - Changes to IBM-supplied default user-definable tables and modules
  - Changes to user-definable table macroinstructions
  - Changes to commands
  - Changes to messages
  - Changes to VTAM application programming interface
  - Changes to installation-wide exit routines
  - Changes to control blocks
- Implementing optional functions and enhancements introduced in VTAM V4R2
  - Overview of each new function and enhancement introduced since VTAM V3R3
  - Pointers to other books in the library where implementation details can be found.

VTAM Migration Guide for VM/ESA (GC31-8071):

This guide helps you upgrade VTAM V3R4.1 or V3R4 to VTAM V4R2. It includes:

- Planning to upgrade to VTAM V4R2
  - Upward and downward compatibility
  - Software and hardware requirements
  - Storage requirements
  - Impacts of new functions and enhancements performed without changes to user interfaces
  - Changes to installation process
- Upgrading user interfaces to VTAM V4R2
  - Changes to start options
  - Changes to buffer pools
  - Changes to definition statements
  - Changes to IBM-supplied default user-definable tables and modules
  - Changes to user-definable table macroinstructions
  - Changes to commands
  - Changes to messages
  - Changes to VTAM application programming interface
  - Changes to installation-wide exit routines
  - Changes to control blocks
- Implementing optional functions and enhancements introduced in VTAM V4R2
  - Overview of each new function and enhancement introduced since VTAM V3R4
  - Pointers to other books in the library where implementation details can be found.

VTAM Migration Guide for VSE/ESA (GC31-8072):

This guide helps you upgrade VTAM V3R4, V3R3, or V3R2 to VTAM V4R2. It includes:

- Planning to upgrade to VTAM V4R2
  - Upward and downward compatibility
  - Software and hardware requirements
  - Storage requirements
  - Impacts of new functions and enhancements performed without changes to user interfaces
  - Changes to installation process
- Upgrading user interfaces to VTAM V4R2
  - Changes to start options
  - Changes to buffer pools

  - Changes to definition statements
  - Changes to IBM-supplied default user-definable tables and modules
  - Changes to user-definable table macroinstructions
  - Changes to commands
  - Changes to messages
  - Changes to VTAM application programming interface
  - Changes to installation-wide exit routines
  - Changes to control blocks
- Implementing optional functions and enhancements introduced in VTAM V4R2
  - Overview of each new function and enhancement introduced since VTAM V3R2
  - Pointers to other books in the library where implementation details can be found.

Estimating Storage for VTAM (SK2T-2007):

This interactive program helps you estimate the storage requirements for VTAM. The diskette includes:

- Step-by-step procedures
- Formulas used to calculate storage.

VTAM Overview for VM/ESA and VSE/ESA (GC31-8114):

This document is designed to be used with each book of the VTAM library. It helps you understand which functions are included with each package available for VTAM V4R2 for VM/ESA and VSE/ESA.

- VTAM V4R2 Client/Server
- VTAM V4R2 MultiDomain
- VTAM V4R2 InterEnterprise.

It also provides instructions for how to order a particular package.

VTAM Glossary (GC31-6558):

This glossary defines terms and abbreviations for VTAM and related products. It includes information from the IBM *Dictionary of Computing*, SC20-1699.

Installation, Resource Definition, and Tuning

VTAM Network Implementation Guide (SC31-6494):

This book presents the major concepts involved in implementing a VTAM network, and includes:

- Buffer pools, slowdown, pacing, storage considerations
- Implementation considerations
- Installation procedures
- Sample major node definitions
- Migration considerations
- Tables and filters
- TSO, VSCS, VCNS, and other programs that run with VTAM
- Tuning procedures
- VTAM start options.

Use this book in conjunction with the *VTAM Resource Definition Reference*.

VTAM Resource Definition Reference (SC31-6498):

This book describes each VTAM definition statement, start option, and macroinstruction for user tables. It also describes NCP definition statements that affect VTAM. The information includes:

- IBM-supplied default tables (logon mode, USS and X.25)
- DTIGEN macroinstruction (VSCS start options)
- Major node definitions
- User-defined tables and filters
- VTAM start options.

If you are unfamiliar with the major concepts involved in implementing a VTAM network, use this book in conjunction with the *VTAM Network Implementation Guide*.

VTAM Resource Definition Samples (SC31-6499):

This book contains sample definitions to help you implement VTAM functions in your networks, and includes sample major node definitions.

Use this book in conjunction with the *VTAM Network Implementation Guide* and *VTAM Resource Definition Reference*.

Operation

VTAM Operation (SC31-6495):

This book serves as a reference for programmers and operators requiring detailed information about specific operator commands. The information includes:

- VTAM commands and start options
- Logon manager commands
- DISPLAY output examples (messages received)
- VSCS commands.

VTAM Operation Quick Reference (SX75-0205):

This book contains essential information about VTAM and VSCS operator commands.

VTAM Messages and Codes

(SC31-6493): This book describes messages, codes, and other information associated with VTAM messages and includes:

- Messages:
  - ELM messages for logon manager
  - IKT messages for TSO/VTAM
  - IST messages for VTAM network operators
  - ISU messages for sockets-over-SNA
  - USS messages
  - VSCS messages
- Codes and other information that display in VTAM messages:
  - Abend codes
  - Command and RU types in VTAM messages
  - Node and ID types in VTAM messages
  - Return codes for macroinstructions including ACB OPEN and CLOSE macroinstruction error fields, RTNCD-FDB2 return code combinations, and LU 6.2 RCPRI-RCSEC return codes
  - Sense codes including VTAM sense code hints, SNA sense field values for RPL-based macroinstructions, and 3270 SNA and non-SNA device sense fields
  - Status codes including resource status and session state codes
  - Wait state event codes and IDs
- Supplemental message-related information:
  - Message additions, deletions, and changes
  - Message flooding prevention
  - Message groups and subgroups
  - Message routing and suppression including descriptor codes, routing codes, and suppression levels for ELM, IKT, IST, and ISU messages
  - Message text and description formats
  - Message text of MSGLVL option messages including general information on the MSGLVL option
  - Message text of all VTAM network operator messages including variable field lengths
  - Online Message Facility.

Using IBM CommandTree/2

(SC31-7013): IBM CommandTree/2 is a workstation product that enables an operator to construct commands and send them to a specified destination for processing. The VTAM command set library includes:

- VTAM commands
- Logon manager commands
- Help for commands and start options.
- VSCS commands

Customization

VTAM Customization (LY43-0063):

This book enables you to customize VTAM, and includes:

- Communication network management (CNM) routing table
- Installing tables and modules in VM
- Logon-interpret routine requirements
- Logon manager installation-wide exit routine for the CLU search exit
- VSCS data manipulation installation-wide exit routine
- TSO/VTAM installation-wide exit routines
- VTAM installation-wide exit routines:
  - Command verification exit (ISTCMMND)
  - Configuration services XID exit (ISTEXCCS) with description of IBM-supplied default exit
  - Directory services management exit (ISTEXCDM)
  - Generic resource resolution exit (ISTEXCGR)
  - SDDLU exit (ISTEXCSD) with description of IBM-supplied default exit
  - Session accounting exit (ISTAUCAG)
  - Session authorization exit (ISTAUCAT)
  - Session management exit (ISTEXCAA) with example
  - TPRINT processing exit (ISTRAEUE)
  - USERVAR exit (ISTEXCUV) with description of IBM-supplied default exit
  - Virtual route pacing window size calculation exit (ISTPUCWC)
  - Virtual route selection exit (ISTEXCVR).

Writing Application Programs

VTAM Programming (SC31-6496): This book describes how to use VTAM macroinstructions to send data to and receive data from (1) a terminal in either the same or a different domain, or (2) another application program in either the same or a different domain. The information includes:

- API concepts
  - Cryptography
  - RUs and exchanges
  - Session establishment and termination
- BIND area format
- Communication Network Management Interface
- Dictionary of VTAM macroinstructions
- OPEN or CLOSE errors
- Operating system differences
- Program Operator Coding requirements
- RAPI DSECTs and control block mappings
- RAPI global variables
- Resource-identification and access-method-support vector lists
- RPL-based macroinstructions
- RPL RTNCD,FDB2 codes
- User exit routines.

VTAM Programming for LU 6.2 (SC31-6497): This book describes the VTAM LU 6.2 programming interface for host application programs. This book applies to programs that use only LU 6.2 sessions or that use LU 6.2 sessions along with other session types. (Only LU 6.2 sessions are covered in this book.) The information includes:

- Allocating and deallocating conversations
- APPCCMD macroinstructions and LU 6.2 DSECTs
- BIND image and response and ISTDBIND
- Conversation states
- Description and use of the following control blocks:
  - CNOS session limits control block
  - DEFINE control block
  - DISPLAY control block
  - RESTORE control block
- FMH-5 and PIP data
- LU 6.2 global variables
- Resource-identification and access-method-support vector lists
- RCPRI,RCSEC codes
- Sample program for retrieving RESTORE information
- Sample VTAM LU 6.2 application program
- Session- and conversation-level security and data encryption
- Sending and receiving data
- Sense codes for FMH-7 and UNBIND
- Summary of register usage
- Sync point services
- User exit routines.

Diagnosis

VTAM Diagnosis (LY43-0065): This book helps you identify a VTAM problem, classify it, and collect information about it before you call the IBM Support Center. The information collected includes traces, dumps, and other problem documentation. The information includes:

- Command syntax for running traces and collecting and analyzing dumps
- VIT entries
- Procedures for collecting documentation (VTAM, VSCS, TSO)
- VTAM internal trace and VIT analysis tool
- FFST Probes
- Channel programs
- Flow diagrams
- Procedures for locating buffer pools
- VSCS dump and traces
- CPCB operation codes
- Storage and control block ID codes
- PIU discard reason codes
- Offset names and locations for VTAM buffer pools

VTAM Data Areas for MVS/ESA (LY43-0064): This book describes VTAM data areas and can be used to read a VTAM dump. It is intended for IBM programming service representatives and customer personnel who are diagnosing problems with VTAM.

VTAM Data Areas for VM/ESA (LY43-0103): This book describes VTAM data areas and can be used to read a VTAM dump. It is intended for IBM programming service representatives and customer personnel who are diagnosing problems with VTAM.

VTAM Data Areas for VSE/ESA (LY43-0104): This book describes VTAM data areas and can be used to read a VTAM dump. It is intended for IBM programming service representatives and customer personnel who are diagnosing problems with VTAM.

VTAM AnyNet Feature for V4R2 for MVS/ESA

VTAM AnyNet Feature for V4R2: Guide to Sockets over SNA Gateway for OS/2 (SC31-6528):

This guide provides information to help you install, configure, use, and diagnose the sockets-over-SNA-gateway function for OS/2. This function allows socket applications running on a TCP/IP network to communicate with socket applications running on an SNA network.

VTAM AnyNet Feature for V4R2: Guide to Sockets over SNA (SC31-6526):

This guide provides information to help you install, configure, use, and diagnose Sockets over SNA. It also provides information to help you prepare application programs to use sockets over SNA.

VTAM AnyNet Feature for V4R2: Guide to SNA over TCP/IP (SC31-6527):

This guide provides information to help you install, configure, use, and diagnose SNA over TCP/IP.

MPTN Architecture Publications

Networking Blueprint Executive Overview (GC31-7057)

Multiprotocol Transport Networking: Technical Overview (GC31-7073)

Multiprotocol Transport Networking: Formats (GC31-7074)

APPC Application Suite Feature for V4R2 for MVS/ESA

APPC Application Suite User's Guide (SC31-6532):

This book documents the end-user interface (concepts, commands, and messages) for the AFTP, ANAME, and APING facilities of the APPC Application Suite for MVS/ESA. Although it's primary audience is the end user, administrators and application programmers may also find it useful.

APPC Application Suite Administration (SC31-6533):

This book contains the information that administrators need to configure the APPC Application Suite for MVS/ESA and to manage the APING, ANAME, AFTP, and A3270 servers.

APPC Application Suite Programming (SC31-6534):

This book provides the information application programmers need to add the functions of the AFTP and ANAME APIs to their application programs.

Table 80 (Page 1 of 3). Related Information on Problem Topics in Other Libraries

Topic	See
3174 Controller	*3174 Functional Description*
Abend codes	*MVS/ESA System Codes* *VM/ESA System Messages and Codes*
Abend dump	*MVS/ESA JCL Reference*
ACF/TAP	*NCP, SSP, and EP Diagnosis Guide*
Alert major vector and its subvectors	*SNA Network Product Formats*
Alerts	*NetView Operation* *SNA Network Product Formats*
Allocating extents	*Planning for NetView, NCP, and VTAM*
CCS	*VM/ESA CP Programming Services*
CCW trace	*MVS/ESA Diagnosis: Tools and Service Aids*
Channel programs	*IBM 4361 Processor Communication Adapter* *IBM 9370 Information System: Telecommunications Subsystem Description and Reference* *Principles of Operation* manual for your communication controller *Principles of Operation* manual for your operating system
CNOS return codes	*VTAM Programming for LU 6.2*
Communication scanner output	*NCP, SSP, and EP Diagnosis* *NCP Reference Summary and Data Areas* *Principles of Operation* manual for your communication controller
CP operator commands, CP SIO	*VM/ESA CP Command and Utility Reference*
CP TRACE command	*VM/ESA CP Command and Utility Reference* *VM/ESA Virtual Machine Operation*
CP traces	*VM/ESA System Operation*
DEDICATE directory statement	*VM/ESA CP Planning and Administration*
Directory services management exit	*VTAM Customization*
DOSVSDMP dump contents	*VSE/ESA Diagnosis Tools*
Dump collecting, formatting, and printing	*MVS/ESA Diagnosis: Tools and Service Aids* *VM/ESA Diagnosis Guide* *VSE/ESA Diagnosis Tools*
Dump information, retrieving from tape	*VSE/ESA Diagnosis Tools*
Dumping CP storage	*VM/ESA System Operation*
DUMPLOAD	*VM/ESA Dump Viewing Facility*
Dynamic tracing of ICAs	*VSE/ESA Diagnosis Tools*
EREP	*EREP User's Guide and Reference* *VM/ESA System Operations*
ERP	*SYS1.LOGREC Error Recording*
Exception request (EXR)	*SNA Network Product Formats*
First Failure Support Technology (FFST)	*First Failure Support Technology for VM and MVS Operator's Guide*
Full-screen mode	*TSO/E Programming Services*
GCS commands	*VM/ESA Group Control System*
GCS SIO	*VM/ESA Diagnosis Guide*
Generalized trace facility (GTF)	*MVS/ESA Diagnosis: Tools and Service Aids*
Generic alerts	See *Alerts.*
HCPRIO	*VM/ESA Planning and Administration*

Table 80 (Page 2 of 3). Related Information on Problem Topics in Other Libraries

Topic	See
Hung NCP, Hung resources attached to an NCP, Hung sessions	*NCP, SSP, and EP Diagnosis Guide*
ICA interface to VTAM ICA sense codes	*IBM 4361 Processor Communication Adapter* *IBM 9370 Information System: Telecommunications Subsystem Description and Reference*
IEBGENER utility	*MVS Utilities*
Intensive mode recording	*NCP, SSP, and EP Diagnosis Guide*
I/O control blocks	*MVS/ESA Diagnosis: Data Areas* *VM/ESA Diagnosis Guide* *VSE/ESA Diagnosis Tools*
I/O traces	*MVS/ESA Diagnosis: Data Areas* *VM/ESA CP Command and Utility Reference* *VM/ESA Diagnosis Guide* *VSE/ESA Diagnosis Tools*
IPCS, running in batch mode, IPCS commands	*MVS/ESA Interactive Problem Control System (IPCS) Commands* *MVS/ESA Interactive Problem Control System (IPCS) User's Guide*
IPCSPRNT	*MVS/ESA Interactive Problem Control System (IPCS) Customization*
IUCV IUCV external interrupts	*VM/ESA CP Programming Services*
IUCV CONNECT return codes	*GCS Reference for ESA*
Job control language (JCL)	*MVS/ESA JCL Reference* *MVS/ESA JCL User's Guide*
Line trace records	*NCP, SSP, and EP Diagnosis Guide*
LIST Service Aid	*MVS/ESA Diagnosis: Tools and Service Aids*
LOGDATA option	*MVS/ESA Interactive Problem Control System (IPCS) Commands*
LOGREC	*SYS1.LOGREC Error Recording* *EREP User's Guide and Reference*
Maintain System History Program (MSHP)	*VSE/ESA System Control Statements*
MVS IKJxxxx system messages	*MVS System Messages*
MVS system codes	*MVS/ESA System Codes*
NCP data areas, registers, and codes	*NCP and EP Reference Summary and Data Areas*
NCP dumps, NCP service aids	*NCP, SSP, and EP Diagnosis Guide*
NCP, tuning	*NTune User's Guide*
NetView Session Monitor	*NetView Operation*
NMVT RUs	*SNA Network Product Formats* *NetView Operation*
Program status words (PSWs) PSWs, locating in dump output	*Principles of Operation* manuals *VSE/ESA Diagnosis Tools*
PRTDUMP	*VM/ESA Dump Viewing Facility*
RECMS RU formats	*NCP and EP Reference Summary and Data Areas*
Reshow processing	*TSO/E Programming Services*
Retrieving dump information from tape	*VSE/ESA Diagnosis Tools*
RTCT	*MVS/ESA Diagnosis: Data Areas* *VM/ESA Diagnosis Guide*
RU opcodes	*SNA Formats*
SDAID	*VSE/ESA Diagnosis Tools*

Table 80 (Page 3 of 3). Related Information on Problem Topics in Other Libraries

Topic	See
SDATA options	*MVS/ESA Operations: System Commands*
SDUMP	*VM/ESA Diagnosis Guide*
Session monitor	See *NetView Session Monitor.*
SIOF	*VM/ESA CP Command and Utility Reference*
SLIP dump	*MVS/ESA Diagnosis: Tools and Service Aids*
SMP	*System Modification Program Extended User's Guide*
SNA sense codes	*SNA Formats*
Socket API calls	*TCP/IP for MVS Programmer's Reference*
SPTAPE	*VM/ESA CP Command and Utility Reference*
Stand-alone dump	*MVS/ESA Diagnosis: Tools and Service Aids*
STATMON	See *NetView Session Monitor.*
SVC dump	*MVS Operations: System Commands*
SVC 93 and SVC 94 entries	*MVS/ESA Diagnosis: Tools and Service Aids*
SYSREC	*VSE/ESA System Control Statements*
Task control blocks	*MVS/ESA Diagnosis: Data Areas* *VM/ESA Diagnosis Guide* *VSE/ESA Diagnosis Tools*
TCB, map of	*MVS/ESA Diagnosis: Data Areas*
TGET options, TGET return codes	*TSO/E Programming Services*
TGET/TPUT option flags	*MVS/ESA Diagnosis: Reference*
Time of day clock entry in trace output	*System/370 ESA Principles of Operation*
TPUT options, editing done by	*TSO/E Programming Services*
TRACERED	*VM/ESA Dump Viewing Facility*
TRSAVE, TRSOURCE	*VM/ESA CP Command and Utility Reference*
Virtual machine branch trace, VM commands	*VM/ESA CP Command and Utility Reference*
VMFPLC2	*VM/ESA Service Guide*
VMFZAP	*VM/ESA Service Guide*
VSE dumps, obtaining and formatting	*VSE/ESA Diagnosis Tools*

Index

Special Characters

Numerics

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Communicating Your Comments to IBM

VTAM™
Diagnosis
Version 4 Release 2 for MVS/ESA, VM/ESA, and VSE/ESA
Publication No. LY43-0065-01

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

If you are mailing a readers' comment form (RCF) from a country other than the United States, you can give the RCF to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by mail, use the RCF at the back of this book.
- If you prefer to send comments by FAX, use this number:

  United States and Canada: **1-800-227-5088**
- If you prefer to send comments electronically, use this network ID:
  - IBM Mail Exchange: **USIB2HPD at IBMMAIL**
  - IBMLink: **CIBMORCF at RALVM13**
  - Internet: **USIB2HPD@VNET.IBM.COM**

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies.

Help us help you!

VTAM™
Diagnosis
Version 4 Release 2 for MVS/ESA, VM/ESA, and VSE/ESA
Publication No. LY43-0065-01

We hope you find this publication useful, readable and technically accurate, but only you can tell us! Your comments and suggestions will help us improve our technical publications. Please take a few minutes to let us know what you think by completing this form.

Overall, how satisfied are you with the information in this book?	Satisfied	Dissatisfied
	☐	☐

How satisfied are you that the information in this book is:	Satisfied	Dissatisfied
Accurate	☐	☐
Complete	☐	☐
Easy to find	☐	☐
Easy to understand	☐	☐
Well organized	☐	☐
Applicable to your task	☐	☐

Specific Comments or Problems:

Please tell us how we can improve this book:

Thank you for your response. When you send information to IBM, you grant IBM the right to use or distribute the information without incurring any obligation to you. You of course retain the right to use the information in any way you choose.

Name

Address

Company or Organization

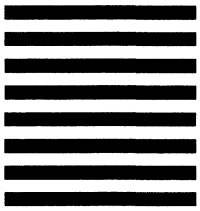
Phone No.

Help us help you!
LY43-0065-01

Cut or F
Along L

Fold and Tape | **Please do not staple** | Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES



BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

Information Development
Department E15
International Business Machines Corporation
PO BOX 12195
RESEARCH TRIANGLE PARK NC 27709-9990

Fold and Tape | **Please do not staple** | Fold and Tape

LY43-0065-01

Cut or
Along L

Help us help you!

VTAM™
Diagnosis
Version 4 Release 2 for MVS/ESA, VM/ESA, and VSE/ESA
Publication No. LY43-0065-01

We hope you find this publication useful, readable and technically accurate, but only you can tell us! Your comments and suggestions will help us improve our technical publications. Please take a few minutes to let us know what you think by completing this form.

Overall, how satisfied are you with the information in this book?	Satisfied	Dissatisfied
	☐	☐

How satisfied are you that the information in this book is:	Satisfied	Dissatisfied
Accurate	☐	☐
Complete	☐	☐
Easy to find	☐	☐
Easy to understand	☐	☐
Well organized	☐	☐
Applicable to your task	☐	☐

Specific Comments or Problems:

Please tell us how we can improve this book:

Thank you for your response. When you send information to IBM, you grant IBM the right to use or distribute the information without incurring any obligation to you. You of course retain the right to use the information in any way you choose.

Name

Address

Company or Organization

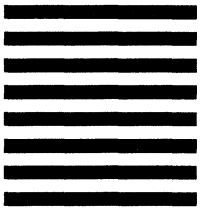
Phone No.

Help us help you!
LY43-0065-01

Cut or Along

Fold and Tape | **Please do not staple** | Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES



BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

Information Development
Department E15
International Business Machines Corporation
PO BOX 12195
RESEARCH TRIANGLE PARK NC 27709-9990

Fold and Tape | **Please do not staple** | Fold and Tape

Cut o Along

Help us help you!

VTAM™
Diagnosis
Version 4 Release 2 for MVS/ESA, VM/ESA, and VSE/ESA
Publication No. LY43-0065-01

We hope you find this publication useful, readable and technically accurate, but only you can tell us! Your comments and suggestions will help us improve our technical publications. Please take a few minutes to let us know what you think by completing this form.

Overall, how satisfied are you with the information in this book?	Satisfied	Dissatisfied
	☐	☐

How satisfied are you that the information in this book is:	Satisfied	Dissatisfied
Accurate	☐	☐
Complete	☐	☐
Easy to find	☐	☐
Easy to understand	☐	☐
Well organized	☐	☐
Applicable to your task	☐	☐

Specific Comments or Problems:

Please tell us how we can improve this book:

Thank you for your response. When you send information to IBM, you grant IBM the right to use or distribute the information without incurring any obligation to you. You of course retain the right to use the information in any way you choose.

Name

Address

Company or Organization

Phone No.

Help us help you!
LY43-0065-01

Cut or
Along

Fold and Tape **Please do not staple** Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

Information Development
Department E15
International Business Machines Corporation
PO BOX 12195
RESEARCH TRIANGLE PARK NC 27709-9990

Fold and Tape **Please do not staple** Fold and Tape

LY43-0065-01

Cut o
Along

Help us help you!

VTAM™
Diagnosis
Version 4 Release 2 for MVS/ESA, VM/ESA, and VSE/ESA
Publication No. LY43-0065-01

We hope you find this publication useful, readable and technically accurate, but only you can tell us! Your comments and suggestions will help us improve our technical publications. Please take a few minutes to let us know what you think by completing this form.

Overall, how satisfied are you with the information in this book?	Satisfied	Dissatisfied
	☐	☐

How satisfied are you that the information in this book is:	Satisfied	Dissatisfied
Accurate	☐	☐
Complete	☐	☐
Easy to find	☐	☐
Easy to understand	☐	☐
Well organized	☐	☐
Applicable to your task	☐	☐

Specific Comments or Problems:

Please tell us how we can improve this book:

Thank you for your response. When you send information to IBM, you grant IBM the right to use or distribute the information without incurring any obligation to you. You of course retain the right to use the information in any way you choose.

Name

Address

Company or Organization

Phone No.

Help us help you!
LY43-0065-01

Cut o
Along

Fold and Tape | **Please do not staple** | Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

Information Development
Department E15
International Business Machines Corporation
PO BOX 12195
RESEARCH TRIANGLE PARK NC 27709-9990

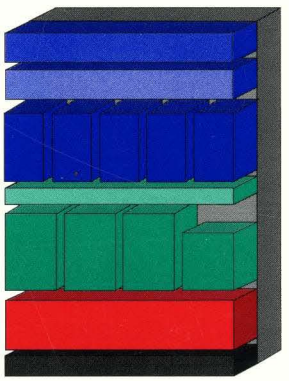
Fold and Tape | **Please do not staple** | Fold and Tape

LY43-0065-01

Cut o
Along

File Number: S370/4300/30XX-50
Program Number: 5695-117 (MVS/ESA)
5654-010 (VM/ESA)
5686-065 (VSE/ESA)

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber



LY43-0065-01