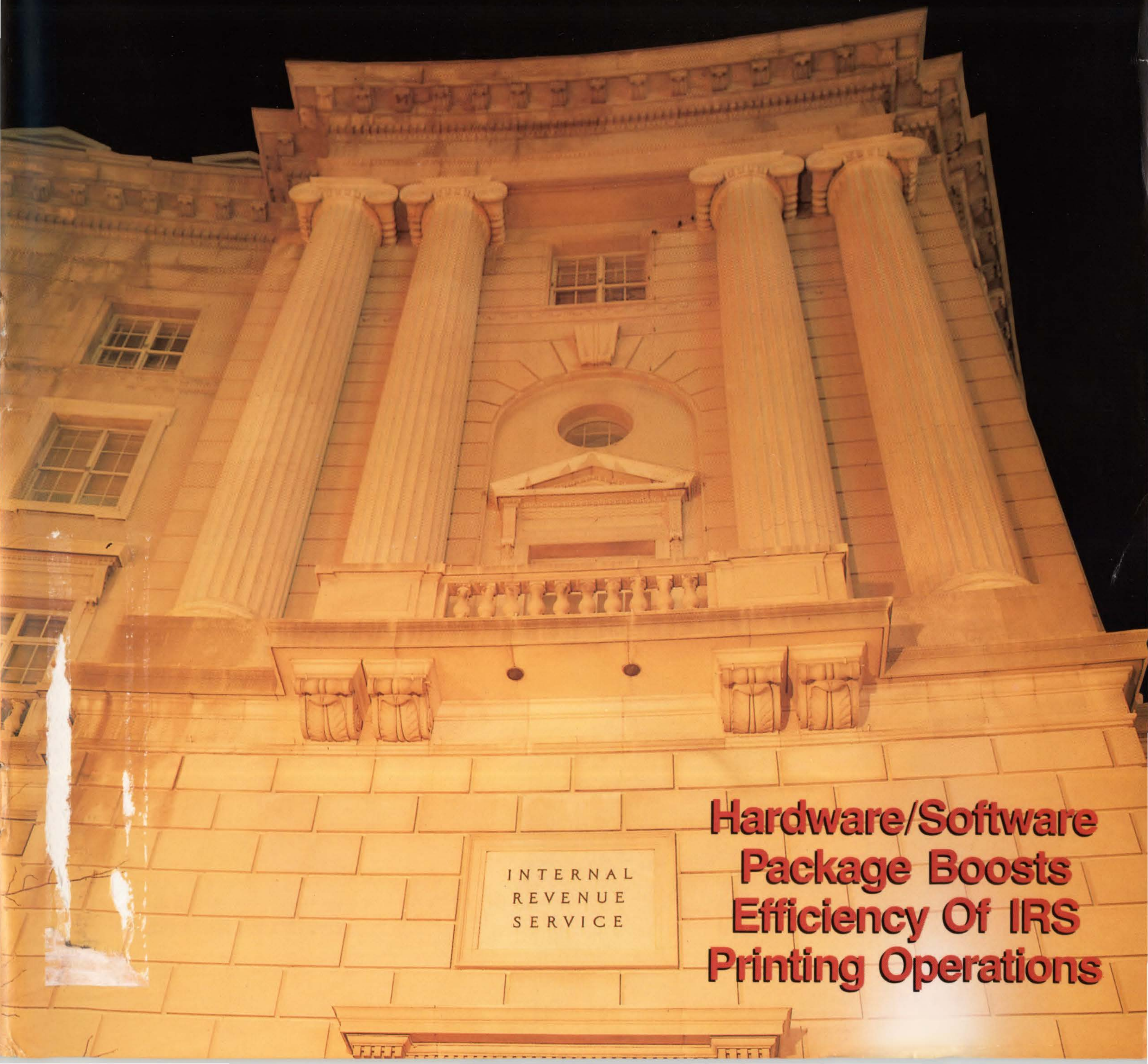
MAINFRAME JOURNAL

For Users of IBM System/370 Architecture & Compatible Systems

July 1989



Hardware/Software Package Boosts Efficiency Of IRS Printing Operations

VMCENTER II™: MASTER THE VM™ ENVIRONMENT.

The #1 systems management tool for VM environments.

Storm clouds gathering over your data center? Don't wait for the deluge. Get VMCENTER II and stay in control for good.

VMCENTER II is a unique software product — an unmatched resource for managing all your VM operations. With VMCENTER II, you can simplify access without endangering security. Save on DASD without compromising performance. Enhance service to users while streamlining administration. And control local and remote sites with unprecedented effectiveness.

With the VMCENTER II automated facilities, you can manage present-day operations more easily than ever. And in the time left over, plan for the future with confidence and accuracy — thanks to comprehensive online utilization and trend reports.

No wonder VMCENTER II is the most widely used systems management tool for VM environments. It's the single most important investment you can make in the quality and cost-effectiveness of your VM operations.

So why wait any longer? Gain the upper hand today. And keep the storm clouds at bay for years to come. For more information on VMCENTER II, call or write Systems Center, Inc., 1800 Alexander Bell Drive, Reston, VA 22091. A New York Stock Exchange Company.

800-562-7100 703-264-8000

SYSTEMS CENTER

FORMERLY VM SOFTWARE, INC.

VM SOFTWARE PRODUCTS NETWORK DATAMOVER PRODUCTS RELATIONAL DATABASE PRODUCTS NETWORK ADMINISTRATION PRODUCTS

CIRCLE #20 on Reader Service Card ▲

1-MFJ-8907

C O N T E N T S

July 1989 • Volume IV • Number 7

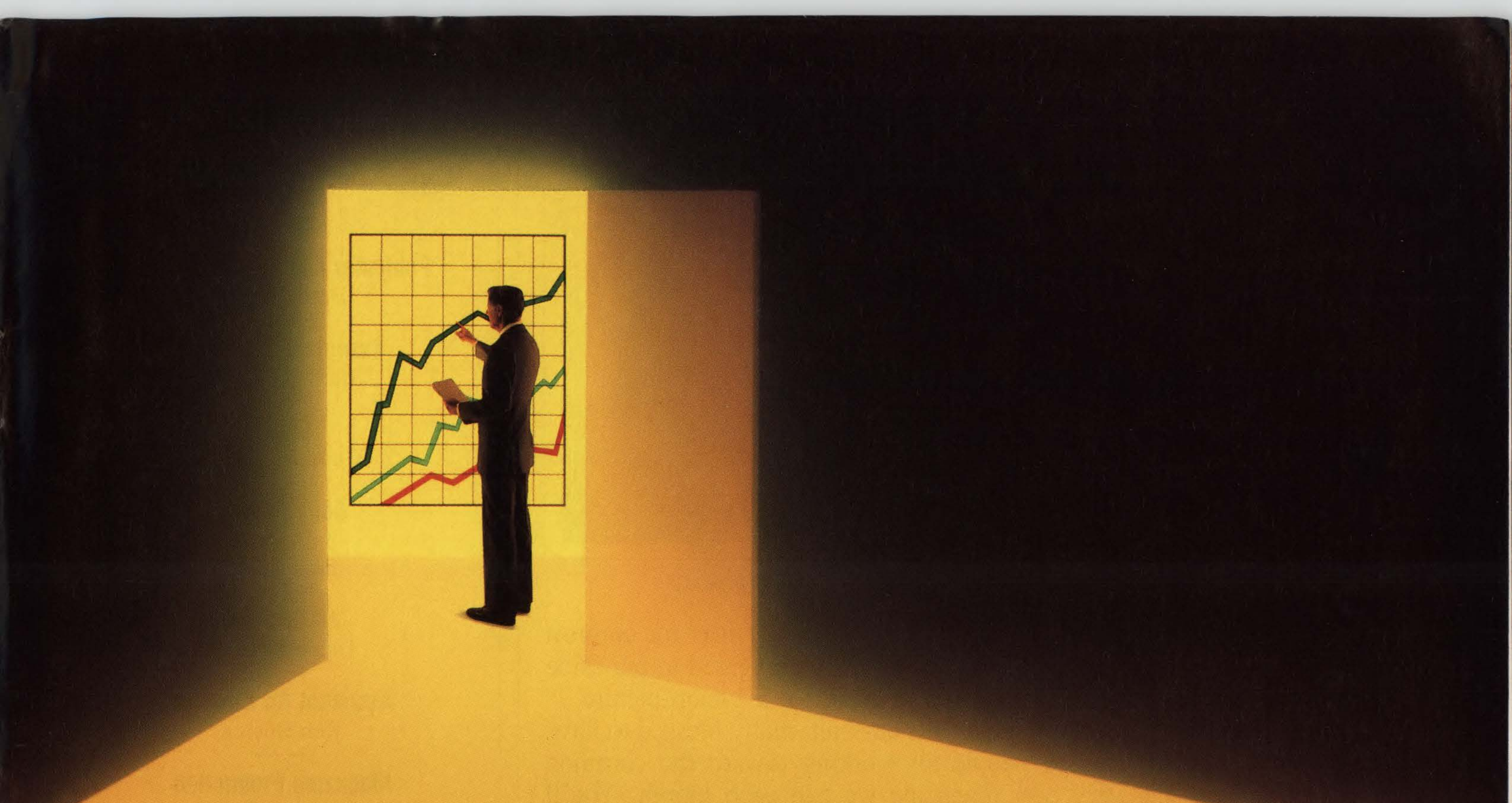
DEPARTMENTS

COVER:

Printing operations at the Remote Communications Center in IRS headquarters, Washington, DC, conquered its printing problem in an unlikely manner outlined in the article on page 77. Cover photography is by Randy Batista, Media Image Photography, Gainesville, FL.

MAINFRAME JOURNAL© (ISSN 0895-5751) is published monthly by Thomas Publications, Inc., 10935 Estate Lane, Suite 375, Dallas, TX 75238, (214) 343-3717. Second class postage paid at Dallas, TX. SUBSCRIPTION RATE: Subscriptions are free within the USA and Canada. One-year foreign subscriptions are $96. POSTMASTER: Send address changes to: *MAINFRAME JOURNAL*, P.O. Box 551628, Dallas, TX 75355-1628.

F E A T U R E S



Systems software for MVS data centers:

Enter the world of total performance measurement, total support.

Computer Associates presents the industry's first integrated solution incorporating performance measurement, capacity planning, resource management, network monitoring and job accounting:

CA-UNIPACK™/PMA
PERFORMANCE MEASUREMENT AND ACCOUNTING

Consisting of CA-FASTDASD™, CA-ISS/THREE™, CA-MAZDAMON™, CA-JARS®, CA-JARS®/CICS, CA-JARS® DSA, CA-LOOK® and CA-MINDOVER®.

With **CA-UNIPACK/PMA**, you can fine tune your system, improving its performance and optimizing the utilization of resources. You can reduce costs by accurately forecasting future hardware needs thereby avoiding unnecessary and costly additions or upgrades. You can accurately charge back costs to end users, allowing you to treat the data center as a profit center. And most importantly, you can easily achieve the CICS response times end users expect, enabling you to consistently maintain service level agreements.

Only Computer Associates has the proven products and expertise to offer this cost-effective, total performance solution.

And only Computer Associates offers **CA-UNISERVICE®/II**, a secure link between your mainframe and CA's Customer Service System 24 hours a day. You get online access to software fixes, interactive problem resolution, plus product tutorials and more! No one else has anything like it.

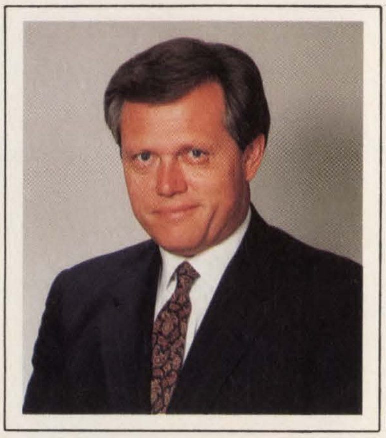
Call Dana Williams today:
800-645-3003

COMPUTER ASSOCIATES®
Software superior by design.

- World's leading independent software company.
- Broad range of integrated business and data processing software for mainframe, mid-range and micro computers.
- Worldwide service and support network of more than 100 offices.

Resource & Operations Management • Financial • Banking • Graphics • Spreadsheets • Project Management

CIRCLE #105 on Reader Service Card ▲



Bob Thomas

Team Picture

The quality you have come to expect from each issue of *MAINFRAME JOURNAL* is directly attributable to the outstanding authors who submit the articles as well as the terrific group of people you see below.

On normal workdays we're a pretty scruffy bunch. Dress for comfort is the "dress code." However, one day recently we all agreed to put aside our jeans, warm-ups and NIKEs just long enough for a "team picture."

Please excuse the corny family pride bit, but I am really blessed to have such a great group of talented people all working toward the common goal of putting out the very best magazine we can each month. You'll never find a more dedicated, harder working, friendlier group of people anywhere!

This issue is being sent to almost 70,000 DP professionals using IBM (and compatible) mainframe computer systems. Just 18 months ago less than 40,000 copies of *MAINFRAME JOURNAL* were sent out. This represents a growth rate of 175 percent in a very short time. More importantly, we also doubled the frequency from bimonthly to monthly without a hiccough in quality. The "team" pictured below is responsible and I am extremely proud of them.

Bob Thomas

The MAINFRAME JOURNAL *staff from left to right are: standing — Marian Davenport, David Kramer, Judy Beller, Diane Dishman, Martha Thomas, Nancy Crawford, Pat Warner, Mary Thomas, Sally Webb and Ken Buerer; seated — Mark Cauto, Janice Porter, Carol Hoag, Bob Thomas and Denise Thomas.*

MAINFRAME JOURNAL
For Users of IBM System/370 Architecture & Compatible Systems

Publisher and Editor-in-Chief
Bob Thomas

Associate Publisher
Martha Thomas

Editor
Carol M. Hoag

Copy Editors
Judy Beller
Pat Warner

Art Director
David Kramer

Assistant Art Director
Ken Buerer

Magazine Production
Diane Dishman

Marketing Services
Sally Webb

Circulation Manager
Janice Porter

Assistant Circulation Manager
Nancy Crawford

Administrative Manager
Marian Davenport

Advertising Sales
Denise Thomas — Mark Cauto

BPA

List Rental Information
Contact AllMedia (214) 985-4060

SUBSCRIPTION RATES & INQUIRIES: Subscriptions are free within the USA and Canada. One-year foreign subscriptions are $96.

All subscriptions, remittances, requests and changes of address should be sent to *MAINFRAME JOURNAL* at 10935 Estate Lane, Suite 375, Dallas, TX 75238, (214) 343-3717.

VSE & IBM

Pete Clark's article on "VSE Announcements" (April 1989) was informative. However, we must differ with his conclusions regarding the statement: "The bottom line is that IBM is continuing to maintain the vitality of VSE." Yes, IBM has upgraded VSE with two new versions. Notwithstanding, most of the upgrades can be classified as either cosmetic or forced by required support so that IBM could sell more hardware (for example, 3990, 3745, 9370 and so on). The major announcement was an increase in the number of regions from three to nine and an increase from 40MB to 128MB of virtual storage was already available from Clark for *free*. The other features were already available through third-party software. The inclusion of the BUFNI/BUFND parameter in the JCL was long overdue. "Vitality" is the addition of 31-bit addressing, XA I/O, multiprocessing, more control over multiprogramming, data spaces, hiperspaces and others.

The increased VSE price mentioned in Clark's article is the real issue. The dollar amounts quoted represent a whopping 78 percent increase. The new enhancements which are not hardware product support are not worth the 78 percent increase. The increased software revenue does not include the revenue generated by the sale of new hardware products which required this upgrade. The net message in Clark's article is that system software is an excellent revenue producer for IBM. At some point in time, older releases of VSE/SP will become "non-supported" and customers will be forced to upgrade. Assuming an average upgrade charge of $30K, these upgrades will result in a $900 million charge for the 30,000 customers mentioned in his article. Not a bad income for a "dead product." Unfortunately, when it comes to operating system software, we are all hostages to the manufacturer's marketing (sales) strategy.

Eugene S. Hudders
President
Multiple Computer Services, Inc.
San Juan, PR

JES3 Resource Management

In Jon E. Pearkins' article, "JES3: Is It Worth The Conversion Costs?" (March 1989), he provided an excellent and balanced overview of the issues an installation faces when considering which MVS spooling system to use. However, JES3 resource management deserves a closer look.

HASP and ASP were both developed to improve OS/360 throughput. Both systems provided better spooling performance than native OS/360 spooling, so much so that native spooling was eliminated in MVS. ASP provided additional support to address two major bottlenecks in OS/360, tape mounts and dataset allocation.

OS/360 waited for a tape mount while holding the system allocation ENQUEUE. Consequently, until the tape mount was satisfied, other jobs could not be scheduled. The lockout led to idle regions or partitions even though work was available for them. Thus, delays for tape mounts translated directly into reduced system throughput. ASP solved this problem by premounting tapes prior to selecting jobs for execution. MVS, unlike OS/360, DEQUEUEs before waiting for tape mounts, so jobs may be scheduled during tape mounts.

OS/360 systems used real rather than virtual storage. Worse yet, the hardware OS/360 ran on was typically starved for memory. For example, the 360/65, a 1/2 MIPS processor, supported a maximum main memory size of one megabyte. One performance rule of thumb is that 360 and 370 system software (OS/360, SVS, MVS, VSE or VM) works best when the MIPS to megs ratio is roughly 1:4. By that rule of thumb, the 360/65 was memory-starved by 50 percent.

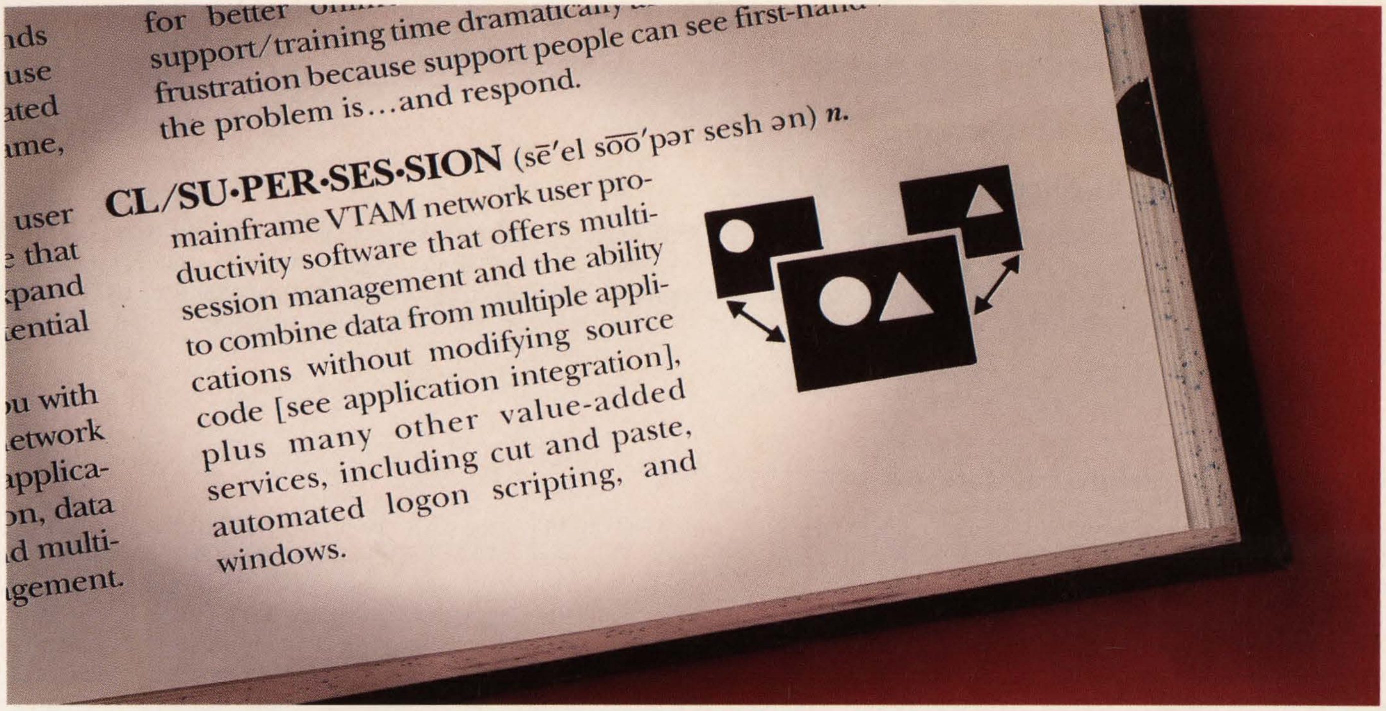
In that memory-starved environment, partitions or regions sidelined due to dataset contention aggravated the problem. ASP solved the problem by pre-checking dataset ENQUEUEs and not selecting jobs unless their path to successful ENQUEUE was clear. MVS systems, running on today's processors, are less vulnerable to these problems because of virtual memory and more reasonable amounts of real storage.

I must take issue with Pearkins on a point he makes on page 78. Jobs get into ENQUEUE contention during job initiation, rather than step initiation. In my experience, an installation strategy of canceling and re-queuing such jobs has never produced problems of restarting partially-run jobs. Such cancellations occur before the job has created anything that could cause it problems later.

JES3 tape management provides an installation with some benefits in controling tape allocation. However, the benefits are not as large as those provided to OS/360 due to MVS' improved architecture. An installation needs some protection from dataset contention. Whether JCL coding standards, installation exits, an alert operations staff or JES3 is the answer depends very much on the operating environment. Because JES3 manages resources at the job level, an installation needs to know how many steps and what allocations its typical job has. If only one step of a multi-step job requires a tape or dataset, JES3's management might reduce tape or dataset utilization by needlessly delaying jobs. In such a situation, JES2's contention approach might actually produce better overall throughput. (MVS deallocates and DEQUEUEs on a step basis.) JES3 does have facilities for delayed setup and early breakdown that might mitigate these problems somewhat. Finally, increasing use of dynamic allocation limits how effective any approach based, like JES3's, on JCL scanning can be. One cannot recognize dynamic allocation by JCL scanning.

So how do I decide between JES2 and JES3? Most installations don't have to confront this issue. Usually, existing investments in products and systems preclude converting between spooling systems. For those few installations not coming to MVS from its precursors, however, Pearkins' article provides valuable guidance. The one point I would add is that none of JES3's additional functions come free: they all require longer instruction path lengths. If those longer path lengths translate into a significantly larger CPU consumption by JES3 than would have been experienced under JES2, the installation must decide whether the CPU time is being wisely invested. For example, every one percent increase in CPU time on a $5 million processor represents a cost of $50,000. If the CPU time provides an installation with comparable value, JES3 is probably the answer. In some environments things like Deadline Scheduling and Dependent Job Scheduling represent real value and may satisfy the installation's requirement for a job scheduling product. In other environments, JES3 features may have nothing to do with the installation's requirements and objectives and, therefore, should not be part of the evaluation.

William S. Mosteller, CDP
Systems Center, Inc.
Reston, VA



Only one product fits the definition of "application integration."

Application integration: we defined it before we designed it. Candle's CL/SUPERSESSION® for MVS offers VTAM network customization and application integration *by design.* Every feature is there for a simple reason: to expand your network's scope by giving you greater flexibility and important value-added services.

CL/SUPERSESSION makes it simple to customize applications across systems *without modifying a single line of source code.* Its dynamic cut-and-paste facility lets you quickly combine information from multiple sources into a single display – or distribute information from one screen to a multitude of applications. So you can virtually eliminate duplicate data entry.

CL/SUPERSESSION's smooth application integration is just one of many productivity payoffs. Because its multi-sessioning features exploit the latest advances in MVS and VTAM, an unlimited number of VTAM end users can have access to unlimited concurrent sessions. For additional power, CL/SUPERSESSION's unique Dialog Manager helps you quickly customize complex network procedures. The once frustrating task of manual logon can now be fully automated...and tailored to each user.

To improve network financial planning, CL/SUPERSESSION's Network Accounting Facility provides complete activity recording, including all network logons, session initiations/terminations, and traffic volumes for precise chargeback billing and usage auditing. And for at-a-glance monitoring of multiple sessions, our new windowing feature will let you view sessions side by side on a single screen. What's more, you'll never compromise on security. CL/SUPERSESSION interfaces with RACF, ACF2, and TOP SECRET.

Application integration and network customization *must* be simple, secure, and streamlined for your shop to boost productivity. That's why there's only one definition worth looking up: CL/SUPERSESSION for MVS. We'll be glad to spell out the details. Just call us today at **(312) 954-3888** (in Illinois) or **(800) 762-7608** (outside Illinois).

Candle®

CIRCLE #116 on Reader Service Card ▲

VSE Now "Strategically Important" To IBM

In a recent high-level conference on VSE Status & Trends, IBM made several pronouncements that should be music to the ears of the 20,000-plus VSE users everywhere. In addition to the key statement that VSE is considered to be of strategic importance to IBM's future, IBM also declared that VSE will support the important subsets of SAA. In a little logic exercise, IBM went on to state that since: a.) the 9370 Series will support ESA, and b.) VSE will support the 9370 Series, therefore, yes, c.) VSE will support ESA.

Rejuvenated IBM Mainframe Market

After maintaining a growth rate of one percent per year from 1984 to 1987, the U.S. population of IBM mainframes has increased a healthy 10 to 11 percent over each of the last two years. This increase has been powered by 309X and 9370 Series systems and, to a lesser extent, 4381 Series systems. The 309X Series population continues to show tremendous growth increasing by more than 1,000 systems each year since 1986. The 9370 Series population has grown to nearly 5,000 systems installed in the U.S. The "new life" of the IBM mainframe market is partially attributed to 11 percent of the new/on-order 309Xs which are not replacing currently installed IBM mainframes. Four percent of these systems are replacing "non-IBM" systems which include processors from Amdahl, NAS, Unisys and Prime. The other seven percent are being installed at "new" sites or as additional systems at a site (not replacing a currently installed system). Of the 9370 Series, only 10 percent of the new/on-order systems are replacing other models within the Series. The greatest percentage (51 percent) are replacing 4300 Series systems and a whopping 31 percent are being installed as new or additional systems. In spite of the declining population of 4300 and 308X models and the fact that the 309X Series is nearing its peak, the announcement of the "Summit," as well as a follow-on to the 4381 processors, should keep the IBM mainframe market on an upward path. *Source: Computer Intelligence (Karen Landis — Industry Analyst), La Jolla, CA.*

DB2 Acceptance Higher In Canada Than U.S.

In general, the Canadian IBM mainframe software market mirrors the U.S. marketplace. However, there are always slight differences between the two markets in the distribution and acceptance of various software products. One example of this is the acceptance and deployment of DB2, which has become one of the hottest software products in the market even though it only runs in the MVS environment. In the U.S., the MVS site base represents 44 percent of the total IBM/PCM mainframe base while in Canada, MVS stands at 48 percent. Currently, 29 percent of the Canadian sites are running DB2 on one or more of their MVS systems whereas just 20 percent of the U.S. shops are doing this. Within the Canadian DB2 site base, 90 percent of them are using other DBMS products in conjunction with DB2 and in the U.S. only 81 percent of the DB2 sites were found to be using additional database software. Not surprisingly, the most commonly found DBMS being used alongside DB2 is IBM's IMS in both the U.S. (55 percent) and Canada (62 percent). *Source: Computer Intelligence (Jerry Berry — Industry Analyst), La Jolla, CA.*

AFCOM To Sponsor Automated Operations Symposium

AFCOM, the Association For Computer Operations Management, is sponsoring a two-day educational symposium dealing exclusively with data center automation. Automation consultants Arnold Farber and Rosemary LaChance of Farber/LaChance, Inc. are organizing the program and will be on hand to present their ideas and views. The seminar, covering a total of 18 automation topics, will be held September 11-12, 1989 in Kansas City, MO. Registration is $395 for AFCOM members and $455 for non-members. For information, call AFCOM headquarters at (714) 997-7966.

Disaster Recovery Symposium and Exhibition

The first Disaster Recovery Symposium and Exhibition will be held September 11-13, 1989 in Atlanta, GA. Between 300 and 400 attendees are expected along with 100 vendors. Some of the vendors who plan to attend and exhibit their products are: IBM (who has just recently announced its involvement in disaster recovery), Comdisco, Sungard, Hotsite, AT&T and US West Communications. For more information contact Rich Arnold, Disaster Recovery Journal, (314) 846-1001.

Mainframe Channel Extenders

The ability to immediately deploy local devices remotely with performance comparable to that of locally attached devices has created wide interest in the application of channel extension.

By Andres Llana, Jr.

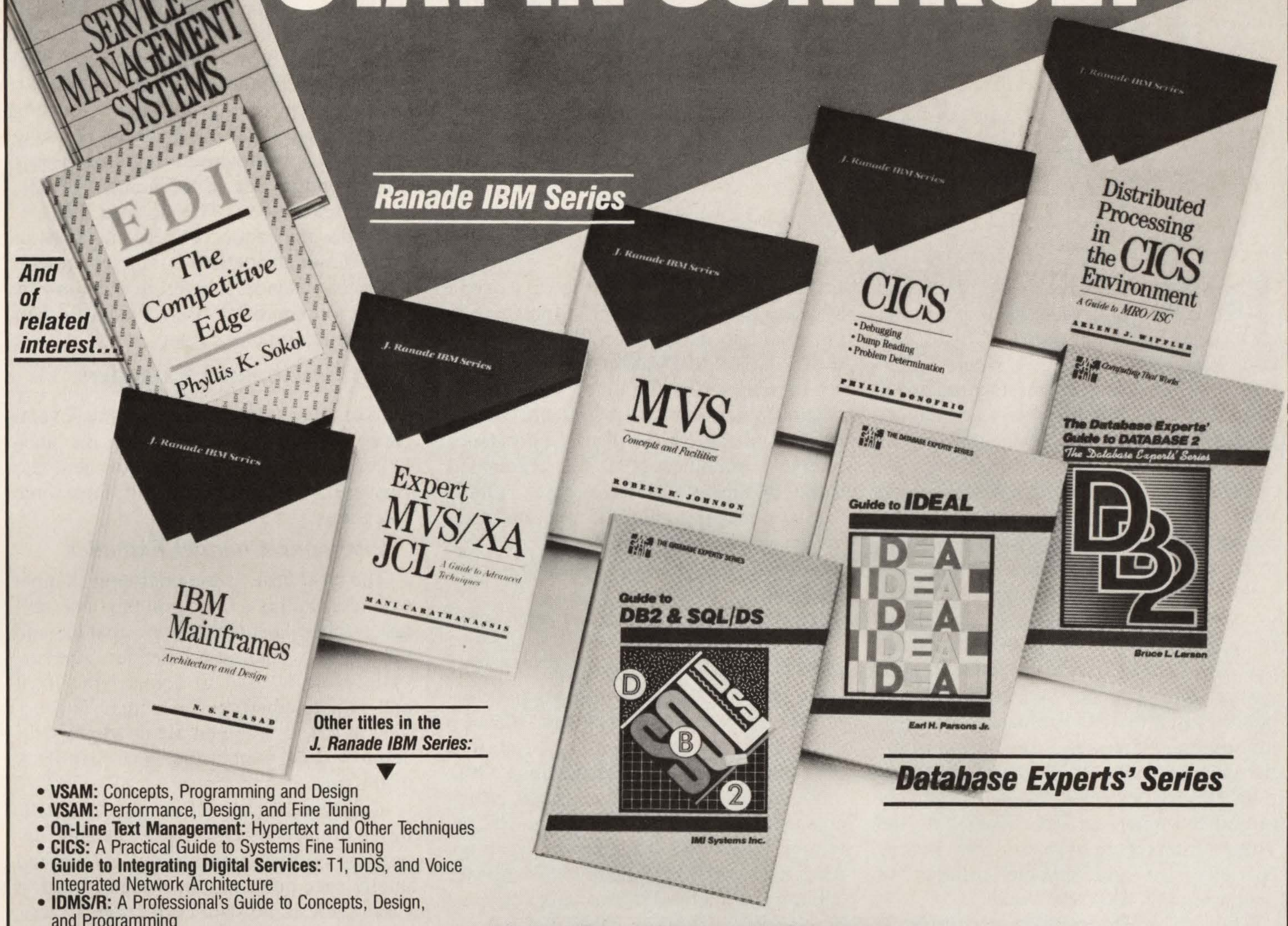
The operational expense associated with multiple large scale computer facilities should be reviewed on a scheduled basis as it relates to a company's profitability. Very often the expense of multiple computer facilities comes about as the result of some corporate acquisition. This usually leaves the company with more computer power than is required. Faced with such a situation, a company may consolidate its computer needs into a single large scale computing facility to support all of its various operations rather than maintain several separate computer centers. However, to accomplish such a consolidation successfully, a company must plan for the ongoing support of high performance capabilities at those sites formerly supported by the phased-out computer facility.

Consider the case of a San Francisco-based bank and its acquisition of a Sacramento bank. Faced with the cost of maintaining multiple computer centers, the bank had to consider options for trimming costs. The Sacramento bank's check processing operations as well as other critical operations had to be carried on during the consolidation. In order to avoid the damaging effects that relocation of such complicated service functions might bring about, a comprehensive solution was sought by bank management.

The most cost-effective solution to this problem lies in the application of channel extension. This technique has gained wide acceptance in recent years since it provides a way to provision the data processing *speed* required to sustain company operations that are remote to the main computing facility.

The application of channel extension has received additional encouragement through other commercial developments. For example, the competition among carriers has fostered a beneficial decline in the cost of high-speed facilities, making the application of 56 KBPS and T-1 (1.554 MBPS) communications lines rather commonplace. The ready availability of such high speed carriers has provided the avenue by which companies can communicate at even greater speeds than before.

STAY IN CONTROL!



Ranade IBM Series

And of related interest...

Other titles in the *J. Ranade IBM Series:*

Database Experts' Series

- **VSAM:** Concepts, Programming and Design
- **VSAM:** Performance, Design, and Fine Tuning
- **On-Line Text Management:** Hypertext and Other Techniques
- **CICS:** A Practical Guide to Systems Fine Tuning
- **Guide to Integrating Digital Services:** T1, DDS, and Voice Integrated Network Architecture
- **IDMS/R:** A Professional's Guide to Concepts, Design, and Programming
- **ISPF:** The Strategic Dialog Manager

AVAILABLE NOW AT THESE FINE STORES

ALABAMA

Walden Software
Madison Square Mall
5901 University Drive
Huntsville, AL 35805
(205) 721-0899

CALIFORNIA

Scholars' Book Store
327 Richmond Street
El Segundo, CA 90245
(213) 322-3161

OPAMP Technical Books
1033 N. Sycamore
Los Angeles, CA 90038
(213) 464-4322

Stacey's Bookstore
219 University Avenue
Palo Alto, CA 94301
(415) 326-0681

Stanford Bookstore Palo Alto
135 University Avenue
Palo Alto, CA 94301
(415) 327-3680

Stacey's Bookstore
581 Market Street
San Francisco, CA 94105
FAX (415) 777-5017
(415) 421-4687

Computer Literacy Bookshop
2590 N. First Street
San Jose, CA 95131
(408) 435-1118

Stanford Bookstore
Stanford, CA 94305
(415) 329-1217

Computer Literacy Bookshop
520 Lawrence Expressway
Sunnyvale, CA 94086
(408) 730-9955

COLORADO

Tattered Cover Book Store
2955 East First Avenue
Denver, CO 80206
(800) 833-5327
(303) 322-7727

United TechBook Company
249 Main Street
Longmont, CO 80501
(800) 247-4808
(303) 651-3184

DISTRICT OF COLUMBIA

Reiter's Scientific and Professional Bookstore
2021 K Street N.W.
Washington, DC 20006
(202) 223-3327

Sidney Kramer Books
1825 I Street N.W.
Washington, DC 20006
(202) 293-2685

FLORIDA

Walden Software
Merritt Square Mall
777 E. Merritt Island Causeway
Merritt Island, FL 32952
(407) 454-9004

Bookstop
7710 N. Kendall, Suite B1
Miami, FL 33156
(305) 598-7292

Walden Software
Palm Beach Mall
1801 Palm Beach Lakes Boulevard
West Palm Beach, FL 33401
(407) 686-4564

GEORGIA

Engineers Bookstore
120 North Avenue N.W.
Atlanta, GA 30313
(404) 892-1669

Oxford Bookstore
2345 Peachtree Road N.E.
Atlanta, GA 30305
(404) 262-3333

ILLINOIS

Kroch's & Brentano's
29 S. Wabash Avenue
Chicago, IL 60603
In Chicago (312) 332-7500
Outside Chicago (800) 833-BOOK

KANSAS

Bookmark, Inc.
14643 W. 95th Street
Lenexa, KS 66215
(800) 642-1288
(913) 894-1288

MARYLAND

Maryland Book Exchange
4500 College Avenue
College Park, MD 20740
(301) 927-2510

MASSACHUSETTS

Boston University Bookstore
660 Beacon Street
Boston, MA 02215
(617) 236-7415

Softpro
112 Burlington Mall Road
Burlington, MA 01803
(617) 273-2919

M.I.T. COOP at Kendall Square
3 Cambridge Center
Cambridge, MA 02142
FAX (617) 621-0856
(617) 491-4230

Quantum Books
1 Kendall Square—Building 400
Cambridge, MA 02139
(617) 494-5042

NEW JERSEY

McGraw-Hill Bookstore
Hightstown-Princeton Road S-1
Hightstown, NJ 08520
(609) 426-5749

NEW YORK

Walden Software
Hudson Valley Mall
1300 Ulster Avenue
Kingston, NY 12401
(914) 336-7121

Classic Bookshops
1212 Avenue of the Americas
New York, NY 10020
(212) 221-2252

Classic Bookshops
133 World Trade Center Concourse
New York, NY 10048
(212) 466-0668

McGraw-Hill Bookstore
1221 Avenue of the Americas
New York, NY 10020
(212) 512-4100

Waldenbooks
57 Broadway
New York, NY 10006
(212) 269-1139

Walden Software
Poughkeepsie Galleria
Route 9 and Cottam Hill Road
Poughkeepsie, NY 12601
(914) 298-9031

NORTH CAROLINA

Intimate Book Shops
119 East Franklin
Chapel Hill, NC 27514
(919) 929-0411

OHIO

Walden Software
Rolling Acres Mall
2400 Romig Road
Akron, OH 44320
(216) 745-4476

OREGON

Powell's Technical Bookstore
32 N.W. Eleventh Street
Portland, OR 97209
(800) 225-6911
(503) 228-3906

TEXAS

Bookstop
6406 N. Interregional Highway, #1400
Austin, TX 78752
(512) 453-7297

Bookstop
820 Preston Forest Shopping Center
Dallas, TX 75230
(214) 363-5744

Brown Book Shop
1715 San Jacinto
Houston, TX 77002
FAX (713) 652-1514
(713) 652-3937

Bookstop
2922 South Shepherd
Houston, TX 77098
(713) 529-2345

Bookstop
9985 I-10 West
San Antonio, TX 78230
(512) 697-0588

WASHINGTON

University Bookstore
4326 University Way N.E.
Seattle, WA 98105
(206) 634-3400

Your Computer Series Publisher

McGraw-Hill Publishing Company
Professional & Reference Division

CIRCLE #159 on Reader Service Card ▲

Fiber Optic Channel Extender Selected

By Mike Russo

Possibly you have heard the mind boggler, "If you are traveling in your car at the speed of light, will your headlights work?" A similar question for many data processing organizations today is, "If you need to extend an IBM 370 channel past 400 feet using fiber optics, will your devices work?"

The 400-foot cabling restriction of the IBM 370 channel is a barrier, especially for data processing organizations needing to cutover to a new computer room, distribute communications equipment closer to carrier demarcation, distribute local terminal controllers in distant locations to cut leased line costs and enhance throughput or even channel connecting printer pools closer to their primary users. Solutions for overcoming the 400-foot barrier must be both cost-effective and of high reliability. Fiber optic channel extenders provided us here at The Sisters of The Third Order of St. Francis with both a cost-effective and reliable solution to overcome the 400-foot "wall."

The Third Order is a not-for-profit health care corporation that owns and operates seven hospitals and one continuation care and nursing home center in the states of Illinois, Iowa and Michigan. For the past 18 months, the new corporate MIS division has been converting from shared systems to a centralized data center with an Amdahl 5890 in Peoria, IL.

The new centralized system is 2,100 feet from the old computer room and anyone who has experienced a cutover process to a new computer room more than 400 feet from the old computer room realizes the challenge and trauma associated with such an opportunity. We needed a short-term cutover solution to eliminate redundant hardware costs as well as a long-term solution allowing us to centrally distribute seventeen 3174s for our largest CICS user base, the St. Francis Medical Center campus. The 3174 distances were as far as 3,000 feet away.

Fiber optic channel extenders presented the best possible solution. Since we connected the new corporate office to a phone system on the St. Francis campus, fiber optic and its required conduit were already part of the cabling requirements. We decided to use the fiber optic channel extenders to connect to the medical center 3174s and the 3725 which controls our remote communications. This eliminated some duplicate hardware costs, duplicate system programming time and duplicate operational testing time. For the long-term, this gave us the flexibility we required.

To really understand how a fiber optic channel extender works and what its limitations are, it is also necessary to understand how the IBM 370 channel architecture works. An IBM 370 channel like any other electronic interface must follow a protocol to communicate between CPU and control unit. The two current protocols are Data Streaming (DS) and DC Interlock (DCI). DS is the newer, more effective protocol. Physically the channel cabling is composed of two bulky copper-based cables, the bus cable and the tag cable, commonly called the bus and tag.

The tag maintains the actual channel synchronization signalling with the CPU while the bus carries the data. Channel programs, including the channel control words, are considered data and use the bus. Nothing you can program from MVS or VSE goes over a tag cable. Data can be transferred over the bus in an interleaved byte method or an interleaved block method; hence, the terms byte multiplexor and block multiplexor channel. Unless you are running a Partitioned Emulation Program or have special device requirements, block multiplexor is the obvious choice. Byte mode or block mode have nothing to do with the DS or DCI protocol, probably one of the biggest misconceptions people have.

The hardest thing to overcome when first using a fiber optic channel extender is the fright associated with it. Combined, a bus cable and a tag cable have the diameter of a baseball bat, whereas a fiber

In addition, many corporate conglomerates have, as the result of their acquisitions, established multiple data centers which in turn have created a need for high speed inter-computer communications.

Since more companies are relocating their operations as a means of reducing corporate housing costs, there has been a corresponding rise in the number of data center relocations. This in turn has fostered the relocation of large numbers of users remote from their computer centers. Channel extenders offer a medium by which to keep these users linked to their relocated computer centers.

Types of Channel Extenders

Like other technology there are several types of channel extenders on the market, each with its own unique benefits. However, these generally fit into three categories.

Fiberoptic Channel Extender

The most basic is the fiberoptic channel extender. In this application the data travels from the host through the channel and through the fiberoptic extender. A fiberoptic cable is used to connect the local and remote fiberoptic extender. The device at the remote end acknowledges receipt of data through the same path back to the host.

This type of channel extender technology is limited to short distances and, depending on the manufacturer, has limited intelligence built into the system. In most cases there is no data correction or verification of data; the fiber extender simply serves as a conduit for the passing of data. In this process, large amounts of bandwidth are used to pass data with no attempt to manage the bandwidth. Fiberoptic channel extenders are only viable within the confines of a building or campus environment.

Wide Area Extender

Another category of channel extender is the wide area extender. Unlike fiberoptic channel extenders, these extenders have a higher degree of intelligence. A major disadvantage of this type of device is that it usually requires special software that must be linked to and resident with the host operating system. This special host resident software is required to interface with the software resident in the extender units. The software controls the transmission of data while ensuring the integrity of the data as it transits the network. These units by virtue of their software design encounter propagation delays as the deployed distances of the extender units increase. For example, as the distance from the host approaches 300 miles, performance decreases due to the end-to-end acknowledgement factor; therefore, distance becomes a governing factor.

Network Systems Corp. (Minneapolis, MN) manufactures the HYPERCHANNEL RDS (Remote Device System). This system is considered to be an example of a wide area extender for support of devices in remote locations from an IBM CPU. They utilize a comprehensive software system to manage and coordinate the passage of data from the host channel to the controller at the distant end. The RDS models provide for the transmission of data at speeds of up to T-1 (1.554 MBPS). The RDS device may require additional bandwidth if the operating range is extended. This system can be considered as being distance sensitive.

Channel/Device Emulation Extender

The third type of channel extender employs a channel/device emulation that functions as an intelligent networking unit. This design concept frees the host channel to immediately perform other tasks. The data is then buffered in the local extender and forwarded on to the remote extender where it is buffered again. The remote extender transfers the data to the actual device controller and interacts with the device as if it were the host channel.

In the emulation design concept, the channel extender software executes specific device protocols and error recovery routines. In this way both the host channel and the communications facility can operate at their maximum potential.

AT&T Paradyne (Largo, FL) markets the PIXNET-XL, a channel/device emulation type of system. The PIXNET-XL is a multiple processor system that emulates the multiplexer channel at the remote end allowing the attached controllers to act as if they were attached directly to the byte or block multiplexer channel at the computer center end. It can function with multiple-line configurations including a mix of T-1, digital, satellite or analog lines. Since it does not require any host resident software, the PIXNET-XL can be used to extend the multiplexer channels of any IBM or plug-compatible mainframe. The PIXNET-XL also features Multi-link Protocol (MLP) that allows the user to group transmission lines thus obtaining the aggregate bandwidth. If one line fails, the transmission group can continue to function but at lower speed. If the entire transmission group fails, then the PIXNET-XL can dynamically switch devices to an alternate route, such as dial-up lines or backup digital service.

IBM manufactures a channel extender, the IBM 3737 which is a host-to-host channel connector. IBM does not offer channel extension for remoting devices. Such a device would in all probability be counter to IBM's marketing strategy.

Applying Channel Extension Technology

Channel extension provides the capability to extend the byte multiplexer channel or the block multiplexer channel of an IBM (or compatible mainframe computer) to a location that is distant from the mainframe computer facility. Multiplexer channels on the host computer serve as

CIRCLE #3 on Reader Service Card ▲

optic cable is about the diameter of a toothpick.

This is why comparing electrical signals through copper to light signals through glass is not a baseball bat and toothpick comparison. It is a magnitude of difference of media as was encountered in going from vacuum tubes to the microchip. Light can go off and on pretty quick!

For our needs at The Third Order, the fiber optic channel extenders we investigated were the IBM 3044, the AT&T 2740 and the Data Switch 904x Series.

After careful evaluation, we selected the Data Switch 9044 model. The three channels we are extending are meeting our throughput needs and have been 100 percent reliable. We are now planning to extend two more channels. They helped make our cutover a great success.

What are the advantages of fiber optic channel extenders? Obviously, the ability to operate "locally" beyond the IBM 370 channel 400-foot wall while providing the flexibility of placing equipment and even distributing the computer room instead of expanding are distinct advantages. They also can significantly cut leased line and 37x5 costs if you have users you can locally attach within seven kilometers.

So what is the answer to the question, "If you need to extend an IBM 370 channel past 400 feet using fiber optics, will your devices work?" The answer we found at The Third Order is a resounding yes. And we are even using our headlights to do it.

Mike Russo, Manager of Technical Services, The Sisters of The Third Order of St. Francis, Peoria, IL

FIGURE 1

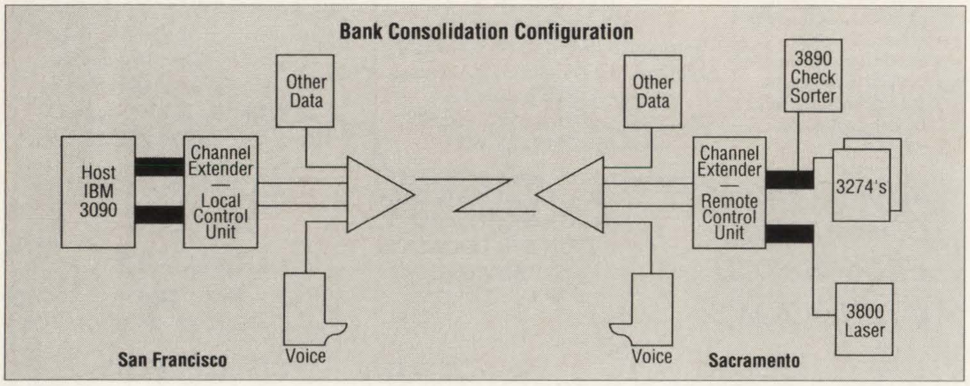
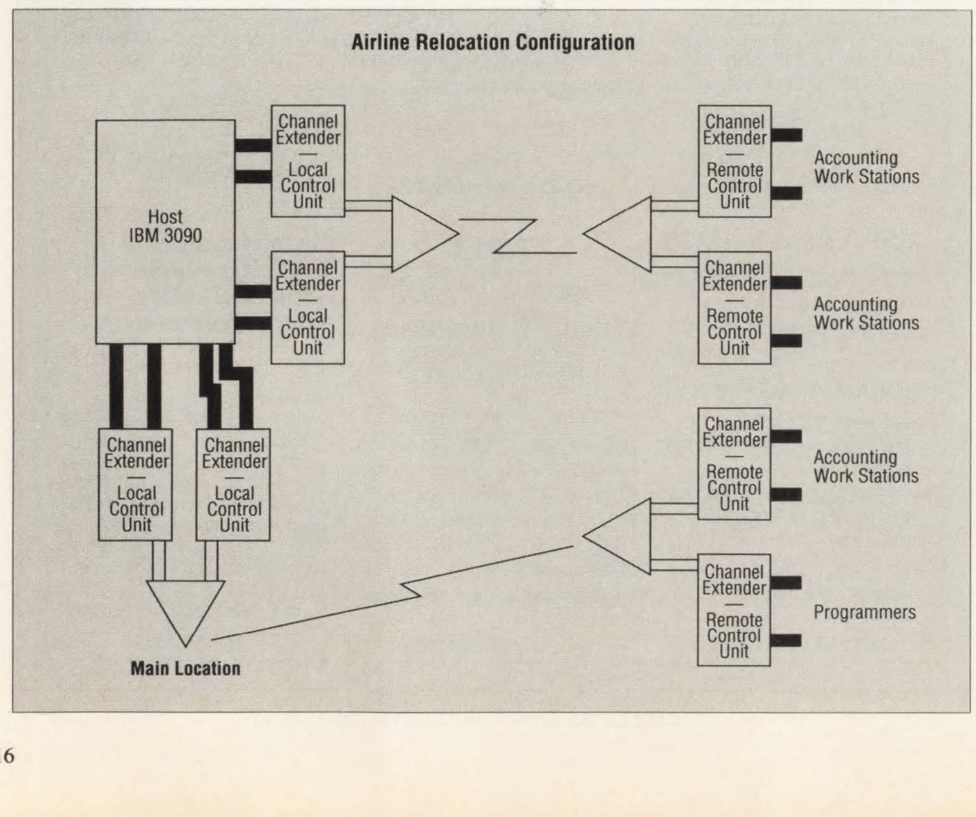


FIGURE 2



the basic attachment interface for various control units supporting terminals, printers, tape drives and other peripheral devices. The problem arises when a controller must be moved beyond the confines of the computer room. Since the electronic characteristics of the multiplexer channel place limits on the distance that a control unit can be located away from the multiplexer channel (greater than 200 feet), most channel attached devices cannot be supported if this distance parameter is exceeded. Therefore, depending upon the devices supported, an additional CPU or special remote hardware and software may be required. Channel extenders offer an attractive alternative to the costs of additional hardware and software, as a means of extending the CPU multiplexer channel to the remote user locations. The ability to immediately deploy local devices remotely with performance comparable to that of locally attached devices has created wide interest in the application of channel extension.

The most common application of channel extension can be seen in the example of the bank discussed previously and illustrated in Figure 1. In this scenario the bank was able to quickly implement its plan for computer center consolidation while leaving intact those operations that were heavily dependent upon mainframe access. In the case of the bank, a channel extender was used to recreate the CPU multiplexer channels at the remote location in Sacramento. This arrangement allowed the user to maintain the controllers at the Sacramento location which supported the complement of printers (3800), check processors (3890), CRT work stations and other devices necessary to the support of the Sacramento operation.

In another application, a large airline was faced with the problem of moving people due to a space constraint (Figure 2). More than 250 programmers and 1,000 accounting personnel were involved. Initially front end processors were considered as one solution to this problem; however, projected terminal response time would not have been acceptable. A channel extender was selected as the solution since it would provide *sub-second* response time and would result in real dollar cost savings.

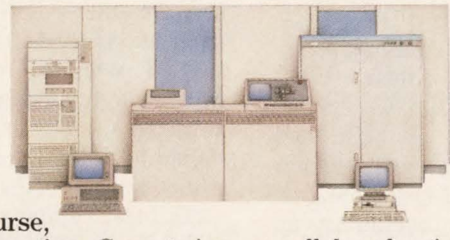
Figure 3 shows the cost associated with a 30-second response time projected over one week and a year respectively.

It can be seen from the example above the cost of a 30-second response time can cost $129.60 per programmer per week or $6,480 per year in lost productivity. A

Our laser printers give new meaning to "sharing"

Until now, sharing has been pretty one-sided: several workstations had to share one laser printer. But Printer Systems has redefined the idea of sharing. Now a laser printer can share across several dimensions.

Different platforms

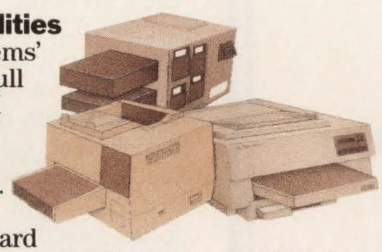


Printer Systems laser printers can be shared by several different kinds of computers. Like IBM. And DEC. And, of course, PCs. All at the same time. Coax, twinax, parallel, and serial connector ports are provided on all our laser printers.

Different applications

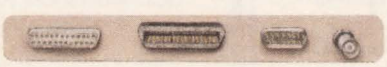
And you can get some pretty fancy things out of these different computers, without a lot of programming. Like barcodes. And FlexFonts from BitStream, with access to over 2,000 fonts. So now your mainframe can do what you thought you needed a PC for.

Different personalities



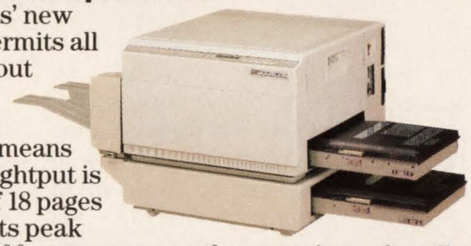
With Printer Systems' new laser printers, full emulation of the IBM 3812, the DEC LN03 Plus, and the HP LaserJet II is a given. In fact, our printers not only come standard with all the same fonts as IBM's, DEC's, and HP's laser printers, but our font matrices are precisely identical to theirs. So ours print exactly the same as theirs.

Real compatibility



Complete compatibility is a given, too, right down to the resolution. Like the IBM 3812, you can have 240 dpi for IBM applications. Or 300 dpi like the DEC LN03 Plus and the HP LaserJet II. In the same machine. Switched–not scaled–by the machine itself depending upon the application.

Introducing the Intelliprint 218®

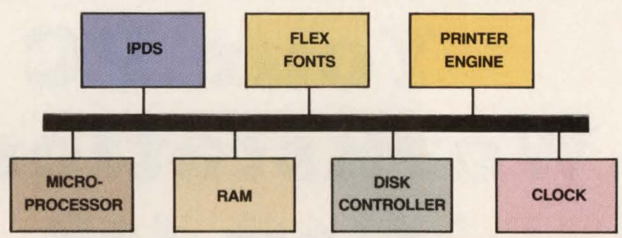


Printer Systems' new Intelliprint 218 permits all this sharing without giving up performance. Its 32-bit RISC technology means that its real throughtput is the rated speed of 18 pages per minute. And its peak duty cycle of 50,000 pages per month means it can handle however much you demand of it.

And the Intelliprint 106®

Sometimes, you can have too much of a good thing. So Printer Systems developed the Intelliprint 106. It, too, is based on 32-bit RISC technology, so its 6 page per minute rated speed is its real throughput. And its duty cycle of 3,000 pages per month is ideal for lower-demand situations.

Open architecture on a PC bus



Printer Systems has also redefined "sharing" by putting a PC AT inside the Intelliprint 218. That provides all the functionality of a dedicated workstation. And, like any other workstation, it affords the flexibility to add options, like a LAN interface, for instance.

Wait, there's more

In fact, there's a lot more. Printer Systems has more up its sleeve, like spooling. And real-time, automatic switching between interfaces. And different graphics emulations. But we can't tell you about all of it yet. At least not here. Why not give us a call at 1-800-638-4041? We'd be delighted to fill you in. So you can start sharing the way sharing was meant to be.

PSC

Compatible in every way

Printer Systems Corporation, Corporate Headquarters: 9055 Comprint Court, Gaithersburg, Maryland 20877
(301) 258-5060, (800) 638-4041, Fax: (301) 926-7333, Telex II: 710 828 9621

CIRCLE #50 on Reader Service Card ▲

F I G U R E 3

Response Time Cost @ 30 Seconds

	$/sec	$/hr	$/wk	$/yr
Programmer @ $20/hr	.0054	@ 20 Trans/hr 3.24	129.60	6,480
Accountant @ $15/hr	.00416	@ 20 Trans/hr 2.50	100.00	5,000

similar loss in productivity can also be established for the accounting operations. Therefore, if a sub-second response time can be achieved, the costs associated with slow response time can be avoided. In this example, real productivity gains can be achieved since the relocated airline operations are heavily engaged in remote transaction processing activities where sub-second response time can affect large numbers of personnel.

Other User Experiences

Disaster recovery is one of the major concerns of every company particularly since the Hinsdale, IL fire that wiped out the communications links between many users and their primary computer facility. Disasters can also occur whenever a point-to-point circuit fails and the user has no alternative route to the computer center.

HOTSITE, the disaster recovery service of CompuSource, Inc. (Cary, NC) utilizes the AT&T Paradyne PIXNET-XL channel extender to provide a unique recovery service for some of their clients. They can air ship a PIXNET-XL channel extender to any location in the United States and have a client on-line in a matter of hours. The client in a disaster situation can bring his computer operations up in one of three HOTSITE locations in the eastern United States. As the channel extender units are being installed at the remote locations, the backup computer facility can be brought on-line. A well-prepared user can in many instances resume operations in a matter of hours utilizing these relocatable channel extenders.

Selecting a Channel Extender

There are several manufacturers of channel extenders. However, the network manager should consider the following factors when considering any system. First, it is important that the system being considered is able to achieve channel-like performance with minimal bandwidth and be capable of managing efficiently all of the available bandwidth. Further, to be an effective device, the channel extender must be able to utilize all types of communications media (that is, analog, digital, T-1, satellite, microwave and so on) with the ability to handle multiple circuits at mixed speeds. In addition, the device should be able to use standard analog facilities for dial backup support.

An ideal system should be insensitive to distance with the capability of channel-like performance over long distance without visible degradation to network operations. Such a system should also be capable of being independent of the host operating system with no host resident software being required.

The channel extender software should support extensive device and link error recovery with sufficient memory buffers to support device performance, error recovery and so on. There should be sufficient on-board hardware diagnostics to support the tracing of data and other related errors at the link, session and device level. Further, the system should be easy to install with on-board link and device testing. The system should also support required devices, CPU-to-CPU transfers and networking. Obviously the vendor selected should be capable of providing full levels of support with a guaranteed response time.

If the user follows the selection criteria as stated above, he will be positioned to deal with most of the problems associated with provisioning high speed data links to support system and user requirements.

ABOUT THE AUTHOR

Andres Llana, Jr. is a nationally known writer on telecommunications topics. He is presently a consultant with Vermont Studies Group, Inc. where his clients include private industry and public institutions. Llana's technical and management career spans more than 30 years. He is a certified systems professional. VSG, Inc., PO Box 372, West Dover, VT 05356, (215) 623-4137.

Is Anybody Out There Listening?

Understanding MVS' Event Notification Facility (ENF)

By Bruce Bordonaro

The Event Notification Facility (ENF) is a little known and even less understood component of MVS which, as the name implies, notifies interested MVS components of the occurrence of particular IBM-defined system events. It came about as an alternative to the usual MVS WAIT/POST mechanism, providing much better performance while also eliminating dependence on the signaller needing to know ahead of time which components it would be signalling. This is unlike the WAIT/POST mechanism in which the POSTer must know of the existence of all tasks WAITing. ENF was first introduced in MVS/SP V1.1.0 and remains relatively the same through all further MVS releases. The exception is that, as new releases of MVS have been introduced, IBM has also introduced new events that ENF could signal.

The routines that detect a particular event and send out the notification of it are called *signallers*. The routines that receive notification are called *listeners*. Both types of routines identify themselves to ENF using a branch entry mechanism and from then on they call or are called by ENF via the same branch entry interface. This is what makes ENF more efficient than the SVC interrupt processing associated with the WAIT/POST mechanism. This also makes ENF caller-independent since, unlike WAIT/POST, the signaller (POSTer) does not need to know the location of each Event Control Block (ECB) that a listener (WAITer) is waiting on; ENF signallers have no knowledge or concern for those components listening for event occurrence.

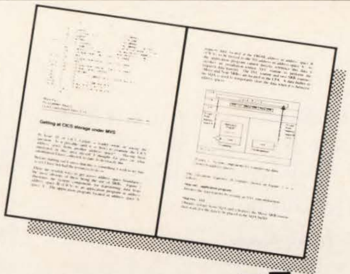
ENF Initialization

ENF is initialized during Nucleus Initialization Processing (NIP) by module IEAVNP47. Input passed to IEAVNP47 includes a pointer to nucleus module IEFENFDM that is a skeleton ENF Control Table (ENFCT) built by the SYSGEN process. Among other things, it is set up with the nucleus address of the ENF interface routine IEFENFIN in the ENCFMOD field, the nucleus address of the ENF service routine IEFENFFX in the ENFCRMOD field and the maximum number of events that ENF will support in the ENFCMAX field. This table is 44 bytes long and is pictured along with other ENF-related control blocks in Figure 1.

During initialization processing, IEAVNP47 builds and initializes an ENF Vector Table (ENFVT) and ENF Process Table (ENFDS) and points to each from the ENFCT (the ENFCVT and ENFCDS fields, respectively). It then loads the address of the ENF internal processing routine IEFENFNM within PLPA and stores it in the ENFCPMOD of the ENFCT. Finally, it turns on a bit flag in the ENFCFLGS field indicating that ENF initialization is complete.

The ENFVT is built in subpool 231 key zero storage and is variable length (a four-byte base plus eight bytes for each event code). It is used to point to the chain of listeners for each event code. The ENFDS is built in subpool 239 key zero storage and is 1,804 bytes long. It consists of a

Why 10,000 CICS specialists read *CICS Update* every month:



'*Excellent!* The topics are timely, and the editors do an excellent job of making contributed material readable. We have implemented ideas into our existing CICS programs from a variety of issues.' *Rick Ricciardi and Joseph M Treese, Thomas Jefferson University, Philadelphia PA* 'I enjoy reading it and find many of the articles "mind-expanding". The articles deal with specific problem that occur in the "real world", ou side of IBM manuals.' *C Richar Adams, Western Computer Services Inc, Salt Lake City UT* 'It helps me understand the idiosyncracies of various versions of CICS. It gives me new ideas and confirms things we are already doing right in our shop.' *W H Petty, Department of Corrections, Tallahassee FL* 'Excellent. Best and most useful magazine I know of, related to CICS systems programmers.' *John Ruth and Frith Powell, Nova Corporation, Canada* 'Excellent publication that provides the user with a learning tool as to how CICS/VS works. Outstanding.' *Eugene S Hudders, MCS Inc, Puerto Rico* 'No CICS shop should be without it.' *Wolfgang von Thuelen, Manitoba Data Services, Canada* 'Excellent! Keep up the good work.' *Curt Swindoll, Insight for Living, Fullerton CA* 'Beautifully done!' *J Logisz, CFS Continental Chicago IL* 'Best sourc... ideas availa... ti... ...cutone IL ... the cost.' *Chuck Jo... Eaton Company, Canada* 'Very useful – worth the cost. Lots of useful tips and utilities.' *Brent Holm, Security Pacific Automation, Brea, CA* 'A very good CICS publication.' *Dennis W Stoudt, SPS Technologies, Newtown PA* 'Very good. It fills a gap.' *Tim Baker, Rolls-Royce, Can...* 'More than worth th... *Ben Driv...* ...the ideas ...e my system.' *Bob ...aefer, Dairy Equipment Co, Madison WI* 'Very informative, there's nothing else like it. Used many of the programs.' *Frank Cervone, Globe Glass & Mirror, Chicago IL* 'I think it is a very useful tool for those involved in developing applications and debugging, and for systems programmers.' *Wayne Bluett, Assurance-vie Desjardins, Canada* 'Great! There's something useful in every issue.' *Ray Walko, Jaguar Cars, Leonia NJ* 'Informative, meaningful, helpful ... Information you can get nowhere else.' *Peter W Puff, City of Regina, Canada* 'Informative, handy CICS utilities provided, other CICS users sharing experiences (almost like having international users group). Very ...d! Keep it up!' *John M Banister, ...da National Bank, Jacksonville FL* ...ellent. One of the few com... journals that are both *prac-*...ut not a "surface skimmer". ... up the good work.' *Fritz Sonnichsen, UNC, Chapel Hill NC* 'I find *CICS Update* to be an excellent reference for problems that we may have or experienced. We look forward to each new issue.' *Stephen Klotz, Delta Faucet, Indianapolis IN* 'Very good, useful, and informative.' *John M McDonald, The Continuum Company, Austin TX*

Free bulletin board

Subscribers have access to Xephon's Illinois-based bulletin board, from which they can download articles and code in machine-readable form.

... and 6000 VM specialists read *VM Update:*

'It is one of the few publications that cross my desk that I look forward to, I always take time to review it immediately. It has helped me and my shop.' *Kirk Weiss, Aeronca, Middletown OH* 'We enjoy it very much. It gives us a chance to get a more technical view, learn some programming, and apply some ideas to help our site.' *Alan Mercer and Tom Carlin, Caesars Casino, Atlantic City NJ* 'I've heard nothing but praise in the VM community. I believe you owe your success to the practicality and ease with which your articles can be put to use.' *Alan W Diaz, Pfizer Inc, Groton CT* 'Good publication – I look forward to it each month!' *Wes Scott, National Bank of Detroit, MI* 'A very good and useful publication.' *T Chris Wille, Winn's Stores, San Antonio TX* 'Very good ... Extremely helpful in developing REXX expertise.' *Tom Brophy, Blue Cross Blue Shield of NH, Concord NH* 'Excellent!' *Larry Peters, Syntex, Palo Alto, CA* 'Very good. We have copied many of the programs etc and use them in our daily operations.' *Patrick Manley, Times Mirror Cable TV, Irvine, CA* 'Brief and to the point. Best technical "odds and ends" publication we receive.' *Ralph H Schwartz, Gancom, Harrisburg PA* 'One of the best magazines of any sort to come out. Only rivalled by *CICS Update* and *MVS Update*!' *Kent Hoeg, Uniroyal Goodrich, Canada*

... and 8000 MVS specialists read *MVS Update:*

'Excellent; informative; I open each envelope with anticipation. We use the NETNAME routine from the first issue for the installation of new terminals. We are moving data centers, and it will be invaluable in installing our local terminals.' *Wayne Bell, National General Insurance, St Louis MO* 'Very good. The articles are very useful.' *John Lendman, Ryder Truck Rental, Miami FL* 'Great! Nice, concise articles.' *David M Lichtel, Kay-Bee Toy and Hobby Shops, Pittsfield MA* 'It has proved very helpful and handy. Provides an arena for the open exchange of ideas and techniques between MVS systems programmers.' *Jim Russell, Cooper Industries, Quincy IL* 'Great! Excellent examples and ideas. We've coded and used several utilities, subroutines, and TSO commands.' *Rick Mayer, Air Products & Chemicals, Allentown PA*

Subscription

Send to Xephon/WPWS,
PO Box 4480,
Winter Park, FL 32793.
Telephone (305) 849 4190.

xephon

Please send me 12 consecutive monthly issues of the journal(s) indicated, starting with the latest issue.

☐ *CICS Update*	⩾40 pages per month	*$130.00 per year*
☐ *VM Update*	⩾50 pages per month	*$135.00 per year*
☐ *MVS Update*	⩾40 pages per month	*$175.00 per year*

☐ Please bill me ☐ Check enclosed payable to Xephon

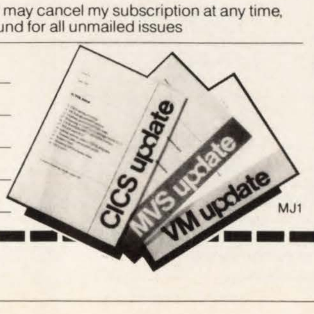
I understand that I may cancel my subscription at any time, receiving a full refund for all unmailed issues

Name ______

Job title/department ______

Organisation ______

Full postal address ______



MJ1

CIRCLE #184 on Reader Service Card ▲

four-byte header and room for up to fifty 36-byte parameter lists used to store information about ENF requests which must be processed asynchronously.

Requesting ENF Services

There are three types of ENF requests that may be issued: (1) identify a listener for an event, (2) delete a listener or (3) signal the occurrence of an event. In order to perform any of these, an ENF Parameter List called the ENFPM must be built (shown in Figure 2) which contains among other things the event code, the type of request (X'0001' = signal, X'0002' = listen and X'0003' = delete listener) as well as a qualifier. The qualifier is used during listening and signalling requests to reduce ENF overhead. Before giving control to a listener, ENF will check if the listener is performing a qualified listen and if so it will match the listener's qualifier with the signaller's qualifier. Only if they match or the listener indicates in his parameter list that qualification is not to be used, is the request passed to the listen exit routine.

Any caller of ENF must be in supervisor state. The caller's ENFPM must indicate whether the request should be processed synchronously or asynchronously. Any caller running disabled or while holding locks must request asynchronous processing to minimize the time in these states. This is done by turning on the ENFPASN bit in the ENFPFLG flag byte that is checked by IEFENFFX along with fields in the Prefixed Save Area (PSA) which indicate processing mode and locks held. For these requests, IEFENFFX copies the ENFPM to a slot in the ENFDS and posts an ECB for an ENF Wait Task called IEFENFWT that resides in the Master Scheduler address space. When this task is dispatched it will process all the requests stored in the ENFDS. For synchronous requests, IEFENFFX branches to the ENF internal processing routine IEFENFNM directly. In both cases, IEFENFNM is the module that performs the actual processing of requests.

To identify a listener, the caller of ENF sets up an ENFPM that must contain the address of a listen exit routine that will get control from ENF when the event is signalled by a signaller. IEFENFNM checks the event code in the ENFPM for validity, and if valid, uses it as an index into the ENFVT. An ENF Listener Element (ENFLS) is built and chained off the appropriate slot in the ENFVT if this is the first listener for the event or off the last ENFLS in the chain for this event via the ENFLPTR field if there are already other listeners identified. A token is returned by IEFENFNM in ENFPTOK which may be used later to delete the listen exit. In MVS/XA, the token turns out to be the 31-bit address of the ENFLS built by ENF for the listen request. I assume that in MVS/370 the token would be the 24-bit address of the ENFLS built. Note that ENF does not allow duplicate listen requests; that is, a listen request will not be honored if the request matches a currently active ENFLS entry. I assume that the match is based on the address of the listen exit routine; however, I have not verified this. In this case, IEFENFNM will set a return code of X'04' and return to the caller.

FIGURE 1

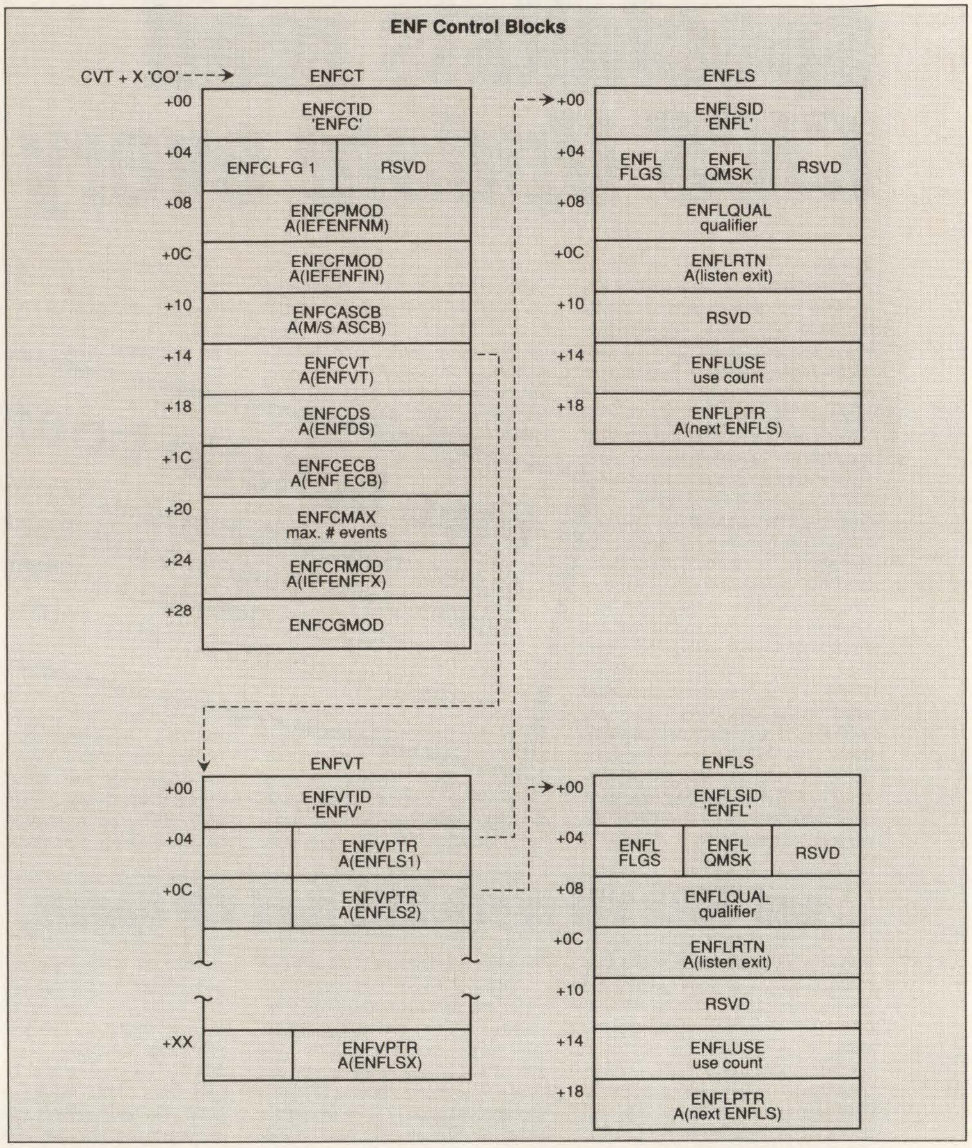
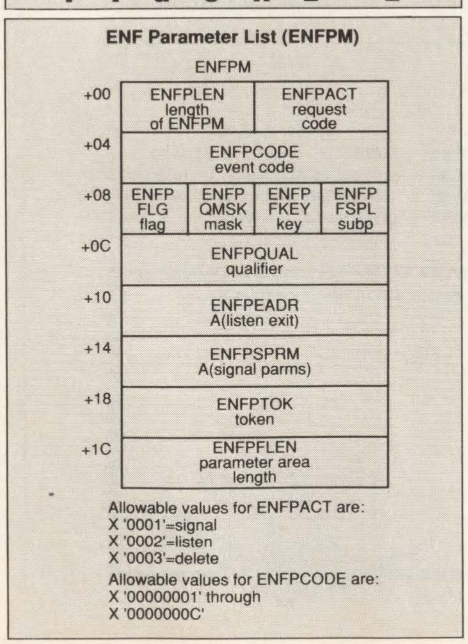


FIGURE 2



To delete a listener, the caller of ENF must also set up an ENFPM that contains the length of the ENFPM, the event code and the token returned when the listen exit was established. IEFENFNM will match the token against currently active listen requests; if there are no matches a return code of X'1C' will be returned. If a match is found, IEFENFNM turns on the high order bit of the ENFLUSE field to indicate that the ENFLS is available for reuse (without the need to free the storage). However, if the use count in the ENFLUSE field is non-zero, indicating that listen exits are currently active, any new signal requests will not invoke a listen exit marked for deletion. When currently active listen exits terminate, the ENFLUSE field will be decremented and when it becomes zero, the ENFLS will only then be actually considered available for reuse.

To signal an event, the caller sets up an ENFPM similar to those described above. IEFENFNM finds each associated ENFLS chained off the ENFVT for the requested event code and gives control to each listen exit routine based on the ENF qualifier as explained earlier. The signaller may also pass the address of an optional parameter list to each listener via the ENFPSPRM field of its ENFPM which may be used for communication between listeners and/or the signaller. When all listen exits have been invoked, an optional signal exit is invoked if the signaller supplied it in the ENFPEADR field of its ENFPM. After the signal exit returns control to ENF, if the signaller turned on the ENFPFREE bit in the ENFPFLG, IEFENFNM will free the parameter area pointed to by the ENFPSPRM field if provided. It does this by retrieving the key of the parameter area, its subpool number and its length in the ENFPKEY, ENFPSPL and ENFPFLEN fields, respectively. These fields would have been filled in by the signaller prior to issuing the signal request.

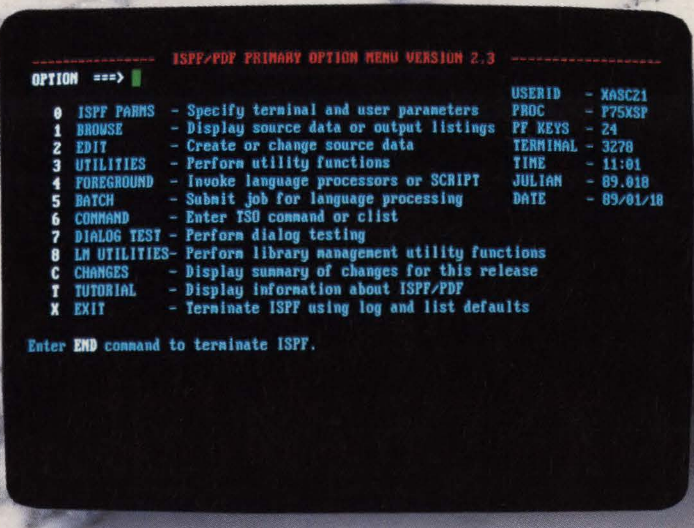
Note that all listen and signal exits as well as parameter lists should be defined in commonly addressable storage. This point should be apparent when considering asynchronous processing. Since the requests are stored in the ENFDS and a task in the Master Scheduler address space is posted to be dispatched to handle these requests, the exit routines and parameter lists, if any, must reside in either (E)SQA or (E)CSA. Alternatively, the exit routines may reside in (E)PLPA, being loaded at IPL time just as the IBM-supplied listen and signal exits are.

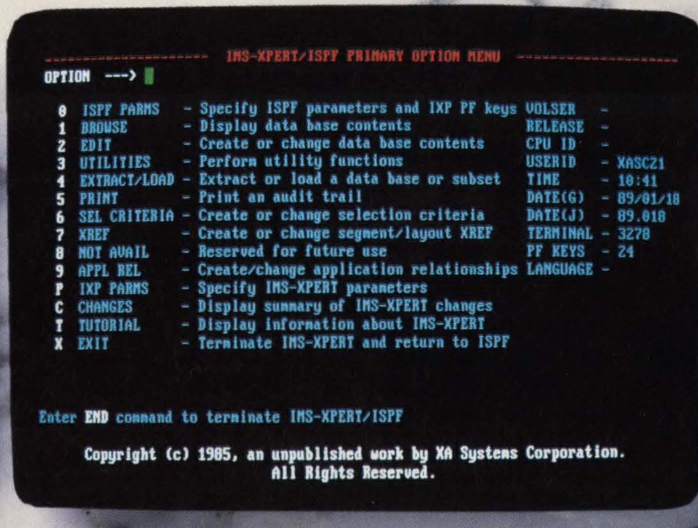
The following is a list of return codes for ENF requests:

00	the request was processed successfully
04*	a duplicate listen request was detected
08	the ENFDS control block is full
0C	an error was found in the caller's ENFPM
10	ENF is not available
14	ENF is not initialized
18*	storage cannot be obtained for the ENFLS
1C*	an invalid token was found for a delete listen request
20*	an error occurred in the signal exit routine for a signal request
2C*	an error occurred when attempting to free a signaller's parameter list.

Those return codes marked with an as-

DejaView.





Productivity has never been so familiar.

It only seemed natural for us to create our XPERT Series of programmer productivity tools with an ISPF-like design. You'll learn skills faster and produce results immediately—something you can never see enough of.

DATA-XPERT, IMS-XPERT and DB2-XPERT all work the way programmers do. They're compatible with all the commonly used programming languages, file organizations and data base management systems in the IBM MVS environments.

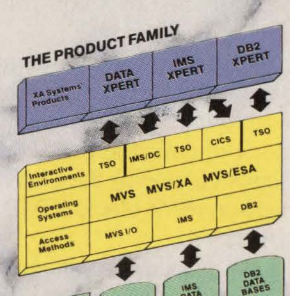
Access to files and data bases like you've never seen.

Without getting tied up by using time-consuming utilities, you can easily access VSAM, ISAM, BDAM, PDS, Sequential files, and IMS and DB2 data bases. Once you're there, you can selectively edit, browse, extract, load, convert, reformat, and print any size file or record to get just the data you need.

See for yourself.

Find out why the XPERT Series is rated #1 over the competition.* Call today for more information or a free 30-day trial of DATA-XPERT, IMS-XPERT or DB2-XPERT.

Call 1-800-344-9223;
in Canada, 1-800-344-9224.



The XPERT Series. Things will never seem the same again.

XA SYSTEMS CORPORATION

CIRCLE #134 on Reader Service Card ▲

*Hancock Survey, January 1989. DATA-XPERT and IMS-XPERT are registered trademarks of XA Systems Corporation. DB2-XPERT is a trademark of XA Systems Corporation. ISPF, IMS and DB2 are products of IBM.

terisk (*) are for synchronous requests only. The conditions described result in a zero return code during asynchronous processing because the conditions have not yet been checked by ENF. For asynchronous requests the ENFPM is stored in the ENFDS and validated later when the IEFENFWT is posted.

ENF Events

As of MVS/SP V2.2.0 there are 14 unique ENF events defined which is indicated in the ENFCMAX field defined at SYSGEN time in IEFENFDM. Currently the System Resource Manager (SRM) listens for vary device, vary path and DDR swap events (codes 7, 8 and 10, respectively). These events are significant to the SRM in terms of workload balancing. Global Resource Serialization (GRS) establishes itself as a listener for event code 5, communications task and TOD clock services which are required to process GRS requests. The Common VTOC Access Facility (CVAF) listens for DASD volume unloads to allow it to invalidate VTOC-related in-core control blocks (VTOC information blocks or VIBs). The definition of each event is as follows:

Event Code	Description
1	Vary on-line of a non-console device
2	Vary off-line of a non-console device
3	Unload of a DASD volume
4	Severe SQA shortage
5	Communications task and TOD clock initialization are complete
6	Status change in the channel measurement block (CMB) data collection
7	Change in device status
8	Change in path status
9	Change in channel path status
10	DDR swap
11	Failure of channel monitoring facility
12	Pending off-line device
13	WTO buffer utilization (MVS/SP V2.2.0)
14	JES3 buffer utilization (MVS/SP V2.2.0).

It is difficult but not impossible for a systems programmer to add space for his own user-defined events. This can be accomplished by an AMASPZAP modification to module IEFENFDM at offset X'20'. The current value at that location is X'0000000C' for MVS/SP V2.1.7 and X'0000000E' for MVS/SP V2.2.0. This can be changed to any other reasonable number which will cause the ENFVT to be built with that many entries at IPL time. The problem associated with this, aside from the obvious problem IBM will have with modification of the code, is that if and when IBM comes out with new event codes, they may conflict with the extra ones added. This may be avoided by expanding the table to a large size and using event codes from the end of the table and working backwards.

This could provide systems programmers with a useful but slightly unstable method of implementing their own events. Some things that you may wish to establish as events or that IBM might consider implementing are WTO buffer shortages, SMF dataset switching, LOGREC dataset filling and SRM/performance-related events (slot shortages, bad page/swap datasets, GRS ring failure and so on). You may notice that some of these events already have standard system exits associated with them. While this is true, these exits must reside either in common storage areas by themselves or as part of other load modules and require loading at IPL time. A systems programmer could code the exit as a signal request, having previously loaded the listen exit into (E)CSA via a started task or batch job. In this way, the listen exit could be easily replaced on-the-fly simply by rerunning the job. The facilities provided by these listen exits could also be easily disabled by deleting the listen exit.

Another more practical approach, and one that would be even more useful, would be for use in automated operations tools. These tools currently rely on the format of the message text to perform their functions. It would be nice to make the components which issue certain messages use ENF instead. This way there would be no dependency on message formats and the pertinent information contained in the message could be passed in the parameter list pointed to by the ENFPM. This would also provide for much more efficient processing of automation events than the current Message Processing Facility (MPF) exits and/or automated operations tools provide. Most automated operations tools get control from the MVS Subsystem Interface (SSI) calls issued as part of WTO and command processing (SVC's 35 and 34, respectively).

Conclusion

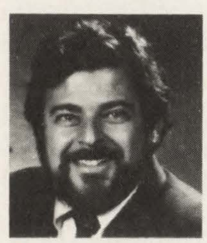
Even though ENF is a relatively obscure facility, I believe that IBM will continue to make more use of it. Its fast branch entry nature makes it more performance-oriented than current SVC-based mechanisms. Its relatively simple interface makes it easy to use. This would be especially true if IBM would publish information on the signallers and the parameter areas they pass to listeners.

Even if you choose not to make use of ENF yourself, become familiar with it now while it is still relatively straightforward and simple to understand. In any event, I will bet there are many people out there who could stand to improve their *listening* skills.

To get started using the facilities of ENF, I have written two sample programs. The first, ENFPGM, establishes a listen exit for event code 1 (or the event of your choice). The second program, ENFLSTEN, is a listen exit which ENFPGM establishes. When entered, ENFLSTEN simply takes an SVC dump and returns to its caller. I wrote this example so that I (or you) could examine the SVC dump with IPCS to locate the parameter list passed to the listen exit. With that information, I (or you) can make some educated guesses as to the content of the parameters passed by the signaller. I do this because IBM cryptically states in the System Logic Library (SLL) manual for ENF that routines using ENF services "have agreed to predefined values for some of the fields in the ENFPM" and that "to determine these values, refer to the module listings of the signallers and listeners of events." This translates to identifying signallers and callers by going through each SLL manual followed by a trip to your microfiche viewer to examine the source code.

Editor's Note: If you would like a copy of the sample programs, ask for "Bruce's Sample Programs" in the comments section of the Reader Service Card and return it to MAINFRAME JOURNAL.

ABOUT THE AUTHOR



Bruce Bordonaro is a systems software manager for Pershing, a division of the brokerage firm Donaldson, Lufkin and Jenrett, New York, NY. He has more than 12 years of experience in the MVS data processing environment, 10 of which has been in systems programming.

New for your MVS or VM environment!

Are you under the gun to create documentation?

HERE'S AN EASY WAY TO BITE THE BULLET.

Everyone knows that their Data Center should document important policies, standards, and procedures. The trouble is, with all the crushing pressure of work, documentation always seems to go on the "back burner." These days, that can land you in hot water. With all the strict regulations and tough compliance requirements you face, it becomes *absolutely necessary to take action* and finally solve the "documentation dilemma."

Luckily, we can make compliance easier than you ever dreamed possible!

Introducing the Information Systems Series.™ High caliber documentation the painless way.

The ***Information Systems Series*** documentation framework is designed to help you create comprehensive policies, standards, procedures, and user support materials for your entire MVS or VM Data Center.

Created by highly experienced data processing experts, the ***Information Systems Series*** is an interlinked set of five manuals and accompanying, corresponding diskettes: *The IS Policy Guide, The Data Center User Guide, Systems and Programming Standards, The Operations Guide,* and *The Project Plan and Implementation Guide.* All you have to do is follow the simple, imbedded instructions. In other words, you *personalize* the documentation using your own IBM PC or compatible – which means you can easily integrate any existing documentation quickly and easily!

An important part of the ***Information Systems Series*** is our unique *Project Plan and Implementation Guide* that leads you effortlessly through the entire documentation process. No wonder many forward-looking government, business, banking, medical, and telecommunications IS Departments have put the power of the ***Information Systems Series*** to work.

FREE Demo Disk!

See for yourself what the ***Information Systems Series*** can do for *you.* Call for your free Demo Disk right now!

Isn't it worth a few moments of your time to prove to yourself that the ***Information Systems Series*** can save you endless hours of work and many thousands of dollars? Don't miss this no-obligation offer. Call right now!

Call toll-free 1-800-777-0970

The Computer Resources Group, Inc.
303 Sacramento Street
San Francisco, CA 94111

CIRCLE #69 on Reader Service Card ▲

CPU Demand And CICS MRO Performance

By Ted C. Keller

Not too many years ago when the Multiple Region Option (MRO) was added to CICS, it was viewed primarily as a way to relieve Virtual Storage Constraint (VSC). At that time, almost all major CICS systems were experiencing some degree of VSC. After the introduction of MVS/XA and a number of features designed to exploit storage above the line, VSC became less of a problem, especially when MRO had been used to split workloads into multiple regions.

Today, as corporations have become more dependent on computers, more work is being done on-line. As a rule, on-line systems are servicing larger segments of most workplaces and doing more than ever before. Not only have the number of CICS applications grown, but also they have become more complex and resource-intensive. And with relief for many of the most severe VSC limitations available, more work can now be done in each CICS region.

As a result of these changes and growth, many CICS systems are being faced with a new limitation: CPU constraint. More processing is required within a region than CICS' architecture will allow. Just as CICS regions were split several years ago to relieve VSC, it is now common to see regions split to address the effects of CPU constraint. Unfortunately, though, the consequences of CPU constraint in CICS are not easily recognized and those affected may be unaware of the nature or extent of these constraints.

CICS Task Dispatching

CPU processing in CICS is different from that of other subsystems such as TSO or IMS. In most other systems, each task or user runs in a separate address space with its own MVS Task Control Block (TCB). The operating system uses a system of pre-emptive interrupts to distribute the use of available processors to the various tasks. As interrupts signal the completion of external events, tasks can be redispatched based on priorities.

Each CICS region functions primarily as a single MVS task with a single TCB. Other MVS tasks exist in each CICS region, but they are used to handle ancillary functions. Only the optional VSAM subtask (VSP) bears any portion of the application workload. All work related directly to the management and processing of transactions is performed by the single MVS task.

CICS maintains a list of internal tasks. When CICS gives control to an internal task, that task will retain control until it issues a CICS command that cannot be satisfied immediately or exceeds the runaway task limit (in which case it will be abended). While a task has control of the processor, no other work can take place in CICS. Even if external events compete for which higher priority tasks were waiting, CICS has no way of interrupting the executing task and must wait until the task allows it to regain control.

Because the CICS application workload functions under a single MVS task, it can utilize no more than a single processor at a time. Even when other processors are available, CICS can dispatch only one task at a time.

CPU Demand

An important concept when analyzing CICS performance is *CPU demand*. CPU demand is the sum of the times CICS is either using or attempting to use a processor. CPU demand includes such things as the time spent waiting for higher priority MVS tasks, the time spent servicing page faults (CICS cannot make use of the CPU while it is waiting for a page fault to be satisfied) and the time spent waiting for CA splits (the entire MVS task will wait while CA splits are being processed; if VSP is being used, VSP will wait instead of the main task). In essence, CPU demand is the amount of time that CICS is attempting to execute instructions and has not voluntarily placed itself in a wait state.

When planning CICS performance or capacity, it is important to use CPU de-

mand, not CPU utilization, since CPU utilization does not reflect the impact of other important factors. For example, if an external monitor showed that a CICS region was utilizing 70 percent of a processor and was experiencing an average of three page-ins per second, CPU demand might be closer to 75 to 80 percent, depending on how long it took to service the page faults. If this same region was running in a uni- or diadic-processor, another five to 10 percent might be attributed to delays associated with waiting for other operating systems tasks. Thus a region using 70 percent of a processor might really be experiencing a CPU demand of 85 to 90 percent.

Once CPU demand exceeds 50 percent, it can begin to have a noticeable effect on CICS performance. From basic queuing theory, you know that the total service time for a resource (including queuing) increases exponentially as the resource becomes busier. Figure 1 demonstrates the relationship between resource utilization and queuing delays. It can be used to estimate the impact of different levels of resource utilization. For example, it can be seen that it will take approximately twice as long to receive the same amount of service when a resource is 50 percent busy as when it is totally idle. Similarly, it will take five times as long at 80 percent busy and 10 times as long at 90 percent.

CPU demand reflects the amount of time that CICS was attempting to schedule the processor and can be treated as a resource for queuing purposes. When CPU demand reaches 80 percent, it should take about five times as long for an application to perform its instruction path length as when CICS is idle. For example, it should take a transaction that uses 40ms of CPU about 200ms to receive that amount of CPU when CPU demand is at 80 percent. It would still use 40ms but would wait a total of about 160ms at various times in its life when it needed to use the processor. Similarly it would take about 400ms to receive the same 40ms when CPU demand was 90 percent.

Since CICS has no way of interrupting tasks, many components of CPU demand can further increase delays. Tasks which perform extensive computations can impede other work. Functions such as matrix manipulations, linear programming or large internal sorts are not appropriate in most CICS environments. Programs which perform large numbers of certain types of CICS services (such as certain LINK, XCTL, READQ TS or LSR-type READ commands which do not require any real I/Os) will similarly dominate the use of the CPU. When a CICS command is executed that can be completed immediately and does not have to wait, CICS will simply return control to the requesting task without performing its dispatching logic in most cases. In effect, as long as a task performs commands which can be serviced immediately, CICS will allow it to continue to process without interruption and not allow other work to be dispatched. It is possible for a task to issue many CICS commands without ever allowing CICS to dispatch other work.

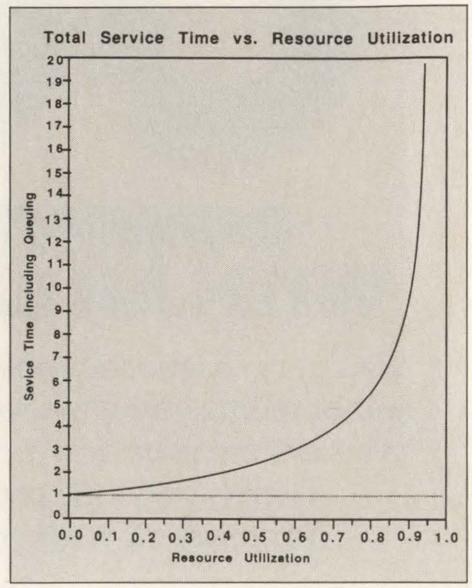
MRO Considerations

In an MRO environment, the effects of high CPU demand have serious performance implications particularly when function shipping is heavily utilized. Whenever resources such as files or temporary storage are accessed via function shipping, high levels of CPU demand can cause delays in servicing these requests.

When a task in one CICS region requests a resource in another CICS region, the requesting region passes the request to the Resource Owning Region (ROR). At that time, if the ROR was waiting, it would be activated and would process the request immediately. However, if the ROR was attempting to use the CPU, the request would not be recognized until the ROR had the opportunity to perform its task dispatching algorithm and dispatch the mirror task. Once the mirror task received control, it could then schedule the I/O request. Assuming that the mirror task then needed to wait for the completion of an I/O event, other work would be dispatched in the ROR. If the ROR was attempting to use the CPU when the I/O event completed, delays would again be introduced until the ROR had the opportunity to dispatch the mirror task. Finally, after the mirror task received control, it would pass information back to the requesting region where similar delays might arise recognizing the completion of the MRO request.

If an ROR is CPU-constrained, delays associated with function shipping can be substantial. CPU demand in regions owning commonly-used resources should be kept to 50 percent or less. When it is much higher than this, queuing delays can be significant whenever other regions attempt to access resources. If CPU de-

FIGURE 1



mand is more than 50 percent but CPU utilization is not, overall system performance can often be improved by increasing the MVS dispatching priority of RORs with respect to other workloads in the system. If there is much paging, it might be beneficial to *fence* the ROR (use storage isolation) to reduce paging delays. (This might not be necessary if most page-faults are satisfied from expanded storage.) In most cases, regions containing resources which are heavily accessed via MRO or ISC should be given a favored status in MVS.

It is often difficult to decide where to place resources which must be used in multiple CICS regions. The decision is often complex and involves determining which regions use the resources most frequently, what the peak CPU demand is in various regions, how much overhead is associated with accessing resources via MRO and whether there are special service requirements which might require the resource to be placed in a particular region.

One commonly used practice is to create one or more File-Owning Regions (FORs) containing only file definitions and shared temporary storage. This configuration not only ensures that CPU demand in the FOR will remain reasonable, but also that other constraints such as virtual storage and max-tasks will not impede access to resources. Resources placed in an FOR can usually be accessed almost as quickly as if they were locally defined in Application Owning Regions (AORs).

The main disadvantage of using an FOR is that all accesses to resources defined in that region will require MRO overhead. Depending on the nature of the request and the timing of accesses, CPU overhead associated with function shipping will typically be as much or more than that actually used servicing the request. The CPU overhead for function shipping is about the same as the CPU used to perform VSAM read and write requests. Function shipping overhead is several times greater than the amount of CPU used for get-next or get-previous requests. Depending on the number of files and type of accesses, the overhead associated with function shipping can be substantial. Heavy use of an FOR can increase overall system CPU utilization significantly.

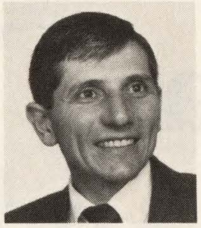
To reduce the overhead associated with function shipping, all requests to an FOR, files and other resources will often be placed in the region accessing them most frequently. The obvious advantage is reduced overhead, but this may be practical only if the level of CPU demand in the ROR remains reasonable. The decision as to where to place resources will usually involve determining which applications are the heaviest users of the resources, how frequently the resources will be accessed by other regions and whether CPU demand in the proposed ROR will constrict access to the resources by other regions.

Conclusions

As the architecture of CICS under MVS/XA has allowed us to overcome many significant constraints, CICS systems have been able to support more transactions and larger, more complex applications. With many other restrictions removed, the next major area of concern seems to be CPU constraint. Because of the way CICS dispatches work, you will begin to experience processor-related constraints long before fully utilizing a processor. These constraints may be magnified in an MRO environment.

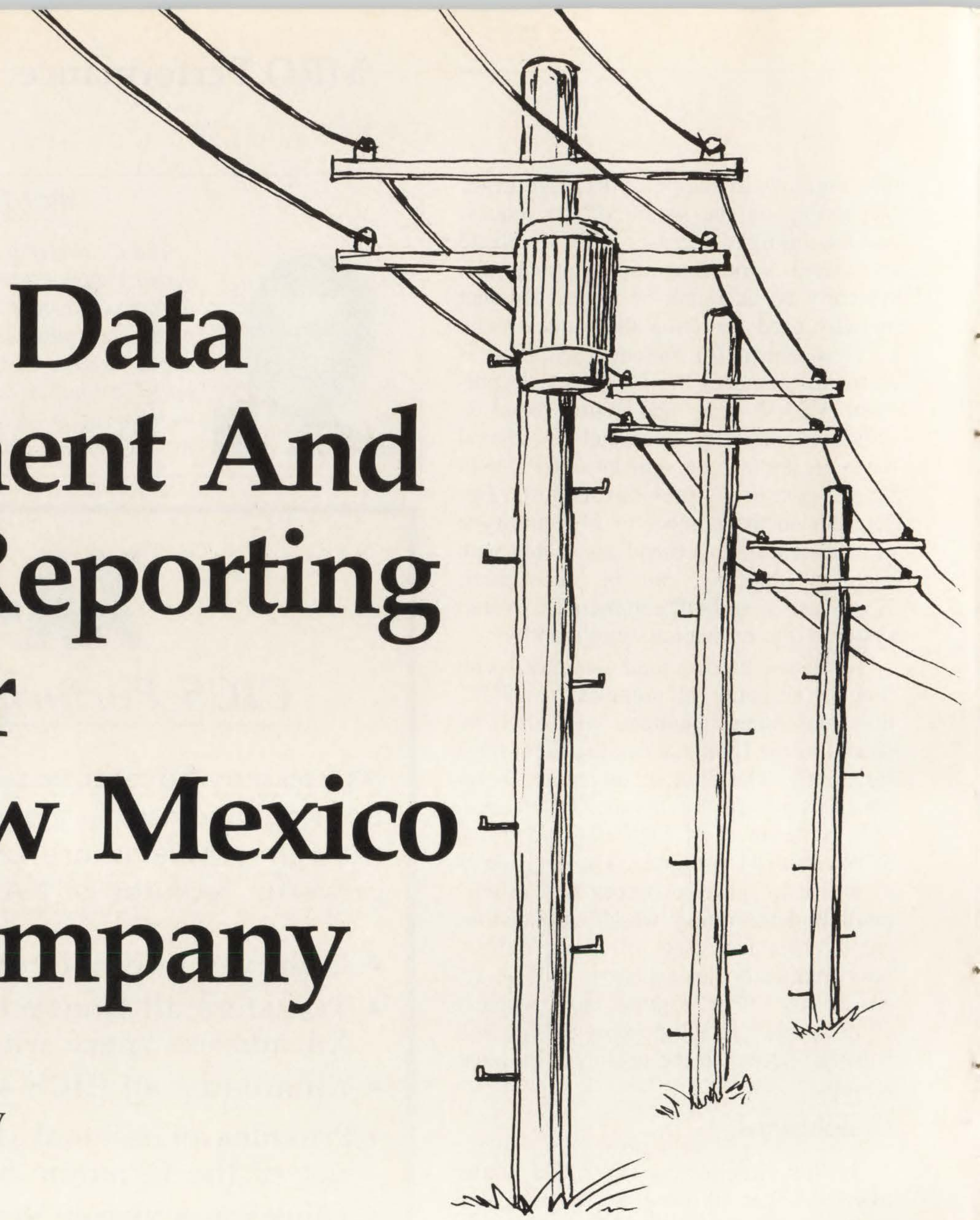
CPU demand can have a noticeable impact on CICS performance. It is important to take action to keep CPU demand at a reasonable level (perhaps below 70 percent) and it is especially important to minimize CPU demand in regions owning resources commonly accessed by other CICS regions. Decisions as to when to split regions or where to place resources are often complex, frequently without any simple, clear-cut answers. Understanding the impacts of CPU demand can help planning systems which will perform consistently.

ABOUT THE AUTHOR



Ted C. Keller is the manager of a group responsible for CICS systems support, performance management and capacity planning at Yellow Freight System, Inc. He has worked in various data processing jobs for more than 21 years. Also, he has been an active member of CMG for the last several years and he is currently serving as co-editor of CMG Transactions and as a board member in the KC regional CMG. Yellow Freight, 10990 Roe Ave., Overland Park, KS 66211, (913) 345-3274.

GFIS Improves Data Management And Graphic Reporting Needs For Texas-New Mexico Power Company



By Cecilia Coburn Perry

"Drive West!" instructed A. P. Barrett who with his Chicago financier, Leo J. Troy, built their utility empire in Texas. When they arrived at a promising-looking little town, they negotiated with the land owner, purchased the property and transferred the land to either the Texas-Louisiana Power Company or one of their varied sister companies. This was the auspicious beginning of the Texas-New Mexico Power Company. Facing bankruptcy during the Great Depression Era, Barrett was forced to sell and the name was changed in 1934 to the Community Public Service Company.

To better identify the company's business, the name was again changed in 1984 to the present, Texas-New Mexico Power Company (TNPC). From the ramblings of its original founders, the utility company now serves 89 communities spanning the northernmost Texas Panhandle to the Gulf of Mexico and west into the White Sands area of New Mexico.

TNPC operates as an electric service company engaged in generation, transmission and distribution of electric power. Annual sales have grown to more than $370 million and the employee base has exceeded 1,000.

To run its massive day-to-day operations and assist in serving approximately 200,000 customers, the Data Processing Operations Department headed by Gary Heinrich houses four IBM 4381 Model 13s. Henrich said that the company will soon acquire the dual processor 4381 Model 14 but has no intention of upgrading to the newer 20 Series.

With a new power plant in central Texas scheduled to begin operations in 1990, the Data Processing Department moved into high gear. In February of this year the Operations Department installed a new DATAMATIC System to facilitate billing and customer relations.

Another important department not only for day-to-day operations, but also for providing information to the "outside world" at TNMP is Graphics. Heinrich reiterated a scenario that puts the importance in layman's terms. "Suppose a worker from the water company drives out to a particular lot with a backhoe to begin laying pipe for a new development. How is he going to know where to dig unless he has a map telling him where other electrical wires or telephone cables are buried? We produce a map that shows the easement, the overhead and underground cables, utility poles — everything!"

All the maps and the reports generated from them are managed by the IBM Geographic Facilities Information System (GFIS). According to a recent IBM newsletter, TNMP's network of distributed GFIS users is probably the largest in the country with eight separate locations.

According to Heinrich, "We will have

New 3270 & Token Ring Support

Remote printing on a PC without a single compromise.

Until now you've had to rely on a S/36 or AS/400® to deliver remote printing. Now a PC with BARR software and adapter sustains print speeds of 6,000 lines-per-minute and line speeds of 128,000 bits-per-second. Only BARR maintains all this performance with the reliability and ease of use PC users expect.

BARR/SNA RJE or BARR/HASP software drives up to five printers from a single PC. What's more, you can enter data, print, and receive output all simultaneously – without interruption. BARR's advanced multi-tasking software easily manages even the most complicated tasks, including LAN access, tape support, file transfer, and special forms printing. In addition, BARR offers one year of friendly, dedicated customer support with each purchase.

Communications adapters and software are available for the IBM PC, PS/2, and compatibles.

Try BARR for 30 days. We've helped thousands save millions. Call 800-BARR-SYS.

Protocols		
SNA RJE with Multiple Sessions		
HASP – BSC Multileaving		
Host Systems		
JES2	DOS/POWER	CDC NOS
JES3	RSCS	VS1/RES

BARR

BARR SYSTEMS INC.
2830 NW 41 Street, Gainesville, FL 32606
800-BARR-SYS, 904-371-3050, FAX: 904-371-3018

CIRCLE #91 on Reader Service Card ▲

13 of these GFIS workstations in our field offices when it is all completed, each with its own IBM 9370. The 9370s are tied into the mainframe through our 3720 communication controllers in Fort Worth, TX.'' Since all users are working from a single source, communication about work in progress is facilitated. The more offices tied into the system, the more productive the environment can be for the entire utility company.

Hands-on operation of the GFIS system takes place in the Engineering Department that works closely with Data Processing. Joel Ivy, one of four engineer technicians under the supervision of Ray Watts, sits at his IBM 5080 Graphics System workstation inputting information either with the alphanumeric terminal or the tracking cursor. With the digitizer tablet, Ivy can place a map on the tablet, tie down known points with points on the screen and start dumping information into the database off the tablet.

Tied to the 3274 Workstation Adapter, the IBM 5085 Graphics Processor is attached to the host processor through the IBM 5088 Graphics Channel Controller that is connected to the host mainframe or the 9370. According to Heinrich, ''All of this equipment is tied together except the 3192. That is still tied straight back in with another coax to the 3274 control unit. And that is where Ivy issues his VM commands to the operating system.''

To understand how the 5080 Workstation and GFIS interact, I will break down each portion of the system.

GFIS Application Architecture

A key component in the overall data management of the organization, GFIS goes far beyond simple graphics display. The GFIS application architecture is composed of a Graphics Program Generator (GPG), an interactive component, and a batch process called the Geo-facilities Database Support (GDBS) that updates and retrieves information assisting utility, petroleum, mining, communication, transportation and construction industries in managing the control of electrical distribution and transmission systems as they exist geographically. Using the concept of an incremental update minimizes the amount of data transferred which in turn improves the system's performance. It also eliminates surprises when two or more users retrieve data from geographic areas that happen to overlap. Data management is coordinated thus eliminating facility problems. This highly interactive system allows for much more efficiency and GFIS maintains a running inventory of the entire system.

This geographic network modeling system provides the basis for support in geoprocessing and mapping applications. It is used to maintain information regarding the company's geographically dispersed assets, allowing the integration of graphics, text and alphanumeric data into a single geographic facilities database. The user no longer has to spend valuable time searching for information in multiple computers or paper files. The needed information is stored in an up-to-date, single integrated database. Duplication of data and costly errors can be eliminated.

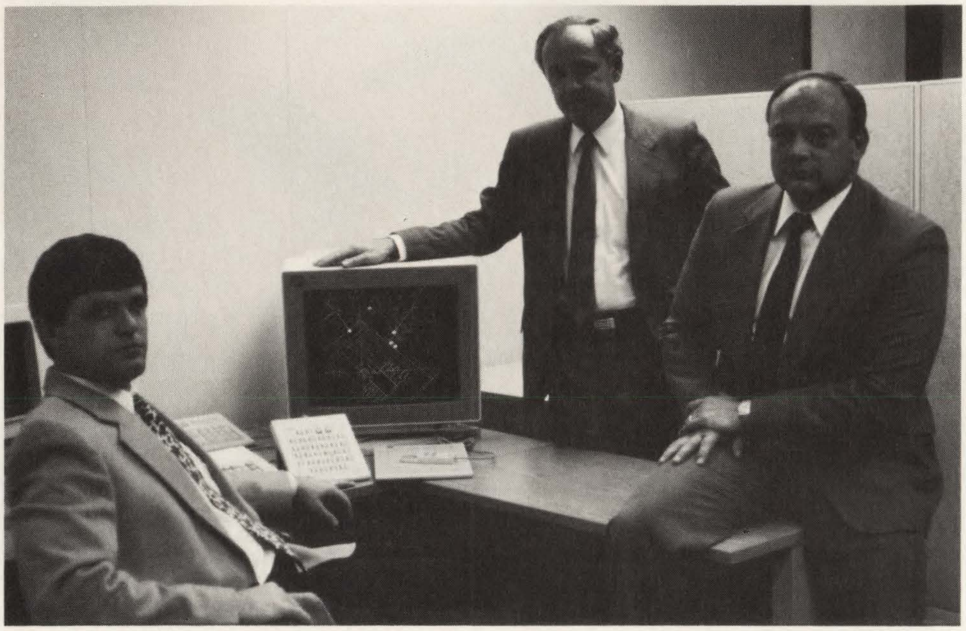
Geo-facilities Database Support

The Geo-facilities Database Support (GDBS) maintains the facilities' database on the host system and is run in the DL/I environment. GDBS also performs retrievals for specified areas or networks for exchange between workspaces using standard interface format files as the exchange medium.

Graphics Program Generator

The Graphics Program Generator (GPG) is a powerful set of high-level interactive programs written in VS/FORTRAN and OS/VS Assembler designed for the creation of a geographic and network modeling system. These programs are intended for use with VM/CMS or MVS/TSO (TNMP uses only VM) and the 5080 Graphics System Workstations. GPG is invaluable for its wide variety of applications such as:

- Creating and maintaining graphic documents with a minimum of user programming
- Data entry, editing, updating and displaying a geographically-oriented facilities database
- Defining and specifying user interaction with the system using menu keys, data entry keyboard and picture components
- Symbol and character generation from user-specified symbol tables and graphic representations of user data items
- The ability to maintain multiple relationships between facilities, picture data and the associated problem data structure of the application.



From left to right are: Joel Ivy, engineering technician; Gary Heinrich, manager of data processing operations; and Ray Watts, supervisor of engineering services.

Much of the information managed by organizations has a geographic association. The need to store and manage large quantities of data becomes a valuable asset for both public and private sectors. The availability of geographic-facilities data greatly enhances daily operational and planning decisions. As a data management system, GPG generates graphic reports for specified networks and can be tailored to meet a user's individual needs. Ad hoc reports may be produced with output displayed either on the screen or on the printer.

Once a GPG user requests a workspace

(a set of sequential records in an interface format file) from the GDBS system, the user may then update or modify the workspace that can be saved in his personal storage. Input and output is provided through both interactive graphics sessions and sequential files. There is even a segment for retaining historical data that has no associated graphics data. "GPG can be used to manage data about any type of facility that can be located in a rectangular two-dimensional coordinate system." The facilities or objects representing an organization's assets might include utility poles, transformers, bridges, railroad tracks and so on. Extraneous information with regard to a physical location might also be included. For instance, the number of traffic accidents at a particular location might be of importance for operational and planning decisions for the highway department.

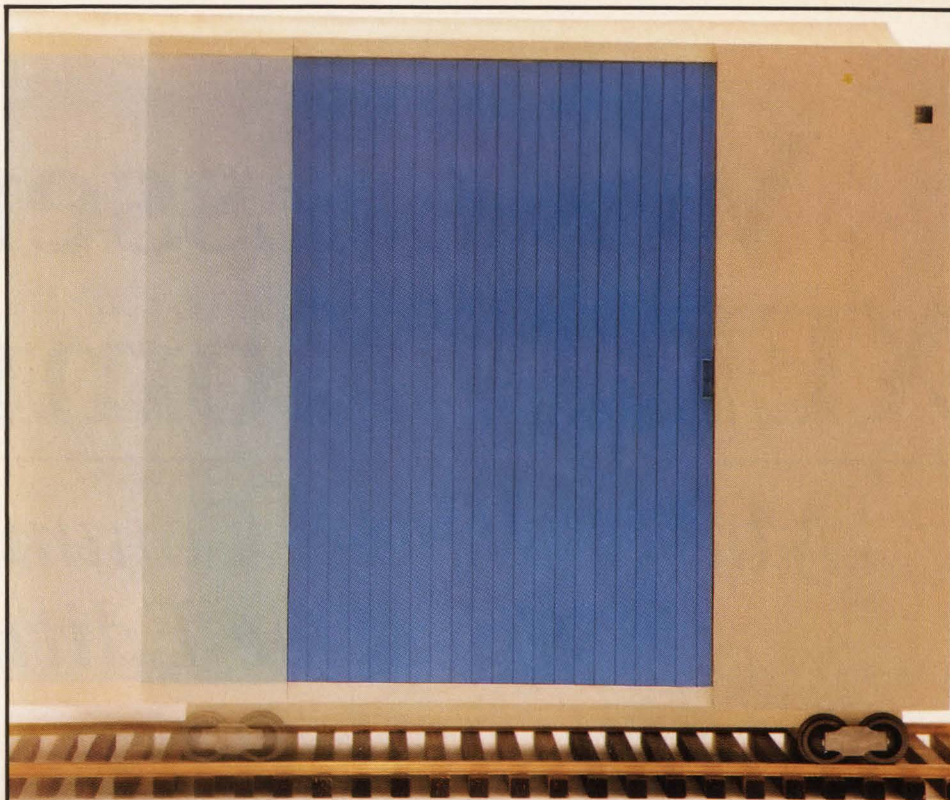
As a "mapping" application, GPG is used where the map has a high data content. Maps that have previously been used as a central source of information to the point of being confusing can now be extracted from the GPG management system with the user displaying only the data needed. Third-party vendors digitize data for TNMP's maps using hand-held computers similar to those used by electronic meter readers.

Field sheets are produced and sent to TNMP on tape. The tape is then loaded into TNMP's own database and can be accessed through any of the workstations. "Each field office has its own work areas on local DASD, but they can't store all of it for their own area, so it is on the mainframe. They call out the data and segments are downloaded to the local system. During an interactive session the user does not have to wait for data to be transferred from the host over the slower telephone lines. It makes for a quite efficient system," says Ivy.

An interactive dialogue on a graphics workstation allows the user to select actions to be performed and to indicate the facility or location involved by pointing to a displayed picture or map. Data can be manipulated any way the user desires. The resulting up-to-date maps allow the user to make better, more informed decisions.

As a key systems component of the GFIS architecture, GPG is designed to meet data management, network modeling and graphic reporting needs utilizing IBM hardware and software products.

See GFIS *page 90*



VM's Essential XEDIT Commands

Master These Commands For Effective Editing

By Steve Eckols

Using XEDIT's default PF key assignments and common prefix commands were two areas covered in the first article (June 1989 issue) dealing with mastery of XEDIT commands. In this article the areas that will be presented include use of XEDIT's vertical scrolling commands, work in input mode, saving your work, ending an editing session, using the on-line help facility and using three XEDIT shortcut commands.

How To Use The Vertical Scrolling Commands

You already know how to use the default assignments for the PF7 and PF8 keys to scroll up and down in a file. In this section, I will introduce you to the commands you can enter in the command area to perform these and other more specific vertical scrolling functions. (XEDIT also provides facilities that let you scroll horizontally.) Figure 10 lists XEDIT's vertical scrolling commands.

The BACKWARD and FORWARD Commands

You use the BACKWARD and FORWARD commands to scroll through a file one screen at a time; they are the commands that are associated by default with the PF7 and PF8 keys. Figure 11 presents their syntax.

If you issue either command without an operand, the default value 1 is supplied for the screens operand. As a result, these commands without operands cause the file to be scrolled up or down one screen. Because those are the functions associated with the PF7 and PF8 keys, there is no reason to go to the extra trouble to key the commands to perform those operations.

However, you will want to use these commands when you need to scroll a large number of screens. When that is the case, all you need to do is supply the number of screens you want to scroll for the screens operand. For example, if you want to scroll down 15 screens, it is easier to enter

FORWARD 15

or the abbreviated form

FO 15

than to press the PF8 key 15 times. It is also more efficient because each time you use a PF key, data transmissions from your terminal to the host system and back are required.

If you want to scroll directly to one end of a file, you can enter an asterisk for the screens operand of either command. For instance,

BACKWARD *

causes the file to be positioned just above its first line (that is, at the TOP OF FILE line on the screen).

The TOP and BOTTOM Commands

An easier way to move to the start or the end of a file you are editing is to use the TOP or BOTTOM command. Because these two commands do not have operands, I have not provided syntax figures for them.

To move to the top of a file, you can enter either

TOP

or

BACKWARD *

These two commands work the same way. They both set your position in the file before its first record (at the TOP OF FILE line).

The two commands to move your current position to the end of a file work a bit differently from one another. If you enter

BOTTOM

your current position becomes the last line in the file. In contrast, if you enter

FOWARD *

your current position is set after the last line in the file (at the END OF FILE line). For most situations, this distinction is not important. However, if you are using these commands in a procedure or if you plan to use certain search options, they can be significant.

The UP, DOWN and NEXT Commands

If you want to perform scrolling operations that move less than a full screen at a time, you can issue the UP, DOWN and NEXT commands. They let you specify the distance to be scrolled in terms of lines instead of screens. Figure 12 presents their syntax. (The DOWN and NEXT commands perform the same function; you can use them interchangeably.)

If you issue one of these commands without an operand, a value of 1 is assumed for the lines operand. As a result, the text scrolls just one line. If you supply the asterisk as the operand of one of these commands, the file is positioned before its first line (for the UP command) or after its last line (for the DOWN or NEXT command).

How To Work In Input Mode

When you create a new file or make extensive additions to an existing file, it is convenient to work in XEDIT's *input mode*. When you enter input mode, lines open up beneath the current line and you are free to type in as much text as you wish. However, you cannot enter other XEDIT commands while you are in input mode. You can work in input mode in two ways: by issuing the INPUT command or by using the SI prefix command.

The INPUT Command

You can enter input mode by using the command

INPUT

Then, a range of empty lines is added to the file you are editing immediately below the current line. You enter text into the new line and you press enter at the end of each. When you have entered the last new line, you press enter twice. That causes XEDIT to leave input mode and return to normal operations.

To understand how this works, take a look at Figure 13. As you can see in part 1 of the figure, I am editing a COBOL source program. The current line is a comment line. In this instance, I want to insert a new block of code in that position, so I keyed the INPUT command into the command line.

When I pressed the enter key, the screen in part 2 of the figure appeared. I want you to notice three things about this display. First, the prefix area has been removed from the screen. Second, the command line has been disabled. (XEDIT puts

* * * INPUT ZONE * * *

into the command line to tell you that it is not available.) Third, notice that all of the lines below the current line have been moved down off the screen and a block of blank lines has been added. The cursor is positioned in the first column of the first of those blank lines.

Each time you press the enter key once in input mode, the line that contains the cursor becomes the current line (in other words, the file is scrolled upward) and more blank lines are added. This is illustrated in parts 3 and 4 of Figure 13. In part 3, I have keyed in a line of code in the first blank line of the input area. Then when I pressed the enter key, the screen in part 4 appeared. Notice that the line I just keyed in is now the current line and an additional blank line has been added to the input area.

If you press the enter key twice in a row, XEDIT leaves input mode and restores normal editing. Blank lines that follow the cursor position when you press enter twice are removed from the file.

Although input mode may seem like an efficient way to enter large blocks of new text, it does have a disadvantage: you end each line by pressing the enter key. Because pressing the enter key causes a data transmission between the terminal and the host system, it can be time-consuming. If you find that you have to wait a long time each time you press the enter key in input mode, it would probably be easier for you (and easier on your system) to add a range of blank lines, then key your text into the file area without using input mode.

FIGURE 10

Vertical scrolling commands

FORWARD	Scrolls a specified number of screens forward (down, toward the end of the file).
BACKWARD	Scrolls a specified number of screens backward (up, toward the beginning of the file).
BOTTOM	Scrolls to the last line of the file.
TOP	Scrolls to the first line of the file.
DOWN	Scrolls a specified number of lines down (forward, toward the bottom of the file). Equivalent to the NEXT command.
NEXT	Scrolls a specified number of lines down (forward, toward the bottom of the file). Equivalent to the DOWN command.
UP	Scrolls a specified number of lines up (backward, toward the top of the file).

Commands to perform vertical scrolling operations

FIGURE 11

The FORWARD and BACKWARD commands

FORWARD

BACKWARD

Explanation

BACKWARD	Specifies that the scroll operation should be toward the beginning of the file (upward). You can abbreviate BACKWARD as BA.
FORWARD	Specifies that the scroll operation should be toward the end of the file (downward). You can abbreviate FORWARD as FO.
screens	The number of screens the display should be scrolled. The default is 1.
	Specifies that the file should be positioned before the first line of the file (for the BACKWARD command) or after the last line of the file (for the FORWARD command).

The FORWARD and BACKWARD commands

FIGURE 12

The DOWN, NEXT, and UP commands

UP

Explanation

DOWN/NEXT	The DOWN command and the NEXT command are equivalent. They specify that the display should be scrolled down (forward, toward the bottom of the file) an indicated number of lines. You can abbreviate DOWN as D. You can abbreviate NEXT as N.
UP	Specifies that the display should be scrolled up (backward, toward the top of the file) an indicated number of lines. You can abbreviate UP as U.
lines	The number of lines the display should be scrolled. The default is 1.
*	Specifies that the file should be positioned before the first line of the file (for the UP command) or after the last line of the file (for the DOWN or NEXT command).

The DOWN, NEXT and UP commands

FIGURE 13 PARTS 1 and 2

Entering the INPUT command . . .

```
  LISTMAST COBOL    A1  F 80  TRUNC = 72 SIZE = 236 LINE = 59 COL = 1 ALT = 2

=====            05  TODAYS-TIME              PIC X(8).
=====            05  TODAYS-TIME-R REDEFINES TODAYS-TIME.
=====                10  TODAYS-HOURS         PIC S99.
SI===                10  TODAYS-MINUTES       PIC XX.
=====                10  FILLER               PIC X(4).
=====        *
=====        01  COUNT-FIELDS                 COMP-3.
=====        *
=====            05  RECORD-COUNT             PIC S9(3)         VALUE ZERO.
=====        *
      |...+....1....+....2....+....3....+....4....+....5....+....6....+....7>
=====        01  INVENTORY-MASTER-RECORD.
=====        *
=====            05  IM-STATUS                        PIC X.
=====                88  IM-IN-PRINT                  VALUE 'I'.
=====                88  IM-OUT-OF-PRINT              VALUE 'O'.
=====            05  IM-DESCRIPTIVE-DATA.
=====                10  IM-BOOK-CODE                 PIC X(4).
=====                10  IM-AUTHOR-LAST-NAME          PIC X(4).
=====                10  IM-BOOK-TITLE                PIC X(4).
====> INPUT_
                                                              X E D I T  1 FILE
```

. . . disables the prefix area and the command line and opens up a range of lines following the current line for text entry.

```
  LISTMAST COBOL    A1  F 80  TRUNC = 72 SIZE = 245 LINE = 59 COL = 1 ALT = 2
INPUT MODE:
            05  TODAYS-TIME              PIC X(8).
            05  TODAYS-TIME-R REDEFINES TODAYS-TIME.
                10  TODAYS-HOURS         PIC S99.
                10  TODAYS-MINUTES       PIC XX.
                10  FILLER               PIC X(4).
        *
        01  COUNT-FIELDS                 COMP-3.
        *
            05  RECORD-COUNT             PIC S9(3)         VALUE ZERO.
        *
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7>
_




====> * * * INPUT ZONE * * *
                                                            INPUT-MODE 1 FILE
```

FIGURE 13 PARTS 3 and 4

I have keyed in a new line in the input area and when I press the enter key . . .

```
  LISTMAST COBOL    A1  F 80  TRUNC = 72 SIZE = 245 LINE = 59 COL = 1 ALT = 2
INPUT MODE:
            05  TODAYS-TIME              PIC X(8).
            05  TODAYS-TIME-R REDEFINES TODAYS-TIME.
                10  TODAYS-HOURS         PIC S99.
                10  TODAYS-MINUTES       PIC XX.
                10  FILLER               PIC X(4).
        *
        01  COUNT-FIELDS                 COMP-3.
        *
            05  RECORD-COUNT             PIC S9(3)         VALUE ZERO.
        *
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7>
        01  INVENTORY-ITEM-TABLE._




====> * * * INPUT ZONE * * *
                                                            INPUT-MODE 1 FILE
```

. . . the file is scrolled upward, the new text is shifted up out of the input area and a new line is opened up at the end of the input area.

```
  LISTMAST COBOL    A1  F 80  TRUNC = 72 SIZE = 245 LINE = 60 COL = 1 ALT = 3
INPUT MODE:
            05  TODAYS-TIME-R REDEFINES TODAYS-TIME.
                10  TODAYS-HOURS         PIC S99.
                10  TODAYS-MINUTES       PIC XX.
                10  FILLER               PIC X(4).
        *
        01  COUNT-FIELDS                 COMP-3.
        *
            05  RECORD-COUNT             PIC S9(3)         VALUE ZERO.
        *
        01  INVENTORY-ITEM-TABLE.
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7>
_




====> * * * INPUT ZONE * * *
                                                            INPUT-MODE 1 FILE
```

The SI Prefix Command

Another way to enter input mode is to use the SI prefix command. Figure 14 shows you how it works. You start by keying in the SI prefix command in the prefix area of the line after which you want to add text as part 1 of the figure shows.

When you press the enter key, a single blank line is inserted after the indicated line as you can see in part 2 of the figure. Notice that the cursor is automatically positioned in the column where the first non-blank character occurs in the preceding line. This can be useful when you are entering program source code or any other text where alignment is important. (Because alignment of similar program elements is one of the techniques used in structured programming to make programs easier to interpret, this editing feature is called *structured input.)*

You enter one line at a time in structured input mode. Each time you press the enter key, a new line is added and the cursor is positioned so text you enter into it will be aligned with the text in the preceding line. To leave structured input mode, you press the enter key on a blank line.

How To Save a File You Have Edited: The SAVE Command

When you are ready to end an editing session, you will almost always want to save the work you have done. (In fact, the only time you will not want to save your work is when you have made changes that you decide you do not want to keep. That is possible but not common.) To save your work, you use the SAVE command illustrated in Figure 15. Most of the time, you will simply enter

SAVE

which causes the file you are working on to be replaced on disk.

If you want to leave the original file intact and save your work under another name, you can supply a different file identifier on the SAVE command. In the file identifier, you can specify equal signs to indicate that one or more of the identifier's current components should remain unchanged. For example, if you are editing a file whose identifier is LISTMAST COBOL A and you want to save your work in a file with the identifier LSTMST2 COBOL A, you would enter the command

SAVE LSTMST2 = =

FIGURE 14 PARTS 1 and 2

Entering the SI prefix command . . .

```
  LISTMAST COBOL    A1  F 80  TRUNC = 72 SIZE = 236 LINE = 59 COL = 1 ALT = 2

=====            05  TODAYS-TIME              PIC X(8).
=====            05  TODAYS-TIME-R REDEFINES TODAYS-TIME.
=====                10  TODAYS-HOURS         PIC S99.
SI===                10  TODAYS-MINUTES       PIC XX.
=====                10  FILLER               PIC X(4).
=====        *
=====        01  COUNT-FIELDS                 COMP-3.
=====        *
=====            05  RECORD-COUNT             PIC S9(3)         VALUE ZERO.
=====        *
      |...+....1....+....2....+....3....+....4....+....5....+....6....+....7>
=====        01  INVENTORY-MASTER-RECORD.
=====        *
=====            05  IM-STATUS                        PIC X.
=====                88  IM-IN-PRINT                  VALUE 'I'.
=====                88  IM-OUT-OF-PRINT              VALUE 'O'.
=====            05  IM-DESCRIPTIVE-DATA.
=====                10  IM-BOOK-CODE                 PIC X(4).
=====                10  IM-AUTHOR-LAST-NAME          PIC X(4).
=====                10  IM-BOOK-TITLE                PIC X(4).
====>
                                                              X E D I T  1 FILE
```

. . . opens a single line below the indicated line with the cursor beneath the first non-blank column of the indicated line.

```
  LISTMAST COBOL    A1  F 80  TRUNC = 72 SIZE = 236 LINE = 59 COL = 1 ALT = 2

=====            05  TODAYS-TIME              PIC X(8).
=====            05  TODAYS-TIME-R REDEFINES TODAYS-TIME.
=====                10  TODAYS-HOURS         PIC S99.
=====                10  TODAYS-MINUTES       PIC XX.
......       *       _
=====                10  FILLER               PIC X(4).
=====        *
=====        01  COUNT-FIELDS                 COMP-3.
=====        *
=====            05  RECORD-COUNT             PIC S9(3)         VALUE ZERO.
      |...+....1....+....2....+....3....+....4....+....5....+....6....+....7>
=====        *
=====        01  INVENTORY-MASTER-RECORD.
=====        *
=====            05  IM-STATUS                        PIC X.
=====                88  IM-IN-PRINT                  VALUE 'I'.
=====                88  IM-OUT-OF-PRINT              VALUE 'O'.
=====            05  IM-DESCRIPTIVE-DATA.
=====                10  IM-BOOK-CODE                 PIC X(4).
=====                10  IM-AUTHOR-LAST-NAME          PIC X(4).
====>
                                                                ' PENDING
```

or

SAVE LSTMST2 COBOL A

If you specify the name of a file that already exists, XEDIT warns you, but gives you the opportunity to replace the existing file. A line like

FILE 'LSTMST2 'COBOL A1'
ALREADY EXISTS.
USE FFILE/SSAVE.

appears. You can then cause the existing file to be replaced by issuing the SSAVE command like

SSAVE LSTMST2 COBOL A

where you identify the file to be replaced. (As you might guess from reading the message, it is also displayed if you use the FILE command for an existing file; I will present the FILE command in a moment.)

How To End Your XEDIT Session: The QUIT, CANCEL and FILE Commands

To leave the editor, you can issue one of three commands: QUIT, CANCEL or

FIGURE 15

The SAVE command

`SAVE [filename [filetype [filemode]]]`

Explanation

filename	The filename component of the file identifier to be used for the file to be saved. You may specify = for the filename operand; if you do, the filename of the file you are currently editing is assumed.
filetype	The filetype component of the file identifier to be used for the file to be saved. You may specify = for the filetype operand; if you do, the filetype of the file you are currently editing is assumed.
filemode	The filemode component of the file identifier to be used for the file to be saved.

The SAVE command

FILE. You are already familiar with the QUIT command: it is the command that is associated with the PF3 key. Whether you issue QUIT direclty by keying it in or indirectly by pressing PF3, it works in the same way. If you have not made any editing changes to the file, XEDIT ends and you are returned to the CMS environment. If you have made changes to the file that have not been saved, XEDIT warns you by displaying the message

FILE HAS BEEN CHANGED.
USE QQUIT TO QUIT ANYWAY.

If you are sure you want to abandon your work, you can key in

QQUIT

and the editor ends. It does *not* record the editing changes on your original file.

The CANCEL command works like QUIT, only it is really intended for use when you are editing multiple files. CANCEL causes QUIT commands to be executed for all open files. When you are working with only one file, CANCEL and QUIT have the same effect.

The FILE command is more powerful: it combines the functions of SAVE and QUIT. So if you want to save your work and end the editor, the command you issue is FILE. The FILE command has the same syntax as the SAVE command as you can see in Figure 16. If you issue it with no operands, like

FILE

the file that is currently open is replaced with your edited work. If you specify a file identifier, your edited work is saved under that name. If you specify an identifier for a file that already exists, XEDIT requires you to use FFILE instead of FILE.

How To Get Help If You Need It: The HELP Command

As you use XEDIT, you will sometimes need to refer to reference material to refresh your memory about the exact syntax to use for a particular command. Of course, you can return to the XEDIT manuals or this article but for quick, simple inquiries you might also consider using the editor's *on-line help facility.*

To use the help facility, you issue the HELP command illustrated in Figure 17. As you can see, the command has three operands. If you issue the command with the MENU operand or without an operand (the MENU operand is the default), the screen in Figure 18 appears. On this screen, you can move the cursor to any command's name and press the enter key. Then, reference information for that command appears.

F I G U R E 16

The FILE command

`FILE [filename [filetype [filemode]]]`

Explanation

filename	The filename component of the file identifier to be used for the file to be saved. You may specify = for the filename operand; if you do, the filename of the file you are currently editing is assumed.
filetype	The filetype component of the file identifier to be used for the file to be saved. You may specify = for the filetype operand; if you do, the filetype of the file you are currently editing is assumed.
filemode	The filemode component of the file identifier to be used for the file to be saved.

The FILE command

If you issue the HELP command in the format

HELP TASK

the screen in Figure 19 appears. In this case, a menu of functions is supplied. You can then select the one you are interested in to display a subordinate menu. As you work down the menu structure, you will eventually reach reference information for the command you need to know to perform a particular task.

The third way you can enter the help facility is by specifying the name of a particular command you are interested in on the HELP command. Then, information for the command you specify is displayed. For instance, the command

HELP DOWN

causes the help facility to display reference information about the DOWN command as you can see in Figure 20.

In Figures 18, 19 and 20, you can see that functions are associated with PF keys. Of most interest are the PF keys you use most often during regular editing: PF7, PF8 and PF3. You use the PF7 and PF8 keys to scroll forward and backward in the help facility displays. You use the PF3 key to leave the help facility (or, if you are at a subordinate screen within the help facility, to return to its superior menu).

Three Shortcut Commands

Finally in this article, I want to tell you about three shortcut commands that can make XEDIT easier for you to use. They are the commands listed in Figure 21: &, = and ?.

How To Retain a Command You Have Entered: the & Command

If you use the ampersand as a prefix for a command you enter in the command line, the command is retained in the command line after it has been processed. That can make it easy for you to issue the same command repeatedly. For example, if you enter

&DOWN

in the command line, all you have to do is press the enter key repeatedly to scroll down one line at a time. The DOWN command with its preceding ampersand remains in the command line until you erase it or replace it by typing a different command over it.

How To Reissue a Command You Have Entered: the = Command

Another way to reissue the last command you entered is to use the = command. By default, = is associated with the PF9 key, so it is easy to issue the same command over and over. All you have to do is key it in once in the command line, execute it once by pressing the enter key, then press the PF9 key as often as you need to reexecute it.

How To Restore a Command You Have Entered: the ? Command

If you forget the last command you issued and you think you might want to use it again, you can issue the ? command. It causes the last command to be restored to the command line. The ? command can be useful when you are issuing long, complex commands and one you want to enter is much like one you just entered. After you recall the last command with ?, you can make changes to it before you issue it again.

Discussion

Now that you are familiar with the commands and concepts this article has presented, you should be able to use XEDIT for a wide range of editing tasks.

This article is an excerpt from Eckols' book titled VM/CMS: XEDIT Commands and Features, *published by Mike Murach & Associates, 4697 W. Jacquelyn Avenue, Fresno, CA 93722, (800) 221-5528; in California (800) 221-5527.*

ABOUT THE AUTHOR

Since 1980, Steve Eckols has written nine books on IBM mainframe subjects ranging from system development to VSE to IMS to VM.

FIGURE 17

The HELP command

HELP

Explanation

HELP	You can abbreviate HELP as H.
MENU	Specifies that a menu of XEDIT commands should be displayed from which you can select one for display. MENU is the default operand for the HELP command.
TASK	Specifies that a menu of XEDIT editing functions should be displayed from which you can select one for additional information.
subcommand	An XEDIT command about which you want more information.

The HELP command

FIGURE 18

```
= = > XEDIT MENU < = = = = = = > HELP INFORMATION < = = = = = > LINE = = = > 1 OF 22

A SUBCOMMAND MAY BE SELECTED FOR VIEWING BY PLACING THE CURSOR UNDER ANY CHARACTER
OF THE SUBCOMMAND NAME AND THEN PRESSING THE ENTER KEY OR THE PF 1 KEY. *PREFIX AND
*SET WILL DISPLAY MENUS OF ALL AVAILABLE PREFIX SUBCOMMANDS AND MACROS, OR SET
OPERANDS. :XEDIT WILL DISPLAY A MENU OF SPECIFIC XEDIT OPERATIONS. FOR A DESCRIPTION OF
THE OPERANDS AND OPTIONS TYPE HELP HELP.

&          CFIRST     CREPLACE  HEXTYPE   NFIND     RENOM     SOS
*PREFIX    CCHANGE    CURSOR    INPUT     NFINDUP   REPEAT    SPLIT
*SET       CINSERT    DELETE    JOIN      OVERLAY   REPLACE   SPLTJOIN
?          CLAST      DOWN      LEFT      PARSE     RESET     STACK
:XEDIT     CLOCATE    DUPLICAT  LOAD      POWERIMP  RESTORE   STATUS
=          CMS        EMSG      LOCATE    PRESERVE  RGTLEFT   TOP
ADD        CMSG       EXPAND    LOWERCAS  PURGE     RIGHT     TRANSFER
ALL        CMSXEDIT   EXTRACT   LPREFIX   PUT       SAVE      TYPE
ALTER      COMMAND    FILE      MACRO     PUTD      SCHANGE   UP
BACKWARD   COMPRESS   FIND      MERGE     QUERY     SET       UPPERCAS
1 = HELP        2 = TOP      3 = QUIT    4 = RETURN        5 = CLOCATE     6 = ?
7 = BACKWARD    9 = FORWARD  9 = PFKEY  10 = BACKWARD ½   11 = FORWARD ½  12 = CURSOR

====> _
                                                              MACRO-READ 1 FILE
```

FIGURE 19

```
= = > XEDIT TASK < = = = = = = > HELP INFORMATION < = = = = = > LINE = = = > 1 OF 12
PURPOSE: TO HELP YOU USE THE SYSTEM PRODUCT EDITOR
         TO CREATE AND CHANGE INFORMATION IN YOUR FILES.

MOVE THE CURSOR TO THE TASK THAT YOU WANT, THEN PRESS THE ENTER KEY.


CREATE A NEW FILE
ENTER NEW TEXT INTO AN EXISTING FILE
MAKE CHANGES TO EXISTING TEXT IN A FILE
MOVE AROUND WITHIN A FILE
END AN EDITING SESSION OR SAVE CHANGES
TAILOR YOUR EDITING SCREEN
* * * END OF FILE * * *



1 = HELP        2 = TOP      3 = QUIT    4 = RETURN        5 = CLOCATE     6 = ?
7 = BACKWARD    9 = FORWARD  9 = PFKEY  10 = BACKWARD ½   11 = FORWARD ½  12 = CURSOR

====> _
                                                              MACRO-READ 1 FILE
```

FIGURE 20

```
= = > XEDIT DOWN < = = = = = = > HELP INFORMATION < = = = = = > LINE < = = = = > 1 OF 55
DOWN

USE THE DOWN SUBCOMMAND TO ADVANCE THE LINE POINTER A SPECIFIED NUMBER OF LINES
TOWARD THE END OF THE FILE. THE LINE POINTED TO BECOMES THE NEW CURRENT LINES.
(THE DOWN SUBCOMMAND IS EQUIVALENT TO THE NEXT SUBCOMMAND.)


THE FORMAT OF THE DOWN SUBCOMMAND IS:

+-----------+------------------------------------------------+
|           |                                                |
| DOWN      | (N I*I 1)                                      |
|           |                                                |
+-----------+------------------------------------------------+

WHERE:

N
1 = HELP        2 = TOP      3 = QUIT    4 = RETURN        5 = CLOCATE     6 = ?
7 = BACKWARD    9 = FORWARD  9 = PFKEY  10 = BACKWARD ½   11 = FORWARD ½  12 = CURSOR

=====> _
                                                              MACRO-READ 1 FILE
```

FIGURE 21

The &, = and ? commands

&	Use the ampersand as a prefix for any XEDIT command to cause it to be retained in the command line after it is executed.
=	Reexecutes the last XEDIT command.
?	Redisplays the last XEDIT command.

The &, = and ? commands

SLOs
SLAs
SLM

The Three Phases of Service Levels

By Cheryl Watson

I am sure you have read quite a bit about service levels: Service Level Objectives (SLOs), Service Level Agreements (SLAs) and Service Level Management (SLM). However, here is a way of looking at them as steps in a process. You might stop anywhere along that process! Service goals is another term often used for SLOs or SLAs; however, I want to differentiate between objectives and agreements in this article and give some recommendations for setting SLOs and SLAs in your installation. I will address the following issues: what is service? what are SLOs? what are SLAs? how do I manage service levels?

What Is Service?

Service provided by the data center to a user includes: on-line response time, system availability, batch turnaround and accuracy. As a side note, IBM markets products and enhancements based on Reliability, Availability and Serviceability (RAS). While IBM uses RAS improvements to market products, RAS is a worthwhile goal for data centers. Reliability basically means that users can rely on the service (it will not abend or produce incorrect results). Availability means that the system is available when it is expected; that is, CICS and TSO are available from 7 a.m. to 6 p.m. without interruption. Serviceability refers to facilities provided to the user. RAS would be a worthwhile objective for any data center to provide for its users; however, it is not readily measurable. Take a closer look at these services.

On-line Response Time

Obviously, on-line response time is critical in most installations. Poor response time impacts productivity and service to customers. The frustration and disenchantment resulting from poor service can be costly. The bottom line: poor service to customers means lost revenues; poor service to the staff can result in a large turnover! There has been more than one time when I have sat at a terminal updating my resume in my mind because I am waiting for a response, *any* response, from the system! There are also several studies that deal with user productivity as a result of consistently good (sub-second?) response times.

The major problem with measuring on-line response time is the lack of tools for measurement. Internal response time is relatively simple to measure. However, it is simply not a measure of user response time. For example, a user in New York that is logged onto a CICS system in San Francisco may experience response times of several seconds (if not minutes!) while internal response time within CICS is in sub-seconds. The equipment and/or software needed to measure the actual end-user response time can represent a large investment.

System Availability

System availability is almost more important than on-line response time. I would rather be able to get to a system with a two-minute response time than not be able to access it at all! There are various pieces of system availability that need to be controlled: is the operating system up? is the on-line system up? are the databases open? are initiators up and running my job class? are the printers on-line and printing reports? is a product up and available when I need it? If the answer is no to any of these questions, users will see the data center as not providing the service they need. That is one black mark for you!

Again, unfortunately, there are not adequate measurement tools to measure availability. Most data centers write their own programs to define availability.

If the system is not up, response time is measured in minutes or hours instead of seconds. (Unfortunately, there are cases of days of downtime).

Batch Turnaround

There are generally two types of batch turnaround: *test* and *production*. Typically, the primary output of a test batch job is the production of a report. Some users may want to look at the report on-line (held SYSOUT) while others may want the report printed and delivered. Production batch jobs, on the other hand, tend to be deadline oriented: the job needs to be completed by 6 a.m. before the on-line systems come up.

Again, measurement sources may not always report the information that is needed. Notice that delivery of reports is a manual process and cannot be measured or trailed without manual input (for instance a delivery log).

Accuracy

Speed may not be the only factor in good service. How many times has the data center been responsible for mistakes in production, mounting the wrong input tape, deleting files, dropping tapes, running a job twice, not loading a database correctly and so on. All of these result in poor service. One objective could be to define invalid runs (either by the number of reruns, the amount of time lost to reruns and so on).

FIGURE 1

Service Level Objectives

Note: These are used internally by the data center.

	min	max
System capacity during peak hours:		
Average CPU busy:	80%	90%
Average demand paging rate:	100	150
Max DASD channel busy:		40%
. . . etc. . . .		
Users — TSO users:	45	55
CICS Transactions per min:	250	300
Batch jobs per hour:	50	75
CICS service during prime time:		
Average response time in seconds:	2	3
CICS availability:	95%	
Batch job service:		
Class S (normal test) turnaround:	30min	45min
Class T (hot batch) turnaround:	10min	20min

What Are SLOs?

Everyone has implied SLOs. If the service that is provided to the end user is not agreeable, the phone rings! In order to avoid complaints, data center management has usually developed an internal set of objectives. These objectives can bear some relationship to the user's view of the world or may simply be *techy* measures of the computer system.

Examples of data center SLOs are shown in Figure 1. Some of the characteristics of SLOs are: 1) easy to measure, 2) easy to report, 3) need no user input or interface, 4) the system capacity thresholds are estimates of possible impact (that is, in the past we may have seen CICS long response times whenever the CPU busy exceeded 90 percent, so do not let it exceed that level again), 5) system availability does not take into account databases or applications not being available, 6) performance analysts and capacity planners can use the values without interface to the users, 7) the objectives can be set up by one person in a day and 8) system loads, such as the number of TSO users, are kept so the data center can determine whether problems are due to increased load or to other problems on the system.

The first phase of setting SLAs is to first determine the implied or documented SLOs that are being used in the installation. In the absence of documented SLOs, I will find a period of time that the installation was running well (that is, no phone calls from the users, normal system load) and use the current measurements. I can always adjust them later! SLOs are a required first step of SLAs! Are yours in place?

What Are SLAs?

Here is where this process gets tricky. An SLA implies that the user agrees to the SLOs and is willing to accept them. An SLA contains several types of information that the SLO did not need. Agreements are made with each user and should include the items shown in Figure 2. Obviously, it will take a great deal of time from the data center and the user to come to some agreements. People tend to approach this step as adversaries; however, it should be a supportive set of meetings (yes, it will take more than one!).

SLAs require the following: a signed contract with each user (without a signature, no responsibility is assumed), on-line response times by application, batch turnaround by class, system availability by application, accuracy limits, load of each application by transactions per hour and batch jobs by day, application load by resource requirements, a plan for reporting SLAs, priorities if service cannot be delivered, penalties if the user exceeds the load (maybe decreased service levels), penalties if the data center does not provide service (maybe reduced charges) and a schedule for follow-up meetings and interface.

Does this sound like a big job? You bet! However, everyone can start simply. The first step is sometimes the biggest. Find out who the users are, who the contact person is, who the manager is, how many resources are they consuming each month (that is, who are my *big* users?), what do they think of the current service? Without job or accounting standards, you cannot even think about service levels. Just take these one step at a time.

Figure 2 shows some sample SLAs. Notice the key elements of response time,

See Service Levels *page 82*

Succeed

PLATINUM *technology, inc.* The DB2 Company™

PLATINUM *technology, inc.* is the only full service software company specializing exclusively in DB2. All our effort and energy is focused on delivering the broadest array of quality solutions for the DB2 user. As a result, IBM® has designated PLATINUM a Business Partner through the Authorized Application Specialist program specifically for DB2.

PLATINUM PRODUCTS

PLATINUM offers a complete line of software tools, education, and published products to ensure your success with DB2.

Software Products

The PLATINUM product portfolio consists of a complete family of administration, development, and end user DB2 software products. All are compatible with DB2 V1.3 & V2.1. The software products include:

- **RC/Query™** - A comprehensive DB2 catalog query tool.
- **RC/Update™** - The industry's best DB2 object management and data editing tool.
- **RC/Migrator™** - A complete object and data migration tool.
- **RC/Secure™** - An extensive DB2 security management tool.
- **PLATINUM Database Analyzer™** - The DB2 database and DASD analysis, audit, and management tool.
- **PLATINUM Report Facility™** - The DB2 query and reporting system for developers and end users operating in TSO and CICS environments.

DB2 Education Courses

A complete series of hands-on DB2 training courses. PLATINUM courses cover all aspects of DB2, QMF, and CSP. All courses are available either at your facility or at our Corporate Education Center.

- Introduction to DB2
- DB2/SQL Application Programming
- DB2 Application Planning and Database Design
- DB2 Database and System Administration
- Using DB2 and QMF
- CSP Application Programming

Published Products

The most recognized and authoritative DB2 standards, methods, and guidelines for DB2 implementation.

- **PLATINUM DB2 Guide/Online™** - The industry's leading standards manual for design, development, and administration of DB2 systems.
- **The PLATINUM Reference™ for DB2** - The quick, pocket-sized reference for DB2 information.

Support

All products and services carry our PLATINUM Quality Assurance Guarantee. Support is available 24 hours a day, 7 days a week via our toll-free hot line.

WORLDWIDE AVAILABILITY

PLATINUM's products and services are available around the world through our Affiliate Network. PLATINUM's full service capabilities include local support, education, and superior DB2 professional services.

with DB2!

PLATINUM *technology, inc.*
555 WatersEdge Drive
Lombard, IL 60148-9930
(312) 620-5000 FAX (312) 953-1923

For further information, in-house demonstration, or our exclusive no-obligation product evaluation call:

1-800-442-6861

CIRCLE #7 on Reader Service Card ▲

Capture Ratios: Fact Or Legend

By H. Pat Artis

Perhaps some of the most interesting aspects of modern life are the myths and legends that somehow obtain acceptance and credibility by simply being retold hundreds and thousands of times. Two of my favorite legends are alligators in the sewers and capture ratios for MVS workloads.

Probably, most readers of this article will have heard the legend of alligators in the New York City sewers at least once. You remember, mom, pop and the kids leave the Boroughs for a vacation in Florida. While there, little Billy falls in love with the reptiles and demands one of the cute little baby alligators to take home as a pet. Even as they drive home, the little reptile rapidly grows on a diet of cookies and Twinkies provided with loving care by Billy. However, once back in the city, the growth trend continues as the little reptile becomes more ravenous every day. Then, one day the moment of truth comes. The family cat is missing and the alligator is leaving fur balls. In a decisive moment, mom flushes the gator down the commode before a child is lost to the growing menace. Unfortunately, the legend does not end here. Unharmed by its new environment in the sewers, the reptile rapidly matures into a 20-foot specimen that is soon feasting on city workers.

While the legend grows richer with every telling, I have never quite been sure why there are not giant alligators in the sewers of Washington, Philadelphia or Boston. Perhaps the parents are more hardhearted or maybe their sewage is not up to New York City standards. Whatever the reason, the legend continues.

Another one of my favorite legends is the concept of a *capture ratio*. Briefly, a capture ratio is an attempt to define a factor that estimates the total CPU time actually used by a job or transaction as a ratio to the CPU time that was attributed to the task by the operating system. In MVS environments, it is not uncommon to find that the total utilization reported for the system's workloads by the System Measurement Facility (SMF) or Resource Management Facility (RMF) is 20 or 30 percent lower than the total CPU utilization that is measured for the hardware. Supposedly, a capture ratio is a silver bullet that addresses this measurement problem. When you know the capture ratio, all you have to do is multiply the reported CPU time by the magic value to estimate the actual CPU time used by the workload. For example, if a hypothetical capture ratio for CICS was 1.7 (the CICS version number) and the CPU time reported for CICS was 60 minutes, the total CPU consumption of the CICS workload could be estimated by multiplying 60 times 1.7. Evaluating this expression, you would estimate that the total CPU time for the CICS workload was 102 minutes rather than the 60 minutes that was reported by SMF or RMF. Clearly, obtaining a reliable understanding of this unallocated utilization is critical for cost allocation and capacity planning activities.

Like the little 'gator who prospered on Twinkies, the legend of capture ratios has been fostered by a long stream of IBM publications from the *Washington Systems Center* and manuals like *IBM's Performance Notebook*. Using benchmark job streams, IBM measures critical workload types like IMS, CICS, DB2, TSO and batch under new operating system releases to provide guidance for users in planning for a system's growth. Like most legends, there are more than a few grains of truth in the capture ratio concept. However, the notion that there is a "one-size-fits-all" capture ratio that will correct the measured CPU time of CICS

GIANT IS NOW ON LINE

The Global Information Access Network

Making decisions about hardware, software and data processing services is a difficult task. Just trying to identify the available products can be an enormous dilemma.

The time has come to benefit from the same technology you work with day-to-day. Explore the numerous products and services available from the comfort of your office. GIANT provides a complete data base at your fingertips through our dial-up network. And best of all, it's FREE!

Free Access — No User-id Required — No Password — On-line Instructions.

Dial-up access is available from your PC. Use any of the popular communications packages that emulate a VT100 terminal (or use a VT100 terminal). We have successfully tested using PROCOMM Plus™, CROSSTALK™ and KERMIT.

2400 → (414) 546-8480
1200 → (414) 546-8492

Terminal type	**VT100**
Line speed	**2400 or 1200**
Parity	**None**
Data bits	**8**
Stop bits	**2**

Dial-up is available Monday through Saturday, 24 hours a day. (BISYNC and SNA dial-up support coming soon)

If a dial-up terminal is not available, then FAX your request to us. We will search our data base and FAX a list of products or services to you.

(414) 546-8499 (FAX)

FAX is available 7 days a week, 24 hours a day.

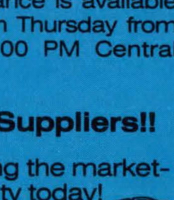
Or you may call us direct, and we will search our data base immediately to find the products and services you need.

(414) 546-8496
(Voice Lines for Personal Service)

Personal assistance is available Monday through Thursday from 9:00 AM to 1:00 PM Central Time.

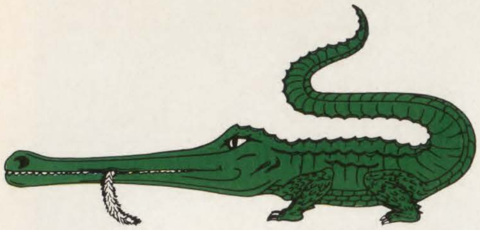
Software Manufacturers!! Hardware Manufacturers!! Service Suppliers!!

Advertise your products and services on GIANT today. Put your message into every data center across the country . . . everyday. Join with GIANT and become part of the GIANT team: we're making the marketing media of the future a reality today! Give us a call and we will send our GIANT information kit to you immediately.



PSI Productivity Systems, Inc.
P.O. Box 20958
Milwaukee, WI 53220
414/321-8688

PROCOMM Plus is a registered trademark of DataStorm Technologies, Inc. CROSSTALK is a registered trademark of Data Communications Associates, Inc.



or any other type of workload is a concept that requires critical evaluation.

In this article, I will examine the efficacy of capture ratios and attempt to shed a little light on the alligator issue too.

Why Is Some Of The CPU Time Missing?

Before commenting on the efficacy of capture ratios, it is important to ask one key question. Why is some of the CPU time missing? Prior to the introduction of OS/MFT and OS/MVT operating systems in the mid-1960s, computers ran jobs one at a time. Since a lot of the elapsed time of a job was represented by waits for tapes to transfer or disks to rotate, actual CPU utilization was quite small. This single job at a time approach to system management is similar to the way DOS manages personal computer tasks today. System designers realized that the best way to improve the overall utilization of the CPU was to use the time that the CPU was waiting on I/Os or other events for one task to service the CPU requirements of another. This approach, called *multiprogramming,* is also the heart of IBM's new OS/2 operating system designed to allow users to realize more benefits from their personal computers.

One consequence of multiprogramming is that waiting tasks leave the operating system with requests to service (that is, they are waiting to be interrupted) while the operating system processes other tasks. While a task is executing, MVS/XA accumulates the CPU time that is consumed by the task. This CPU time is called Task Control Block (TCB) CPU time. In addition, MVS accumulates Service Request Block (SRB) CPU time for asynchronous activities scheduled by the task. However, when a task is not in control of the CPU, it is not simple for the operating system to attribute the CPU time that is used to service operating system requests to the task that originated it. From a practical standpoint, the overhead of addressing the control blocks for a waiting task and adding in the CPU overhead as the requests occur can be prohibitive in some operating system designs since the number of instructions required to service many of the requests is relatively small. Hence, MVS/XA allows the time to evaporate rather than attempting to bill services back to the requesting task. This approach to attributing CPU time is common to most operating systems.

For example, assume that it takes the operating system five thousand instructions to service an I/O request for a task. On a 10 MIP uniprocessor, this would correspond to approximately one-half of a millisecond of CPU time. The term *path length* is used to refer to the estimate of the number of instructions (that is, CPU time) that is required to service a request. If you assume that it takes MVS just 500 instructions to establish addressability to a task's control blocks, an important question becomes obvious. Is it worth increasing the overhead of the operating system by 10 percent in this hypothetical example to improve the accuracy of system measurement? When you realize that the operating system may be servicing hundreds of I/O requests per second, it is clear that the overhead required to produce exact measurements is likely to be unacceptable based on the hypothetical numbers that have been described.

Hence, return to capture ratios. If you could just estimate the types and frequencies of operating system requests for a workload like CICS, then you could develop a ratio that estimates the actual CPU time based on what was measured by the system. Before accepting this premise, ask what are the primary operating system services offered by MVS and what causes them to occur.

Operating System Services

Although MVS/XA offers dozens of different types of services to the tasks it manages, the most frequently used services are paging, swapping and EXCP (that is, I/O) processing. To understand why "one-size-fits-all" capture ratios are unreliable, examine when and why a task invokes these services.

Begin by considering a hypothetical CICS task executing stand-alone on a large dedicated MVS processor. Periodically, CICS will issue I/O requests to obtain data for transaction programs and to get or send messages to user terminals. The rate at which I/Os are issued is a function of the complexity of the application programs being executed by the CICS users. Message switching applications may issue an I/O every few CPU milliseconds while rocket science applications tend to be somewhat more CPU intensive. Since this I/O intensity will tend to vary from application to application and from site to site, the concept of a silver bullet capture ratio for all CICS systems begins to look somewhat questionable. In addition, the hypothetical CICS workload introduced in this paragraph will never page or swap since paging and swapping are the result of memory contention from competing processes rather than any characteristic of the CICS application itself. Unfortunately, few CICS systems execute as stand-alone applications on dedicated processors.

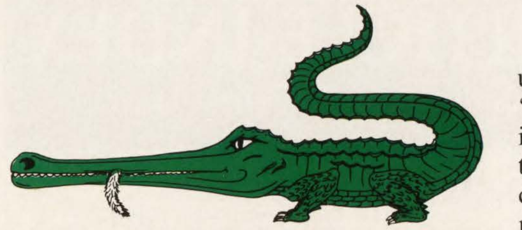
While most installations mark CICS non-swappable, CICS is subject to demand paging as the memory requirements of the processor's other workloads increase. As more and more of the system's CPU resource is consumed by servicing page faults, the percentage of total CPU utilization measured by SMF and RMF tends to decrease. Hence, the concept of a silver bullet capture ratio has to assume not only an average I/O intensity for a workload, but also an average memory demand for the system as a whole. Since it is unlikely that your system will mimic all of these assumed average conditions, the plausibility of published capture ratios for typical types of workloads is subject to great concern. While my experiences have shown that these published values are reasonable within perhaps 20 percent, they simply do not offer the resolution that most analysts want for detailed capacity planning or accounting studies. Moreover, the ratio of used to reported CPU time for an application even varies within an installation as the system's memory demands increase or decrease during a day.

Thus, estimating the path lengths (that is, CPU times) for page, swap and I/O operations appears to be a more realistic approach to attributing unallocated CPU overhead than use of capture ratios.

Estimating Path Lengths

For MVS/XA 2.2 systems, path lengths for page, swap and EXCP operations can be estimated using the RMF CPU Activity (Type 70) and RMF Workload Activity (Type 72) records. These records are created at user defined intervals, typically every 15 to 20 minutes. For each interval, a CPU activity record as well as one Workload Activity Record for each workload defined by the system's Installation Performance Specification (IPS) are generated by RMF.

Prior to MVS/XA 2.2, RMF Workload Activity Records (RMF Type 72) pro-



vided measures of TCB and SRB CPU time as well as EXCP and swap counts for each workload in the system. However, the workload activity records did not include page counts. While page counts were provided by SMF records, they were of little benefit because the SMF records were difficult or impossible to synchronize with the RMF records since they are generated at job and/or step termination rather than fixed intervals like the RMF records. The addition of page counts to the Workload Activity Records in MVS/XA 2.2 facilitated the estimation and application of path lengths for page, swap and EXCP operations for MVS workloads.

Using the data provided by the RMF Type 70 and 72 records, path lengths may be estimated using a technique known as multiple regression. For each measurement interval, the following data elements are selected from the Type 70 record: system identifier, interval start time and date and total CPU busy time.

In addition to these variables from the Type 70 record, the following variables are summarized from all of the Type 72 records for the interval: system identifier, interval start time and date, total SRB CPU time, total TCB CPU time, total pages, total EXCPs and total swaps.

These files are then merged to create a single file comprised of the following data elements: system identifier, interval start time and date, unallocated CPU time (that is, total CPU time minus total TCB plus total SRB time), total pages, total EXCPs and total swaps.

This file can be analyzed using multiple regression to determine the CPU time required for page, swap and EXCP operations.

Appendix I provides SAS code compatible with Merrills' MXG package that employs SAS PROC REG to perform the regression analysis. A sample report generated using this software for MVS/XA operating under VM on a 3090 300E processor is shown in Figure 1 based on 15-minute RMF interval data. As can be seen in the figure, the regression estimated that a swap operation required 0.052 seconds of CPU time and that EXCP and page operation required 0.0005 and 0.0001 seconds respectively. These values may be found in the column labeled "Parameter Estimates" in the figure. The intercept value, 47.16 seconds, represents the CPU overhead like dispatching that could not be explained by the proposed model. While 47.16 seconds initially seems somewhat large, the reader must note that this is less than two percent of the 2,700 seconds of CPU time that were available from the three processors during the 15-minute interval. For best results, a minimum of a week of RMF interval data should be provided to the analysis routine. Moreover, the analysis should be repeated periodically to validate and refine your path-length estimates.

Once the number of CPU seconds required by page, swap and EXCP operations has been estimated, the unaccounted CPU time used by a workload can be determined by multiplying the page, swap and EXCP counts recorded by the respective path-length estimates. When this value

See Capture Ratios *page 94*

MAINFRAME JOURNAL'S ARTICLE INDEX

To receive articles from the selected listing below, fill out the form in the lower right-hand corner.

148. VSAM Catalogs Concepts (*J. Clark, Jul 89, pg 64*)
149. VSAM Optimization & Design (Part 1) (*E. Hudders, Jan 89, pg 72*)
150. VSAM Optimization & Design (Part 2) (*E. Hudders, Feb 89, pg 78*)
151. VSAM Optimization & Design (Part 3) (*E. Hudders, Mar 89, pg 80*)
152. VSAM Performance & Tuning Considerations (*R. Larson, Jul/Aug 87, pg 61*)
153. VSAM Sort Work Areas (*T. Spaulding, Mar/Apr 88, pg 44*)
154. VSAM Specification and Tuning (*F. Bereznay, May/Jun 88, pg 22*)
155. Why Tune VSAM? (*S. MacKinney, May/Jun 87, pg 34*)*

VTAM

156. Basics Of ACF/VTAM Logmode Tables (*L. Hagemo, Jr., Nov/Dec 88, pg 37*)
157. Converting BTAM Driven 3270s to Non-SNA VTAM (*J. Baugher, Mar/Apr 87, pg 54*)*
158. GCS Allows ACF/VTAM To Run Under VM (*L. Hagemo, Jr., Nov/Dec 87, pg 17*)
159. How VTAM Works (*B. Weaver, Apr 89, pg 48*)

3090 Series

160. 3090 Expanded Storage: Evolution Of A New Technology (*W. Mullen, Apr 89, pg 8*)
161. 3090 PR/SM Feature Provides Configuration Flexibility (*P. Artis, Sep/Oct 88, pg 34*)

4300 Series

162. A 4300 Update Management Software Analyzing Hype & Hope (*L. Bergstrom, Apr-Jun 86, pg 52*)*
163. Analyzing The Impact Of IBM's Recent 4381 Announcements (*K. Harvey, Apr-Jun 86, pg 56*)*
164. Are Your 4300 "Rates Great" Or Do Users Shout "Less Billing?" (*Bergstrom/Capelli, Jul-Sep 86, pg 19*)*
165. Despite Popular Opinion, MVS Can Run Well on 4341s (*J. Bebeau, Jan/Feb 87, pg 30*)*
166. IBM Announces New High-End 4381 Models (*Jul/Aug 87, pg 64*)
167. Improving Response Time, Or Making The 4300 Dance (*T. Georgens, Oct-Dec 86, pg 21*)*

9370 Series

168. 9370 Enhanced To Provide Increased Connectivity (*Sep/Oct 87, pg 42*)
169. 9370: Computer Room In A Rack (*P. Mugge, Mar/Apr 87, pg 42*)*
170. 9370: Mini-Mainframe With Maxi-Connections (*E. Sterling, Jan/Feb 88, pg 24*)
171. IBM's New 9370 Series: An In-Depth Analysis (*T. Foth, Jan/Feb 87, pg 40*)*

Application Development

172. Application Development: A Guide Through The Maze Of Tools & Methods (*Tinnirello/Lorentzen, Feb 89, pg 58*)
173. COBOL II: Its Differences And Idiosyncrasies (*H. Bookman, May 89, pg 40*)
174. Change Control & Configuration Management (*L. Stanton, Jan 89, pg 56*)
175. Fourth-Generation Language Dilemma (*A. Lee, Oct-Dec 86, pg 16*)*
176. Guidelines For Selecting Productivity Tools For Development & Maintenance (*M. Fey, Apr 89, pg 69*)
177. ISPF Spells Productivity (*J. Pearkins, Nov/Dec 88, pg 29*)
178. ISPF Techniques (*J. Pearkins, Feb 89, pg 36*)
179. Improving Mainframe COBOL/CICS Programmer Productivity (*Tiernam/Danker III, Jul 89, pg 73*)
180. Improving Software Maintenance Attitudes (*P. Tinnirello, Nov/Dec 87, pg 66*)
181. Quantity Is No Substitute For Quality (*A. Lee, May/Jun 87, pg 36*)*
182. Reengineering: Buzzword or Business Advantage? (*K. Petzold, Jun 89, pg 98*)
183. Screen Generation Software Surfaces As A Real Treasure (*D. Shideler, Jul/Aug 87, pg 13*)
184. Software Maintenance: Undermanaged & Understaffed (*J. Snyders, Jul/Aug 88, pg 30*)
185. The Eight Myths Of Measuring Software Development Performance (*J. Johnson, May 89, pg 29*)
186. What Programmers Don't Know And Why They Don't Know It (*H. Bookman, Mar 89, pg 86*)
187. Why COBOL Efficiency Still Makes Sense (*H. Bookman, Feb 89, pg 22*)
188. Why Prototype? (*J. Pearkins, Apr 89, pg 44*)

CASE

189. CASE And The Balkanization Of Systems Development (*D. Mohlstrom, Jun 89, pg 12*)
190. CASE: A Technology In Transition (*P. Privateer, Jun 89, pg 61*)
191. Selecting CASE Technology And Software (*H. Miller, Jul/Aug 88, pg 63*)
192. Understanding CASE Technology (*H. Miller, May/Jun 88, pg 44*)
193. Washington State University Adopts CASE (*M. Graham, Jun 89, pg 74*)

DBMS

194. Relational Technology . . . Is It For You? (*J. Fair, Sep/Oct 88, pg 56*)
195. CICS And DB2: A Marriage Made In Heaven? (*Shane/Cook, Nov/Dec 87, pg 56*)
196. DB2 In An IMS World (*J. Olson, Apr 89, pg 82*)
197. DB2 Reorgs? Don't Waste Your Time (*B. Cunningham, Feb 89, pg 48*)
198. DB2 Version 2 Offers Major Functional & Performance Enhancements (*J. Goldstein, Sep/Oct 88, pg 43*)
199. DB2's Query Management Facility: One User's Experience (*M. Cater, Feb 89, pg 30*)
200. DB2: Do You Really Want To Use Dynamic SQL? (*J. Goldstein, May/Jun 88, pg 30*)
201. Design, Performance & Capacity Planning For DB2 Application (*J. Goldstein, Nov/Dec 88, pg 56*)
202. Large DB2 Buffer Pools May Hurt On-line Transaction Performance (*J. Goldstein, Mar 89, pg 49*)
203. Should DB2 Be Your DBMS Of Choice? (*J. Goldstein, May 89, pg 52*)
204. Using DB2: A User's Perspective & Experience (*D. Pollock, Jul/Aug 88, pg 10*)
205. IMS Fast Path Data Entry Database Tuning (*G. Rygmyr, Jun 89, pg 46*)
206. IMS/VS Performance Considerations (*J. Fair, May/Jun 88, pg 66*)
207. Performance Issues In The MSC Network Environment (*Green/McGill, Jan/Feb 88, pg 62*)
208. SQL/DS In Full Production (*R. Herbst, et al, Jun 89, pg 80*)
209. Strategies For Improved SQL/DS Performance (*Smith/Spund, Nov/Dec 88, pg 68*)

Operations

210. Automated Job Scheduling (*J. Williams, Nov/Dec 87, pg 12*)
211. Electronic Report Distribution (*H. Miller, Sep/Oct 88, pg 74*)
212. Making Automated Operations A Win-Win Proposition (*B. Carico, May 89, pg 71*)
213. Operations Automation: A Managerial Discussion (*G. Cogicchia, Nov/Dec 88, pg 88*)
214. Planning For Unattended Data Center Operation (*H. Miller, Jan/Feb 88, pg 10*)
215. Unattended Computer Center Operation: 50 Questions & Answers (*H. Miller, Apr 89, pg 58*)

Storage Management

216. CKD versus FBA DASD Comparison (*S. O'Donnell, Jul-Sep 86, pg 44*)*
217. Cache Management: Avoid The Pitfalls (*Goosmann/Nethercote, Jul 89, pg 80*)
218. Caching Controllers Improve Computer System Performance (*D. Corburn, Sep/Oct 88, pg 87*)
219. Control DASD Growth Through Effective Storage Management (*S. Adil, Nov/Dec 88, pg 40*)
220. Controlling The DASD Dollar (*J. Kador, Mar/Apr 87, pg 16*)*
221. DASD Management: One Size Fits All? (*L. Boyle, Jan/Feb 88, pg 50*)
222. FBA Is Here To Stay (*K. Harvey, Jan/Feb 87, pg 16*)*
223. Improvements In Architecture & Technology Result In New Storage Hierarchy (*F. Moore, Sep/Oct 87, pg 15*)
224. Key Compression: Is It Ruining Your DASD Utilization (*D. Martin, Sep/Oct 88, pg 25*)
225. Poor Man's DASD Management System (*W. Meriwether, Jan 89, pg 42*)
226. Solid State Disk Is Making DASD Tuning A Thing Of The Past (*D. Corburn, Mar 89, pg 62*)
227. Storage Management's Relationship to DFSMS (*S. Adil, May/Jun 88, pg 74*)
228. Storage Systems Planning: The Competitive Edge (*E. Nieves, Mar/Apr 88, pg 42*)

Disaster Recovery

229. Are Fully-Equipped Disaster Recovery Facilities The Answer? (*C. Hoag, Jul/Aug 87, pg 18*)
230. Banking Consortium Solves Disaster Recovery Dilemma (*B. Rosenthal, Feb 89, pg 63*)
231. Checklist For Disaster Recovery Planning & Testing (*K. Chang, Sep/Oct 88, pg 91*)
232. Contingency Planning: Where To Begin (*R. Koch, Mar/Apr 87, pg 30*)*
233. Disaster Recovery For 4300 Sites (*C. Lowrie-Skains, Apr-Jun 86, pg 34*)*
234. Network Recovery Alternatives: Complete Or Partial Backup? (*N. Harris, May/Jun 87, pg 41*)*
235. What Happens To Your Business When Your Network Goes Down? (*S. Steinmetz, Jan/Feb 88, pg 36*)
236. Your Legal Liability In A Data Processing Disaster (*R. Miller, Nov/Dec 87, pg 70*)

Connectivity

237. Apply Large Network Management Strategies To Small Networks (*S. Steinmetz, Jul/Aug 88, pg 68*)
238. Gateway Systems Solve Connectivity Problems (*E. Sterling, May/Jun 87, pg 30*)*
239. How to Use SNA On X.25 Networks (*C. Gallucci, Jan-Mar 86, pg 44*)*
240. IBM Token Ring Considerations (*A. Weissberger, Jan-Mar 86, pg 37*)*
241. PC Connectivity: A DP Manager's Guide (*E. Sterling, Jan/Feb 87, pg 20*)*
242. PC Connectivity: A Guide To PC File Transfer Protocols (*E. Sterling, Mar/Apr 87, pg 22*)*
243. SNA Network Management Using NetView & NetView PC (Part 1) (*Ranade/Sackett, May/Jun 88, pg 52*)
244. SNA Network Management Using NetView & NetView PC (Part 2) (*Ranade/Sackett, Jul/Aug 88, pg 26*)
245. SNA's Logical Unit Type 6.2 (*D. Peterson, Sep/Oct 88, pg 12*)
246. TCP/IP: A Well-Connected Protocol (*E. Sterling, May/Jun 88, pg 40*)

Product Reviews

247. Product Review: The Information System Series (Computer Resources Group) (*J. Glick-Smith, Mar 89, pg 42*)
248. Product Review: CS Monitor (Boole & Babbage) (*J. Kador, Jun 89, pg 90*)
249. Product Review: DOCS (Smartech Systems) (*J. Kador, Jan 89, pg 76*)
250. Product Review: Delta IMS Virtual Terminal (BMC Software) (*J. Kador, May/Jun 88, pg 58*)
251. Product Review: EPIC/VSE & EPIC/CMS (Tower Systems) (*J. Kador, Jul/Aug 88, pg 76*)
252. Product Review: Fast Dump Restore (Innovation Data Processing) (*J. Kador, Jan/Feb 88, pg 42*)
253. Product Review: SAS (SAS Institute) (*J. Kador, May 89, pg 89*)
254. Product Review: SYSD (H&W Computer Systems) (*J. Kador, Apr 89, pg 86*)
255. Product Review: SyncSort (SyncSort) (*J. Kador, Mar/Apr 88, pg 16*)
256. Product Review: The Monitor For CICS (Landmark Systems) (J. Kador, Nov/Dec 87, pg 76)
257. Product Review: V/SEG (VM Systems Group) (*J. Kador, Sep/Oct 87, pg 27*)
258. Product Review: VM Applications Planner (BGS Systems) (*J. Herschel, Jan/Feb 88, pg 48*)
259. Product Review: VMOPERATOR (VM Software) (*J. Kador, Nov/Dec 88, pg 100*)
260. Product Review: VSAM/EASY (MB Associates) (*E. Morgan, Apr 89, pg 79*)
261. Product Review: Vital Signs (BlueLine Software) (*J. Kador, Sep/Oct 88, pg 46*)

User Experiences

262. Catalina Marketing Corp.: Electronic Coupon Distribution System Supported By 9370 & VM (*E. Myers, Sep/Oct 88, pg 64*)
263. Corning Glass Works: ES/9370 & VM Generate Computer Simulations (*S. Quanrud, Feb 89, pg 34*)
264. Dorr-Oliver Installs Sophisticated DBMS & 4GL (*J. Kador, Oct-Dec 86, pg 24*)*
265. First Atlantic Savings: A One Terminal Per Desk Approach (*P. Sherline, Jan/Feb 87, pg 66*)*
266. Foster & Gallagher: Direct Efforts Pay Off For A "Growing Business" (*R. Kelley, Jul/Aug 87, pg 62*)*
267. Internal Revenue Service: BARR/RJE Provides Fast Printing (*S. Taylor, Jul 89, pg 77*)
268. James River Corp: Facilitating End-User Computing With VM (*C. Lowrie-Skains, Apr-Jun 86, pg 28*)*
269. Joy Technologies Rides Out Triple Conversion (*L. Wood, Nov/Dec 88, pg 72*)
270. Legi-Slate, Inc.: The Search For Fixed-Cost Computing (*J. Main, Oct-Dec 86, pg 58*)*
271. Lincoln National: Sessioning Software Boosts Claim Processing (*D. Stone, Jul/Aug 87, pg 66*)*
272. Loyal American Live Insurance: VSE To MVS Conversion (*L. McClain, Oct-Dec 86, pg 46*)*
273. Murphy Oil Saves Time With An Electronic Spreadsheet (*Griffin/Lloyd, Jan-Mar 86, pg 49*)*
274. NOREX Reduces Processing Time With VSAM Replacement (*J. Kador, Mar 89, pg 31*)
275. North American Philips: Solutions Center Widens Spectrum Of Computing (*J. Kador, Jul/Aug 87, pg 10*)
276. Reading & Bates: Networking Offshore Data To A Midwest 4381 (*C. Mersch, Jan/Feb 88, pg 60*)
277. Reichert-Jung: Productivity Up 25-30% In Accounts Payable & Purchasing (*J. Cogovan, Jan/Feb 87, pg 24*)*
278. S & A Restaurant Corp: 4381s, MVS & VM Support Aggressive Growth (*C. Lowrie-Skains, Jul-Sep 86, pg 32*)*
279. San Bernardino County's Checklist For Change Control (*J. Hernandez, Jun 89, pg 20*)
280. Schwinn Bicycle's Come-From-Behind Win (*C. Lowrie-Skains, Jan-Mar 86, pg 28*)*
281. TRI-STAR Gains "Picture Perfect" Financial Planning (*N. Ford, Mar/Apr 88, pg 60*)
282. Texaco Links Diverse Computers With Fiber Optics & Twisted Pair (*P. Sherline, Mar/Apr 88, pg 58*)
283. Texas-New Mexico Power: GFIS Improves Data Management & Graphic Reporting (*C. Perry, Jul 89, pg 32*)
284. Texasgulf Chemicals Improves Productivity By Switching From VM (*J. Kador, Mar/Apr 87, pg 36*)*
285. Transworld Oil Uses Telex System To Save Time & Money (*J. Kador, May/Jun 87, pg 14*)*

**Articles from 4300 QUARTERLY or 4300 JOURNAL prior to the merger into MAINFRAME JOURNAL*

Article Order Form

Name ______________________________

Company ______________________________

Address ______________________________

City-State-Zip ______________________________

INDICATE DESIRED ARTICLE #'s ______________________________

____

NOTE:

The price of each article is $2.50, minimum of four (4) per order. A CHECK MUST ACCOMPANY EACH ORDER.

MAINFRAME JOURNAL, P.O. Box 551628, Dallas, TX 75355-1628

Managing Memory-Resident Data In CICS Applications

By David Nicolette

Some CICS applications can benefit from maintaining frequently-referenced data in memory rather than on databases or files. This is true when the resulting performance improvement is worth the additional design and coding effort and the virtual storage resources of the production environment make it feasible.

In an ESA environment, the contents of databases or files can be loaded into data spaces where they can be accessed easily. Application programs are coded with conventional database calls or file I/O statements and operating system services provide access to the data residing in memory.

Most CICS installations are not running ESA at this time and do not have access to data spaces. Many CICS systems run in small- to medium-sized installations where there are no plans to upgrade to ESA within the next several years, if ever. In these environments, when data must be maintained in memory the application programs must contain the logic required to manage the resident data areas.

This article discusses considerations for managing memory-resident data in CICS applications and compares three popular methods of doing so: Temporary Storage (TS) main, the LOAD command with the HOLD option and a user-acquired area in the CICS shared subpool.

The main benefit of maintaining data in memory is the reduction of I/O overhead. Any application that frequently accesses a particular database or file to pick up a small amount of information, in the nature of a table look-up operation, could eliminate considerable I/O overhead by keeping that data in memory.

The determination that an application can benefit from managing resident data areas is best made during the design phase. A large scale rewrite undertaken as a reaction to unanticipated poor performance is usually not cost effective.

If the application is to store a large amount of data in memory, then the increased virtual storage usage may have implications for the paging rate and CPU resource utilization. Coordination with systems personnel is recommended before implementing an application that allocates a large portion of virtual storage.

Figure 1 is a summary of the key design considerations for the three methods of managing resident data described in this article.

The Examples

The examples presented in this article are based on the same data table, programming language and CICS release throughout. This enables you to focus clearly on the differences in the required processing logic for the three methods.

The data table used in the examples contains 50 entries. Each entry consists of two fields: a five-byte table argument and a 20-byte table function. Programs know the five-byte argument value and use it to search for the 20-byte function value.

The examples assume the data to be loaded into memory resides on a VSAM KSDS. Each KSDS record contains the data for a single table entry. The exception is the LOAD/HOLD technique for which the data table is coded as a program.

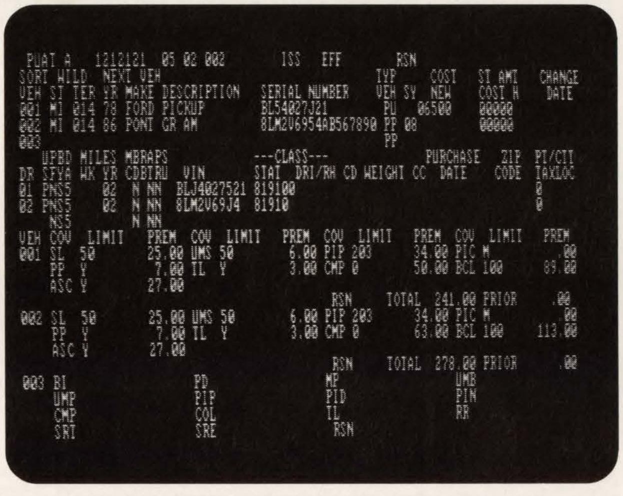
The program code segments used in the examples are written in VS COBOL II, except the LOADed module which is

COMPUWARE CICS SIMULCAST

Now You Can Show Instead Of Tell.

Go ahead: try to describe the problem on this screen to someone who can't see it. Until now, that's been one of the big problems with the assistance you receive from Help Desk support.

Fortunately, CICS SIMULCAST from Compuware eliminates this trial-and-error style of trouble-shooting. Now both the troubled user and technical support can look at the same screen at the same time and get to the bottom of what went awry. With CICS SIMULCAST, screen images can be sent from one terminal to one or more terminals in a CICS region, regardless of the distances between them.



This simple concept provides you some extremely profound bottom-line benefits. Because "seeing is believing," your help desk staff can confirm user problems more quickly and resolve them faster. And the support team can join in with a simple conference command. What's more, by using CICS SIMULCAST to send complete training sessions and demonstrations to remote user sites, you'll be able to reduce travel and training time with this new tool as well.

CICS SIMULCAST works well either as a stand-alone or in tandem with Compuware's CICS PLAYBACK quality assurance and testing procedures. For more information on this eye-opening new product, write or call us soon.

COMPUWARE®

Because experience is everything.

31440 Northwestern Highway, Farmington Hills, MI 48018, 1-800-521-9353. In Michigan, call (313) 737-7300.

CIRCLE #67 on Reader Service Card ▲

F I G U R E 1

Notes:
(1) Any resident data area can be accessed remotely if a synchronous LU Type 6.1 or 6.2 application is developed to provide such access. However, this may cancel out some of the performance benefits of keeping data resident and will increase the complexity of the application.
(2) Main TS queues are accessible from remote CICS regions under MRO (same processor complex) if both regions are at the same CICS release level. Also see Note 1.

Comparison Of Three Methods Of Managing Memory-Resident Data In CICS Applications

	TS Main	Multiple Load Hold Commands	Shared Subpool
PLTPI program required for initialization	Yes		Yes
Recompilation required to update data		Yes	
Remote access is supported (1)	Yes (2)		
Uses common work area			Yes (3)
Data can reside above the line	Yes	Yes (4)	Yes (5)

(3) The pointer to the shared area can be stored in a main TS queue if use of the CWA is not convenient. Performance will be better if the CWA is used.
(4) LOADed modules can reside above the line if they are linkedited AMODE(31) RMODE(ANY). Supported languages are VS COBOL II, PL/1 5.1 or later and Assembler H.
(5) An area allocated by a GETMAIN command will reside above the line if the FLENGTH option is used, at least 4K of storage is requested, and the requestor is executing in 31-bit mode. Requests for less than 4K may be allocated from above the line if FLENGTH is used. Otherwise, the storage is always allocated from below the line.

F I G U R E 2

Loading A TS Queue From A VSAM KSDS
This Logic Is Performed During Post Initialization

```
WORKING-STORAGE SECTION.
01  KSDS-RIDFLD                            PIC X(5) VALUE LOW-VALUE.
01  DATA-TABLE.
    05  TABLE-ENTRY OCCURS 50 INDEXED BY TABLE-INDEX.
        10  TABLE-ARGUMENT                 PIC X(5).
        10  TABLE-FUNCTION                 PIC X(20).
PROCEDURE DIVISION.
    EXEC CICS STARTBR
        DATASET         (KSDSNAME)
        RIDFLD          (KSDS-RIDFLD)
        GTEQ
        RESP            (EIBRESP)
        END-EXEC.
    IF  EIBRESP EQUAL DFHRESP (NORMAL)
        NEXT SENTENCE
    ELSE
        ** ERROR **
    PERFORM
        VARYING TABLE-INDEX FROM 1 BY 1
        UNTIL EIBRESP NOT EQUAL DFHRESP (NORMAL)
           OR TABLE-INDEX GREATER THAN 50
        EXEC CICS READNEXT
            DATASET         (KSDSNAME)
            RIDFLD          (KSDS-RIDFLD)
            INTO            (TABLE-ENTRY (TABLE-INDEX))
            RESP            (TD-RESP)
            END-EXEC
        MOVE TABLE-ARGUMENT (TABLE-INDEX) TO KSDS-RIDFLD
    END-PERFORM.
    IF  EIBRESP EQUAL DFHRESP (ENDFILE)
        NEXT SENTENCE
    ELSE
        ** ERROR **
    EXEC CICS WRITEQ TS MAIN
        QUEUE           (TSQNAME)
        FROM            (DATA-TABLE)
        ITEM            (1)
        NOHANDLE
        END-EXEC.
```

coded in Assembler. The programs could be written in any programming language supported by the CICS command level interface. VS COBOL II was chosen because it is concise and resembles pseudocode, so it should be understandable to any CICS programmer regardless of the programming language used in his or her environment.

The examples are based on CICS Release 1.7, but the methods demonstrated are valid for other releases. For earlier releases, the INQUIRE and SET commands and the RESP option for handling exception conditions are not available. Equivalent facilities are available, however.

Temporary Storage (TS) Main

A main TS queue is well suited to maintaining a small-to-moderate amount of data in memory. It has the advantages of being simple to code and well understood by most CICS application programmers, as well as providing good performance when used appropriately.

A CICS system can be made to use only main TS resources by coding TS=(,0) in the SIT. However, this is rarely done. Rather than relying on the system default, the MAIN option should be coded on the WRITEQ TS command to ensure a main TS queue is used. Otherwise, you may simply be trading application file I/O overhead for DFHTEMP I/O overhead.

TS resources are intended for storing small amounts of data for short periods of time; hence the term ''temporary.'' Therefore, the use of TS queues to store data throughout CICS execution should be kept within reasonable limits.

In an extended addressing environment, CICS allocates storage for main TS queues from OSCOR above the line rather than from the Dynamic Storage Area (DSA).

Even when main TS queues are used, the Control Interval (CI) size defined for the DFHTEMP dataset is a performance consideration. A pointer to each queue item in a CICS system resides in the control subpool of the CICS DSA below the line. If the amount of data written to a queue item is greater than the DFHTEMP CI size, then more than one CI and more than one pointer is used. System-wide, there is an upper limit of 32,767 TS CIs. This sounds like quite a bit, but some CICS customers have exceeded this limit.

Consider the data table containing 50 entries in the examples. If you load this

For CICS/VSAM Data Recovery, Integrity Solutions is the Global Choice.

These leading IBM mainframe companies trust the integrity of their corporate data to Integrity Solutions. They selected Integrity Solutions' DATA RECOVERY SYSTEM (DRS) because it provides the most comprehensive forward and backward recovery for files accessed by CICS/VS and VSAM batch processing.

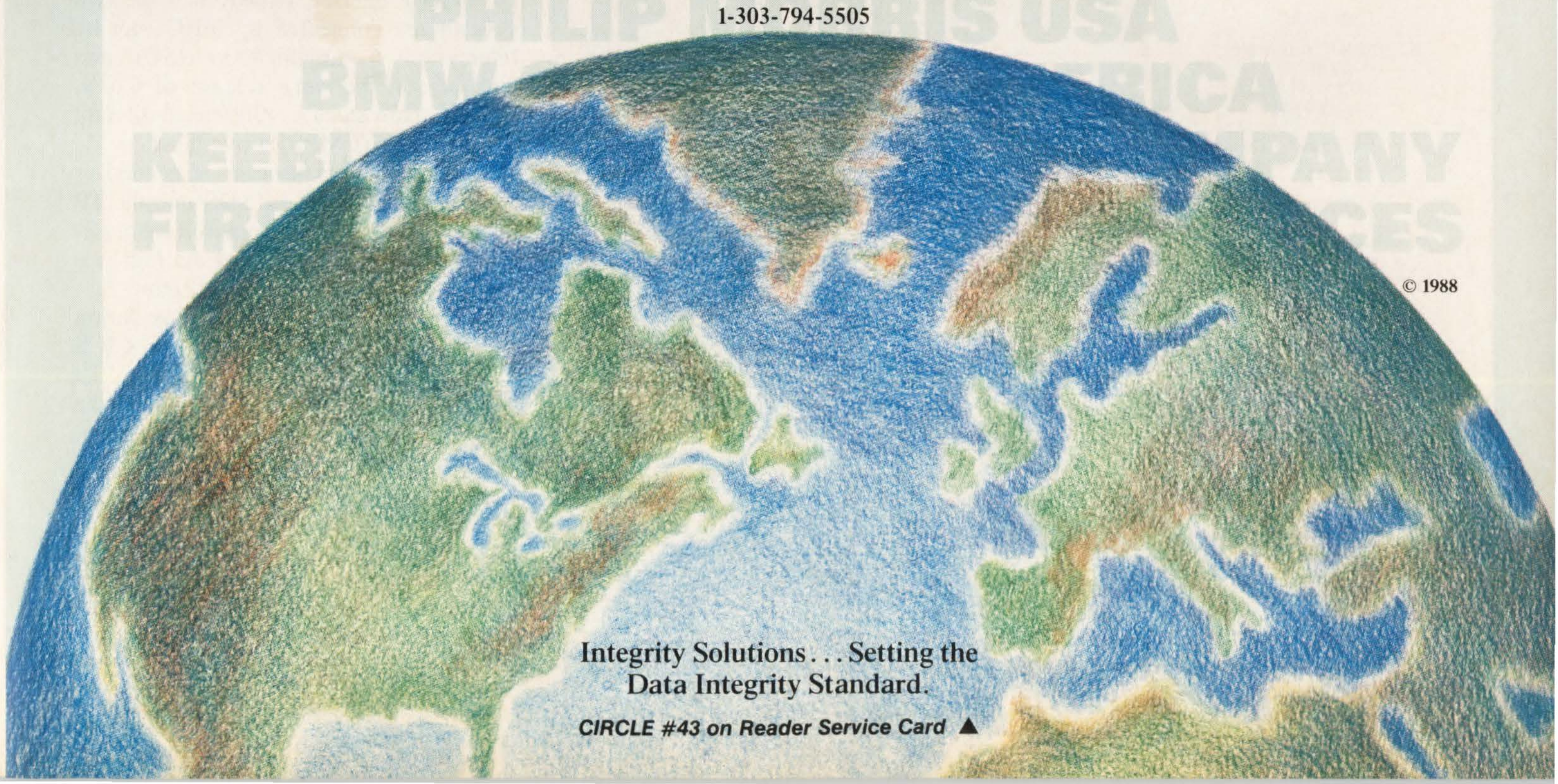
Integrity Solutions introduced DRS more than 8 years ago, and with over 1000 product installations worldwide, Integrity Solutions will continue setting the data integrity standard. By providing leading-edge products and the highest level of support.... to serve your data recovery needs now and in the future.

Guided by what users say they need next, Integrity Solutions is setting the data integrity standard for the emerging electronic vaulting and mirror processing marketplace as well.

The DATA RECOVERY SYSTEM from Integrity Solutions, because your data is worth it.

Integrity Solutions, Inc.

1-800-289-9900
1-303-794-5505



Integrity Solutions... Setting the Data Integrity Standard.

CIRCLE #43 on Reader Service Card ▲

FIGURE 3

Temporary Storage Table (TST) Entry Defining A Remote Queue

```
        DFHTST  TYPE=REMOTE,         IDENTIFIES ENTRY FOR REMOTE        X
                DATAID=TSQNAME,      LEADING CHARS OF QUEUE NAME        X
                SYSIDNT=SYSA,        SYSIDNT OF OWNING SYSTEM           X
(1)             RMTNAME=ABCDEF       QUEUE NAME IN OWNING SYSTEM
```

Notes:
(1) RMTNAME=option is not needed if the queue is known by the same name in both CICS systems.

FIGURE 4

Accessing The TS Queue Read-Only

```
LINKAGE SECTION.
01  DFHCOMMAREA . . .
01  DATA-TABLE.
    05  TABLE-ENTRY OCCURS 50 INDEXED BY TABLE-INDEX.
        10  TABLE-ARGUMENT                  PIC X(5).
        10  TABLE-FUNCTION                  PIC X(20).
PROCEDURE DIVISION.
    EXEC CICS READQ TS
        QUEUE           (TSQNAME)
        ITEM            (1)
        SET             (ADDRESS OF DATA-TABLE)
        RESP            (EIBRESP)
        END-EXEC.
    IF  EIBRESP EQUAL DFHRESP (NORMAL)
        NEXT SENTENCE
    ELSE
        ** ERROR **
```

FIGURE 5

Updating The Data In The TS Queue And In The KSDS LINKAGE SECTION Coding Is The Same As In Figure 4

```
PROCEDURE DIVISION.
    EXEC CICS ENQ
        RESOURCE        (TSQNAME)
        END-EXEC.
    EXEC CICS READ UPDATE
        DATASET         (KSDSNAME)
        • • •
        END-EXEC.
    EXEC CICS REWRITE
        DATASET         (KSDSNAME)
        • • •
        END-EXEC.
    IF  EIBRESP EQUAL DFHRESP (NORMAL)
        NEXT SENTENCE
    ELSE
        GO TO DEQUEUE.
    EXEC CICS READQ TS MAIN
        QUEUE           (TSQNAME)
        ITEM            (1)
        SET             (ADDRESS OF DATA-TABLE)
        RESP            (EIBRESP)
        END-EXEC.
    IF  EIBRESP EQUAL DFHRESP (NORMAL)
        NEXT SENTENCE
    ELSE
        * * ERROR **
*  (MOVE MODIFIED DATA FIELDS TO TS ITEM AREA NOW)
    EXEC CICS WRITEQ TS REWRITE
        QUEUE           (TSQNAME)
        ITEM            (1)
        FROM            (DATA-TABLE)
        RESP            (EIBRESP)
        END-EXEC.
    IF  EIBRESP EQUAL DFHRESP (NORMAL)
        NEXT SENTENCE
```

Figure 5 continued on next page

data into a TS queue designed in such a way that each table entry occupies one queue item, then you will use up 50 CI pointers. When loading the entire table into a single queue item, as shown in Figure 2, you use up only one CI and one pointer, leaving more TS resources available for other purposes.

The overhead required to search the table must also be taken into account in the application design. If the table in the examples were designed with one entry per queue item, then 51 READQ TS commands would be required by each task that accessed the table. With the entire table stored in ITEM(1), only one READQ TS command is required.

The use of TS main has an advantage over the other methods when remote access via Multiple Region Operation (MRO) is required. It is the only one of the three techniques that can provide remote access to data in main memory without requiring additional application development effort.

Access to remote TS resources is accomplished by defining TYPE=REMOTE entries in the Temporary Storage Table (TST). An example is shown in Figure 3. In the example, a TST entry is defined such that a TS queue called ABCDEF on system SYSA can be accessed by an application running in a system other than SYSA by referring to queue name TSQNAME.

When a remote TS main queue is referenced, CICS normally copies the queue to auxiliary TS on the requesting system's DFHTEMP dataset. However, if the two systems are connected by MRO, not by Intersystem Communication (ISC), and both are on the same release of CICS, then the main queue is referenced directly with no DFHTEMP I/O overhead.

An initialization process is needed to load the data from the KSDS into the TS queue before it can be accessed by application tasks. An example of the logic required to do this is shown in Figure 2.

The initialization program runs during the post initialization phase of CICS startup. It is defined in a Program List Table (PLT) which is referenced by the PLTPI= startup override or SIT option. The program browses the KSDS, copying each record into an entry in the data table until all the data has been copied. Then, it issues a WRITEQ TS MAIN command to initialize the queue. From this point on, any task running in the CICS address space can reference the table simply by issuing a READQ TS command.

```
        ELSE
            EXEC CICS SYNCPOINT ROLLBACK
                END-EXEC.
    DEQUEUE.
        EXEC CICS DEQ
            RESOURCE            (TSQNAME)
            NOHANDLE
            END-EXEC.
```

F I G U R E 6

Coding The Data Table As A "Program" For Use With The LOAD Command

```
TABLEPGM   CSECT
           DC       CL5'00101'                       ARGUMENT    1
           DC       CL20'ABCDEFGHIJKLMNOPQRST'       FUNCTION    1
           • • •
           DC       CL5'29050'                       ARGUMENT    50
           DC       CL20'ZZZZZZZZZZZZZZZZZZZZ'       FUNCTION    50
           END
```

F I G U R E 7

Accessing The LOADed Table Program
No Post Initialization Processing Is Needed With This Method

```
LINKAGE SECTION.
01    DFHCOMMAREA . . .
01    DATA-TABLE.
      05    TABLE-ENTRY OCCURS 50 INDEXED BY TABLE-INDEX.
            10    TABLE-ARGUMENT                PIC X(5).
            10    TABLE-FUNCTION                PIC X(20).
PROCEDURE DIVISION.
      EXEC CICS LOAD
            PROGRAM           ('TABLEPGM')
            SET               (ADDRESS OF DATA-TABLE)
            HOLD
            NOHANDLE
            END-EXEC.
```

F I G U R E 8

Replacing LOADed Table Program With A New Version
Method 1: Issue RELEASE Commands Until APCN Abend Occurs

```
PROCEDURE DIVISION.
      EXEC CICS HANDLE ABEND
            LABEL             (CHECK-ABCODE)
            END-EXEC.
REPEAT.
      EXEC CICS RELEASE
            PROGRAM           ('TABLEPGM')
            END-EXEC.
      GO TO REPEAT.
CHECK-ABCODE.
      EXEC CICS ASSIGN
            ABCODE            (TEST-ABCODE)
            END-EXEC.
      IF    TEST-ABCODE EQUAL 'APCN'
            NEXT SENTENCE
      ELSE
            ** ERROR **
      EXEC CICS SET
            PROGRAM           ('TABLEPGM')
            NEWCOPY
            NOHANDLE
            END-EXEC.
```

Figure 4 shows the code required to access the table. There are no special requirements; the READQ TS command is all that is needed.

When the data must be updated during CICS execution, the TS queue item can easily be updated. Figure 5 shows the code required to update both the TS queue and the VSAM KSDS and to keep the updates in sync.

First, the TS queue is enqueued to prevent another task from updating it concurrently. Of course, any other programs designed to update the queue must enqueue on it in the same way in order for the protection to be effective. Both the update of the queue and the update of the KSDS are within the scope of the enqueue. This ensures the updates will be in sync.

The sequence of updating is significant in terms of data reliability. The KSDS is updated first, followed by the TS queue. Assuming dynamic transaction backout is defined for the KSDS, this will keep the data in sync in the event of a transaction abend during the file update. In case the file update cannot be completed due to some exception condition, then the update of the queue can be aborted. If the file update is successful but the queue update fails, then the file update can be backed out.

If the queue were updated first, then a window of vulnerability would exist that could cause the data in the queue to become unsynchronized with the data in the KSDS. Assume the update to the queue is successful, but a transaction abend occurs before the KSDS update has been completed. Deferred Work Element (DWE) processing will release the enqueue but will not back out the queue item update because a "main" queue cannot be defined as recoverable.

In summary, a TS main queue offers several advantages for managing memory-resident data. It is easy to code and maintain, provides efficient processing when used appropriately and allows the data to be updated easily and reliably while CICS is active. In addition, remote access to a main queue is supported when certain conditions are met.

This method becomes undesirable when there is a requirement to maintain a large amount of data in memory. Applications designed to use many queue items or to access queues sequentially make poor use of TS resources and may cause as many performance problems as they seek to solve.

Load Command With Hold Option

The CICS LOAD command loads a link-edited module from a load library (OS) or core image library (DOS) into the program subpool of the CICS DSA. When the HOLD option is included on the

FIGURE 9

Replacing LOADed Table Program With A New Version
Method 2: RELEASE And Test Residence Count

```
WORKING-STORAGE SECTION.
01  WS-RESCOUNT                              PIC S9(8) COMP VALUE +1.
PROCEDURE DIVISION.
    PERFORM
        UNTIL WS-RESCOUNT EQUAL ZERO
        EXEC CICS RELEASE
            PROGRAM      ('TABLEPGM')
            NOHANDLE
            END-EXEC
        EXEC CICS INQUIRE
            PROGRAM      ('TABLEPGM')
            RESCOUNT     (WS-RESCOUNT)
            NOHANDLE
            END-EXEC
    END-PERFORM.
    EXEC CICS SET
        PROGRAM      ('TABLEPGM')
        NEWCOPY
        NOHANDLE
        END-EXEC.
```

FIGURE 10

Allocate And Initialize An Area In The Shared Subpool
This Logic Is Performed During Post Initialization

```
WORKING-STORAGE SECTION.
01  GETMAIN-INITIMG                          PIC X VALUE SPACE.
01  KSDS-RIDFLD                              PIC X(5) VALUE LOW-VALUE.
LINKAGE SECTION.
01  DFHCOMMAREA . . .
01  COMMON-WORK-AREA.
    05  FILLER                               PIC X( . . . ).
    05  CWA-DATA-TABLE-PTR                   POINTER.
01  DATA-TABLE.
    05  TABLE-ENTRY OCCURS 50 INDEXED BY TABLE-INDEX.
        10  TABLE-ARGUMENT                   PIC X(5).
        10  TABLE-FUNCTION                   PIC X(20).
PROCEDURE DIVISION.
    EXEC CICS GETMAIN
        FLENGTH     (1250)
        INITIMG     (GETMAIN-INITIMG)
        SET         (ADDRESS OF DATA-TABLE)
        SHARED
        END-EXEC.
    EXEC CICS STARTBR
        DATASET     (KSDSNAME)
        RIDFLD      (KSDS-RIDFLD)
        GTEQ
        RESP        (EIBRESP)
        END-EXEC.
    IF  EIBRESP EQUAL DFHRESP (NORMAL)
        NEXT SENTENCE
    ELSE
        ** ERROR **
    PERFORM
        VARYING TABLE-INDEX FROM 1 BY 1
        UNTIL EIBRESP NOT EQUAL DFHRESP (NORMAL)
            OR TABLE-INDEX GREATER THAN 50
        EXEC CICS READNEXT
            DATASET     (KSDSNAME)
            RIDFLD      (KSDS-RIDFLD)
            INTO        (TABLE-ENTRY (TABLE-INDEX))
            RESP        (TD-RESP)
            END-EXEC
        MOVE TABLE-ARGUMENT (TABLE-INDEX) TO KSDS-RIDFLD
    END-PERFORM.
    IF  EIBRESP EQUAL DFHRESP (ENDFILE)
```

Figure 10 continued on next page

LOAD command, the loaded module remains resident after the requesting task has terminated.

Storage in the program subpool is allocated in full pages. Therefore, although the sample data table is only 1,250 bytes long, with this technique 4,096 bytes of virtual storage will be occupied. In a virtual storage constrained environment, this could become a consideration if many modules are LOADed. Under normal conditions, it is not a problem.

CICS returns the address of the LOADed module to the requesting program which can then use a pointer reference to access the data. If the module is already resident when a LOAD/HOLD request is processed, then CICS simply returns the address of the module without reloading it. Once the module has been loaded, subsequent LOAD/HOLD commands involve little overhead.

The LOAD/HOLD technique is best suited for data that is rarely, if ever, updated on-line. A CICS task can move new values to the program and the updates will be recognized by other CICS tasks that LOAD the module. However, once CICS terminates, the updates are lost. The only way to make a permanent update to the data is to modify and recompile the program.

For data that does not need to be updated on-line, the LOAD command is the simplest method to code and maintain. No special processing is needed at post initialization, since the first task to issue a LOAD/HOLD command will cause the module to be loaded.

Figure 6 illustrates how the sample data table could be coded as a "program." Any programming language supported by CICS can be used to code such a program, but Assembler is normally used since the SET option of the LOAD command will return the address of the first byte of the data table.

If written in COBOL, the "table" program would contain WORKING-STORAGE code but no executable statements. The PROCEDURE DIVISION would contain only a GOBACK statement. The address returned in the SET option of the LOAD command for a COBOL program is 232 bytes before the start of WORKING-STORAGE.

This is a disadvantage of using LOAD commands. You must either maintain the data table in two languages, Assembler for loading the table and COBOL for accessing it, or you must modify a BLL cell with a hard coded value after each LOAD

```
            NEXT SENTENCE
        ELSE
            ** ERROR **
        EXEC CICS ADDRESS
            CWA             (ADDRESS OF COMMON-WORK-AREA)
            END-EXEC.
        SET CWA-DATA-TABLE-PTR TO ADDRESS OF DATA-TABLE.
```

F I G U R E 11

Accessing Data In The Shared Subpool Read-Only

```
LINKAGE SECTION.
01  DFHCOMMAREA . . .
01  COMMON-WORK-AREA.
    05  FILLER                              PIC X(. . .).
    05  CWA-DATA-TABLE-PTR                  POINTER.
01  DATA-TABLE.
    05  TABLE-ENTRY OCCURS 50 INDEXED BY TABLE-INDEX.
        10  TABLE-ARGUMENT                  PIC X(5).
        10  TABLE-FUNCTION                  PIC X(20).
PROCEDURE DIVISION.
    EXEC CICS ADDRESS
        CWA             (ADDRESS OF COMMON-WORK-AREA)
        END-EXEC.
    SET ADDRESS OF DATA-TABLE TO CWA-DATA-TABLE-PTR.
```

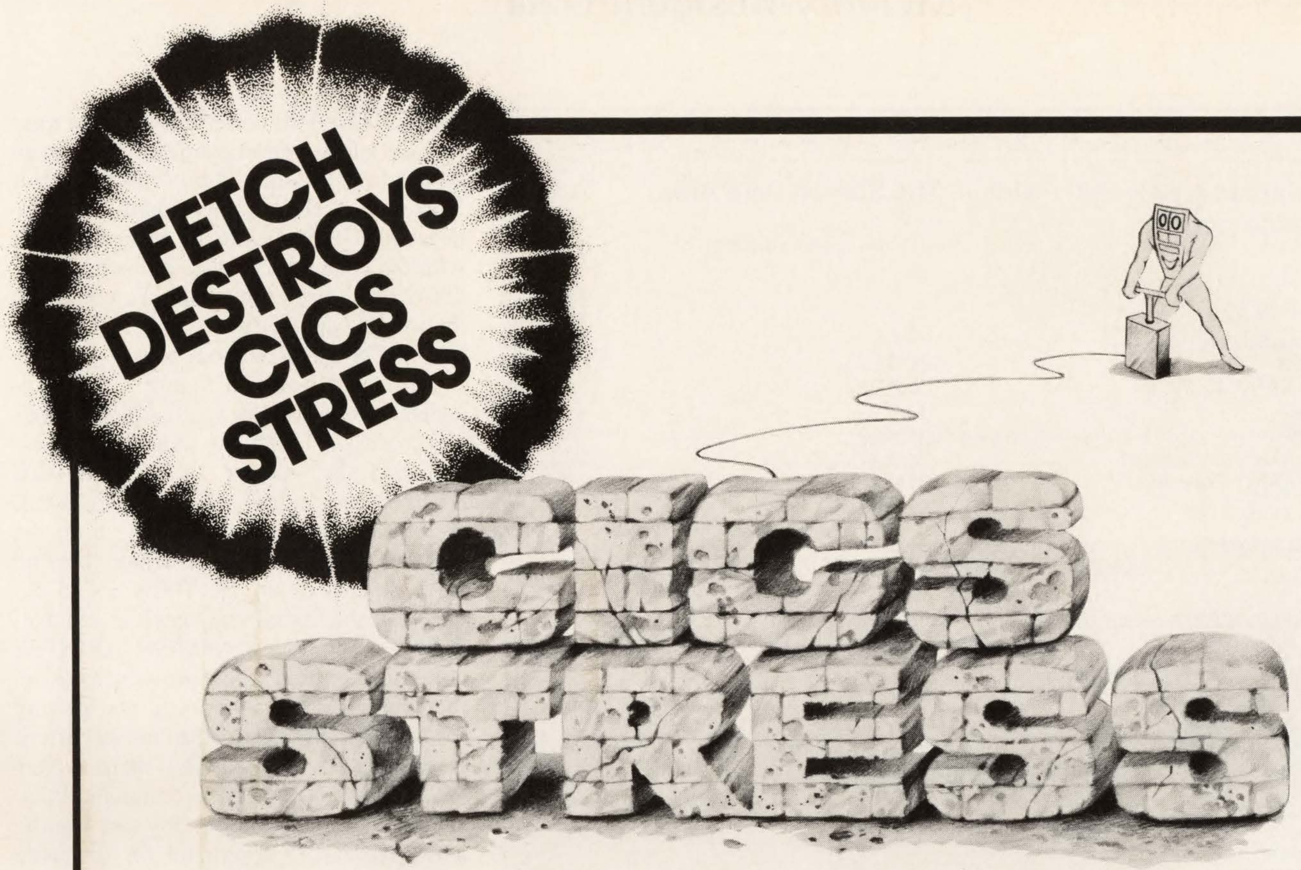
command, which introduces a potential release dependency into the application code. Either approach will probably cause confusion for maintenance programmers.

Figure 7 shows how to access the LOADed program to reference the data table.

The LOAD/HOLD method is poorest when the resident data must be updated while CICS is running. There is a field in the resident PPT entry for each program where a count is maintained of the number of LOAD requests which have been issued for the program. Each LOAD command with the HOLD option causes this field, called the *residence count*, to be incremented by one despite the fact that the program is not actually loaded each time.

The RELEASE command causes the residence count to be decremented by one. When a RELEASE command is issued for a program whose residence count is zero, then the pages occupied by the program in the Program Subpool of the DSA become available for other use.

A program can be New Copied only when its residence count is exactly zero. A New Copy request issued for a program with a non-zero residence count will receive an IN USE response. Therefore, when a program needs to New Copy a new version of the table module, it must first issue RELEASE commands until the residence count has been reduced to zero. The New Copy operation itself does not reload the program, but the next LOAD/HOLD command will load the new version.



FETCH is a powerful solution to the problems that cause CICS stress. FETCH will dramatically improve CICS performance from day one. Just how good is it, really? Well, you could say FETCH is dynamite. Install FETCH on your CICS and get instant relief. Relief from CICS "bottleneck" problems like virtual storage constraints and slow response time.

You see, FETCH goes right to the heart of the matter by satisfying an unlimited number of CICS load requests simultaneously. So FETCH improves your response time right off the bat. Also, FETCH eliminates the need to define high use application programs resident for performance purposes. FETCH's unique multi-thread load mechanism will load programs as quickly as if they were core resident. And, if you're an XA shop, our FETCH/XA product will let you take advantage of address space above the XA line without any program changes.

- Reduce resident program storage requirements by 90% or more.
- Eliminate 87% of program wait on the load library and instantly improve response time.
- Reduce I/O at least 50%.
- Reduce compression problems.

FETCH™ & FETCH/XA™
OPERATES UNDER MVS/SP & XA

AXIOS PRODUCTS, INC.

1455 Veterans Highway
Hauppauge, NY 11788
(516) 348-1900

**FREE
30 DAY
TRIAL**

F I G U R E 1 2

Updating Selected Fields In The Shared Data Area

```
WORKING-STORAGE SECTION.
01    ENQNAME                                  PIC X(6) VALUE 'UPDATE'.
LINKAGE SECTION.
01    DFHCOMMAREA . . .
01    COMMON-WORK-AREA.
      05    FILLER                             PIC X( . . . ).
      05    CWA-DATA-TABLE-PTR                 POINTER.
01    DATA-TABLE.
      05    TABLE-ENTRY OCCURS 50 INDEXED BY TABLE-INDEX.
            10    TABLE-ARGUMENT               PIC X(5).
            10    TABLE-FUNCTION               PIC X(20).
PROCEDURE DIVISION.
      EXEC CICS ADDRESS
            CWA              (ADDRESS OF COMMON-WORK-AREA)
            END-EXEC.
      SET ADDRESS OF DATA-TABLE TO CWA-DATA-TABLE-PTR.
      EXEC CICS ENQ
            NAME             (ENQNAME)
            LENGTH           (6)
            END-EXEC.
*   (UPDATE THE KSDS NOW)
*   (MOVE UPDATED FIELDS TO THE SHARED DATA AREA NOW)
      EXEC CICS DEQ
            NAME             (ENQNAME)
            LENGTH           (6)
            END-EXEC.
```

F I G U R E 1 3

Replacing The Entire Data Area In The Shared Subpool During CICS Execution

```
WORKING-STORAGE SECTION.
01    SAVE-TABLE-PTR                           POINTER.
01    ENQNAME               PIC X(7) VALUE 'REFRESH'.
LINKAGE SECTION.
01    DFHCOMMAREA . . .
01    COMMON-WORK-AREA.
      05    FILLER                             PIC X( . . . ).
      05    CWA-DATA-TABLE-PTR                 POINTER.
01    DATA-TABLE.
      05    TABLE-ENTRY OCCURS 50 INDEXED BY TABLE-INDEX.
            10    TABLE-ARGUMENT               PIC X(5).
            10    TABLE-FUNCTION               PIC X(20).
PROCEDURE DIVISION.
      EXEC CICS ENQ
            NAME             (ENQNAME)
            LENGTH           (7)
            END-EXEC.
      EXEC CICS ADDRESS
            CWA              (ADDRESS OF COMMON-WORK-AREA)
            END-EXEC.
      SET SAVE-TABLE-PTR TO CWA-DATA-TABLE-PTR.
*   (NOW GETMAIN AND INIT NEW AREA AS IN FIGURE 10)
      SET CWA-DATA-TABLE-PTR TO ADDRESS OF DATA-TABLE.
      EXEC CICS DEQ
            NAME             (ENQNAME)
            LENGTH           (7)
            END-EXEC.
      EXEC CICS DELAY
            INTERVAL         (6)
            END-EXEC.
      SET ADDRESS OF DATA-TABLE TO SAVE-TABLE-PTR.
      EXEC CICS FREEMAIN
            DATA             (DATA-TABLE)
            END-EXEC.
```

The RELEASE command does not raise any sort of exception condition to indicate when the residence count reaches zero. However, if the residence count should go negative, CICS abends the requesting task with the code APCN. When this happens, CICS also zeros the residence count, so the module can be New Copied following the APCN abend without error.

The code shown in Figure 8 takes advantage of the way in which APCN abends are processed. First, an abend exit is established by means of a HANDLE ABEND command. Then, RELEASE commands are issued in a loop until the residence count is decremented beyond zero. At this point, an APCN abend occurs and the abend exit receives control. In the example, an ASSIGN ABCODE command is used to ensure you have, in fact, received the abend you are looking for and that a real error has not occurred. If the abend was APCN, then a SET PROGRAM NEWCOPY command is issued. From this point on, any task issuing a LOAD/HOLD command for the table will see the updated version.

Figure 9 shows another method of New Copying a LOADed module. In this example, a RELEASE command is followed by an INQUIRE PROGRAM command to test the residence count. As soon as the residence count reaches zero, the loop is terminated and a SET PROGRAM NEWCOPY command is issued.

In releases of CICS prior to 1.7, the INQUIRE and SET commands are not available. However, the program can LINK to DFHEMTP to perform the New Copy operation using the technique illustrated in Figure 8. The LINK would be done in the abend exit routine after the APCN abend occurs. A string such as 'S PRO(xxxxxxxx) N' is passed in COMMAREA to accomplish the New Copy operation. DFHEMTP returns no information to the caller to indicate the success or failure of the operation.

In summary, the LOAD/HOLD technique is best for relatively static data. The code required to access the data is the simplest of the three methods discussed here, no special post initialization processing is required and access to the data is quite fast.

When the data must be updated on-line with any frequency, the LOAD/HOLD technique becomes the most cumbersome of the three. Another negative aspect of the LOAD technique is that you must either code the table program in Assembler or manipulate the address returned by the SET option of the LOAD command.

Shared Subpool

The shared subpool of the CICS DSA is used for data that remains in memory for an extended period of time and is used by multiple CICS tasks. Data areas allocated from the shared subpool are not associated with any particular task.

The GETMAIN command normally allocates storage from the Task Subpool. Such storage is automatically freed when the requesting task terminates or issues a SYNCPOINT command. When the SHARED option is coded on the GETMAIN command, then the storage is allocated from the Shared Subpool. In this case, the storage is not automatically freed at end of task. It remains allocated until explicitly freed by a FREEMAIN command. The FREEMAIN command can be issued by any task.

The shared subpool is a convenient place to store application defined data in memory. It is easy to access and to update and involves less processing overhead than the TS main technique. However, CICS makes no assumptions and gives you no help with managing data areas in the shared subpool. The application bears full responsibility for allocating, freeing and managing the data areas it requires.

When a GETMAIN SHARED command is processed, CICS returns the address of the allocated area to the requesting program. When the requesting task terminates, the storage remains allocated, but the address is not automatically stored anywhere and there is no CICS command that can obtain the address. It is up to the application to keep track of the address.

Typically, the addresses of such storage areas are maintained in the CICS Common Work Area (CWA). In most installations, CICS systems personnel will assign an offset in the CWA where you can store the address of a shared data area upon request. In most installations, you will not be allowed to store the entire table directly in the CWA for two reasons: (1) it is limited in size to a maximum of 3,084 bytes and (2) the CWA is physically part of the CSA which would be vulnerable to the kind of application program bug that might overwrite it.

As with the TS main queue method, the data from the VSAM KSDS must be loaded into memory during post initialization. Figure 10 shows the code required to do this.

Since you are working with a fixed-length table of 1,250 bytes, the first step in Figure 10 is to GETMAIN 1,250 bytes in the shared subpool. However, if you are loading a table whose length is determined at execution time, you can construct the table in WORKING-STORAGE first, then compute the length for the GETMAIN.

In Figure 10, the GETMAIN command includes the SHARED option to cause storage to be allocated from the shared subpool.

Storage allocated from above the line in response to a CICS GETMAIN request comes from OSCOR. Storage allocated from below the line and all storage allocated in a non-XA environment comes from the CICS DSA. To allocate storage above the line, the program must be executing in 31-bit mode, the FLENGTH option must be used rather than the LENGTH option on the GETMAIN command and the length specified must be at least 4K. The maximum length allowable for the FLENGTH option in 31-bit mode is one megabyte. For programs executing in 24-bit mode, the maximum length allowed for a single GETMAIN is 64K and it will always be allocated from the DSA below the line.

In my example, although VS COBOL II supports extended addressing and the FLENGTH option is coded on the GETMAIN command, the storage for the table will be allocated below the line because the length requested is only 1,250 bytes.

Next, the KSDS is browsed and the records loaded into the table exactly as in the TS queue example in Figure 3.

The last step is to save the address of the table in the CWA where subsequent CICS tasks can find it. An ADDRESS CWA command is used to obtain the address of the CWA itself and then the pointer reference value is saved at the assigned location in the CWA.

Figure 11 shows the code needed for a task to access an area in the shared subpool. First, the CWA is accessed. Then, a pointer reference is set to the address stored in the CWA.

In Figure 12, the data table in the shared subpool is partially updated and the KSDS is updated to keep the data in sync. First, the table is accessed in the same manner as in a read-only access. Then, an ENQ command is issued to prevent concurrent updating of the table or the KSDS. CICS

See Memory-Resident Data *page 92*

VSAM Catalogs Concepts

Understanding the VSAM catalog structure and some of its components

By James W. Clark

VSAM catalogs are an interesting topic and always a source of confusion. When catalogs are mentioned in conversation, most begin speaking of their Sears & Roebuck catalog or some other favorite mail order catalog. Many people are unaware of the "old" VSAM catalog or the Basic Catalog Structure (BCS). In fact, my wife Tracey and two young children had to be drilled constantly for awhile until they understood these important issues. My youngsters now eagerly await their bedtime stories which contain animated pictures of VSAM control blocks and other fascinating catalog concepts. My wife was a little slower to accept these catalogs; she had a much better grasp of the mail order catalogs and clearly understood how they worked. As with any story-telling time, a little history helps, so here goes . . . .

Once upon a time, back in the dark computer ages of 1974, the big blue giant spoke loudly about a new access method called Virtual Storage Access Method (VSAM). This new method provided for new dataset formats, including a new catalog structure to contain these new files. The big blue giant created this old VSAM catalog with a system defined Control Interval (CI) size of 512 bytes.

This ugly layout consisted of high key and low key ranges and an embedded index. The high key range entry contained a 44-byte dataset name and a three-byte CI pointer to the corresponding low key range record. The low key was a fixed 505-byte record that contained all the ugly but necessary items to find these little VSAM file critters now running around in the system. Many of these 505-byte records required far less than this to contain such information. The VSAM catalog was allocated with a Control Area (CA) of three tracks, one of which was for the embedded index.

With a 512 CI size (which many records did not use entirely) and a CA size of three tracks (one used for embedded index), it was easy to see why these catalogs were not only ugly, but also they were kind of chubby if not outright fat. These old catalogs also preformatted internal free slots which were for future entries. This free space was built with a chaining concept that had pointers to the next free slot. Unfortunately, it was not too difficult for the VSAM catalog to have some system crash or interruption when updating this free chain and lost pointers. Additional free space was easily lost and, of course, more space could be consumed since the free space may never have been fully utilized.

Other fun items about this old catalog includes the fact that since the catalog contained all the information about the VSAM files in one fat spot on a DASD volume, if a different volume needed to be restored, the catalog and the VSAM files on the recovered DASD volume were probably out of sync and possibly inaccessible. This could easily have occurred since the VSAM catalog contained the physical extent information (on its DASD volume) and if files went into secondary or were reallocated on to another DASD area, this would cause a mismatch between the catalog and the recovered DASD volume. Recovery of the catalog and VSAM file(s) was difficult when something did break and there was little in the way of tools to assist in this process. This catalog also had an ego problem that caused headaches to many DASD space users and the Direct Access Device Space Management (DADSM) software. This catalog wanted to be in the space allocation business in addition to DADSM.

VSAM catalogs had to "own" the DASD volume if they had any VSAM datasets on that pack. This meant that only one catalog could contain VSAM files on that pack. If another VSAM file were in need of space on a DASD volume, it could only use that DASD space if its catalog "owned" it, regardless of how much space was available. VSAM "ownership" of DASD volumes was a difficult problem for allocating file space.

In addition, DADSM was not too happy about VSAM and its suballocated data space which were fighting for bragging rights to the DASD volume and its Vol-

If...

You thought there were no new alternatives in MVS job scheduling software,

ume Table of Contents (VTOC). After some period of time, the big blue giant went back to his Etch-A-Sketch machine and designed a new software toy that he proudly bellowed was a new VSAM catalog structure. This was to be available, but it was going to cost you. This new VSAM structure was a program product called Data Facility/Extended Function (DF/EF) and it became available in 1981. Since that time, DF/EF and a few other related data management products have been incorporated into one larger product known as Data Facility Product (DFP). This new structure and software was known as Integrated Catalog Facility (ICF). This concludes the history lesson, down the ICF brick road we go!

From The "Old" To The "New"

The restructuring of the old VSAM catalog was drastic. The big blue giant had done a major facelift and it apparently was at the "fat farm;" regardless, a prettier and slimmer catalog was unveiled. Like all major surgery, of those old catalogs who wanted the newest fashion look, some took a battered and bruised beating going through the conversion process (IDCAMS convert catalog utility). I know a couple of my old fat cats were so battered around that they had mistaken 3375 VSAM files as 3380 VSAM files or they found dead fish in their old free space chains. I thought for sure a malpractice suit was coming, but once the old fat cats were converted and recuperated, I was into the VSAM health food from there on.

Integrated Catalog Facility Overview

The old VSAM catalog was broken into two components: the first one was known as the Basic Catalog Structure (BCS) and the second, the VSAM Volume Dataset (VVDS). After a long fatherly talk, VSAM submitted to letting DADSM handle the space management. VSAM would not be into having late night data space parties; thus, all requests for DASD space were to be handled by DADSM. All VSAM files were now *unique* and no petty squabbling among the catalogs as to who owned each DASD volume would be heard. DADSM would let up to 36 different catalogs allocate space on a DASD volume.

The catalog recovery process was to be enhanced in that better tools were made available. A doctor's kit to do DIAGNOSE function was provided and various IDCAMS parameters could be used for small scrapes and bruises. Although the recovery process is better than the previous structure with much less chance of catalog problems in the first place, it still can cause sleepless nights at the thought of repairing it. In a recent release of DFP, a Catalog Address Space (CAS) was created to handle the catalog management modules, control blocks and catalog information. This can be a large virtual storage relief for large ICF catalog users; up to 1MB of virtual storage below the line may be freed for a shop that has a few dozen ICF catalogs. This isolation of VSAM catalog management modules/control blocks into an address space (CAS) also provides MVS MODIFY commands which can list or even close catalogs. An additional recovery aid is the now automatic copying of aliases when doing an IDCAMS EXPORT (backup) function; this relieves the need to redefine such aliases if an IDCAMS import is required to recover the catalog.

The ICF software also allows the BCS catalog to be moved, merged or split apart including different device types. An IDCAMS EXAMINE command is also available to provide a check on the data and/or index component structure for possible errors on a VSAM file.

Another relatively new feature is the RACF (or equivalent) profile definition of a resource named IGG.CATLOCK. The IGG.CATLOCK feature may be used to *lock* or *unlock* the ICF catalog access (for catalog error recovery process). Once the RACF profile has been set up, an IDCAMS function with the LOCK keyword may be issued to *freeze* access to a catalog in need of repair. Those few systems people who need this capability must have the RACF READ capability for IGG.CATLOCK and ALTER capability for the ICF catalog in need of repair. If an IDCAMS function issues a LOCK parameter, all unauthorized users (RACF IGG.CATLOCK checking) will receive a system message that states the catalog is temporarily unavailable. When recovery has been completed, the catalog may be unlocked and made available for access.

Basic Catalog Structure

The first component of an ICF catalog is the BCS. The key word is "basic" because it is similar in function to the old OS Control Volume catalog (CVOL) in that it points to the DASD volume(s) where the entry resides and where more specific information is available in the VTOC or VVDS (described later). The BCS is a VSAM Keyed Sequence Dataset (KSDS). It is much more flexible than the old VSAM catalog in that many KSDS parameters may also be specified for the BCS such as the CI/CA sizes, record size, free space, imbed, replicate and string numbers. For more information on defining a BCS user catalog, see the appropriate VSAM manuals.

The BCS contains variable length records with the SPANNED attribute, thus the CI and record size specification contributes to when records must be spanned. The record size may be specified from 4086 to a maximum of 32400. The BCS records consist of *cells* which represent the smallest logical grouping of entry related data. Many different types of cells exist; these are grouped into the next higher level called a component. A component record may be grouped into a subrecord, and finally, the largest unit is known as a *sphere* record. The actual layout of each of these groups and various types of cells are described in detail in VSAM manuals. The sphere record represents the basic transfer unit for doing I/Os, similar in concept to the CI size being the basic transfer unit for VSAM files. Since the sphere record is logically constructed with the necessary cells, significantly fewer I/Os are required to retrieve dataset catalog information in comparison to the older VSAM catalog.

There are two types of sphere records — a sphere for VSAM and Generation Data Group (GDG) entries and a nonsphere for non-VSAM, alias, user catalog connects and so on.

The BCS variable length and sometimes *spanned* records, coupled with user supplied KSDS parameters (CI/CA sizes, free space, record size and so on) requires much *less* DASD space in comparison to the older VSAM catalog. The GDG entries are handled much better in the BCS than the old VSAM catalog. Since entries may be reused, significant DASD may be saved. Another improvement over the original VSAM catalog is the free chain pointer system that does NOT exist in the BCS (uses regular CI/CA free space). Like other unique VSAM files (no suballocated allowed in an ICF), the BCS may have up to 123 extents. For performance reasons, large amounts of secondary extents are *not* desired.

The names of the BCS catalog are the following.

- BCS Cluster name is a 44-byte dataset name that contains all binary zeros plus a one-byte extension indicator as its KEY. This ensures that the first record on the BCS is its own self-describing record.

- BCS data component name is the 44-byte name that the user specified via the IDCAMS define user catalog statement. The 44-byte name is identical to the FORMAT 1 DSCB in the VTOC on the DASD volume.

- BCS index component name is in the following format: "CATINDEX.Tbbbbbbb.yyddd.T99999999" will be generated by the system. . . . where Tbbbbbbb and T999999999 will be a generated value based on a time stamp and yyddd as the date the file was created. Like the data component, the system generated index name matches the FORMAT 1 DSCB in the VTOC on the DASD volume.

In most dataset catalog entries, the BCS only provides the basic information on the dataset entry with initial pointer(s) to the DASD volume(s) where the dataset resides. For NON-VSAM DASD datasets, the BCS will contain the relative track address (TTR) that points to the NON-VSAM's FORMAT 1 DSCB in the VTOC for the associated DASD volume. For NON-VSAM DASD datasets, the BCS depends on the VTOC on the DASD device to locate datasets. For VSAM datasets, the BCS will contain a Relative Byte Address (RBA) pointer to a VSAM Volume Record (VVR) that is stored in the VSAM Volume Dataset (VVDS) that resides on the DASD volume where the data/index component is allocated. The VVR and VVDS will be described later; the point here is that the BCS contains pointers to any dataset (VVDS) that contains additional and crucial information about each VSAM component.

The BCS will have a RBA pointer to a VVR in the VVDS that describes in more detail the information which is used to OPEN/CLOSE a VSAM component. A BCS entry (VVDS type) will exist for each DASD volume that contains a VSAM dataset that is cataloged in the BCS. The VVDS dataset name will be in the same format as the VVDS itself — the special name which VSAM management prefers, Sys1.VVDS.Vxxxxxx where "xxxxxx" is the actual DASD volume serial number. The BCS works in conjunction with the VVDS on each volume it currently has VSAM datasets cataloged. If the BCS contains an invalid RBA pointer(s) to a VSAM component (VVR in the VVDS), when VSAM tries to locate (direct read via RBA pointer in BCS) that component(s), VSAM management will take corrective action. VSAM Management will (in case of this bad RBA pointer) *sequentially* browse the VVDS until it finds the correct VVR for the VSAM component. VSAM management will then update the BCS to the now correct RBA pointer for that dataset. This readjustment is done automatically by VSAM and is transparent to the user (except small overhead).

The ICF software has also tried to improve on the data integrity during catalog update processing. When a BCS sphere record needs to be updated, an update in place indicator is turned on (like CI/CA splits). Each cell (possibly multiple cells) within the sphere record is logically updated (if needed) until all the possible cells are completed. When all required cells within the sphere have been updated, the sphere record is written and the update in place indicator is turned off. If a system crash or some other interruption would have occurred before this process completed, it is possible in many cases to reissue the same command/function that was in progress and did not successfully complete. The BCS will attempt to continue from the point of interruption and proceed through as it would normally.

As noted in the overview of the BCS, this part of the ICF catalog is the first stage in finding datasets. The basic cells and pointers to a VTOC or VVDS require another resource to complete the process. The second component of the ICF catalog is the VVDS that is described next.

VSAM Volume Dataset

The VVDS is a VSAM Entry Sequence Dataset (ESDS) with a system specified CI size of 4096. For each DASD volume that is to contain a VSAM dataset in a BCS, a VVDS will be defined per DASD volume. The VVDS is normally accessed randomly by using RBA pointers from the BCS. As mentioned previously, the VVDS has a special naming format (includes DASD volume serial ID) that allows VSAM management to provide the VVDS with specialized features that are not available for a regular ESDS. Unfortunately, the VVDS may *not* be accessible by standard IDCAMS functions such as EXPORT, IMPORT, REPRO or any other standard backup/recovery utilities. The only VVDS backup method supported by ICF is a full-volume DASD backup by such DASD dumping utilities available, although some third-party vendors' software products can be acquired to do this and other related VSAM functions.

The VVDS may be predefined on any or all DASD volumes *before* the first VSAM dataset is created. A simple IDCAMS DEFINE CLUSTER (sample in

F I G U R E 1

VVCR	SELF DESCRIBING VVDS RECORD	VVR	VVR	. . . more VVRs as space permits . . .

VSAM manual) may allow the space allocation DEFAULT (3,2 tracks) to be overriden and possible positioning on the DASD volume if desired. As a standard procedure for new DASD volumes, predefine a VVDS of at least one cylinder before the DASD volume is made available for use.

If a BCS is also to be created on the new DASD volume, for DASD performance reasons, it may be beneficial to have the VVDS next to the BCS. The process of predefining a VVDS does *not* require an entry in any BCS. The VVDS and a BCS do not become *connected* until the first VSAM dataset allocation on the DASD volume (not counting the VVDS itself). If a BCS would be allocated on the DASD volume (hopefully *after* a VVDS was created but not required), this would constitute the first VSAM dataset allocation since a BCS is a regular KSDS. The BCS and VVDS (if not predefined, it will be dynamically allocated by VSAM) will then have a connection established between them. The BCS will contain RBA pointers (one for data component and one for index) to the associated VVRs (its own) in the VVDS on the DASD volume.

You may or may have not noticed that the VVDS has been mentioned only with VSAM. The VVDS only contains information for VSAM datasets; the BCS and VTOC are responsible for non-VSAM DASD datasets.

The intent of the VVDS is to allow for the changing day-to-day information of VSAM datasets to be contained on the same device as the VTOC and the VSAM data components themselves. This design was to alleviate the previous out-of-sync conditions that were prevalent with the old VSAM catalog structure and DASD volume restore processing.

A look at the VVDS will find three types of records as shown in Figure 1. The first record in a VVDS is the VSAM Volume Control Record (VVCR). The VVCR contains two main sections within its 4096-record layout. The first 2048 bytes allow for up to 36 BCS catalog names (44 bytes each) with a couple of additional fields per each BCS entry. As a VSAM dataset is allocated onto this DASD volume, if it is the first time for this BCS, the BCS name will be updated into the VVCR. This entry is referred to as a *back pointer* to the BCS and represents the various BCS catalogs that may have VSAM datasets on this DASD volume. This record may be printed off by an IDCAMS print for possible recovery information purposes. Note that the old VSAM catalog ownership problem is nonexistent; up to 36 different catalogs may have datasets using DASD space on this volume. The BCS also creates a VVDS entry in its component that reflects this new business partner in addition to the VVR RBA pointer(s) of its new VSAM dataset. The BCS may have numerous VVDS pointers depending on how many DASD volumes its VSAM datasets reside on.

The second half of the VVCR's 4096 record layout is a *free space mapping* of the available/used control intervals within the VVDS. This is similar in concept to the Index VTOC free space mapping. When a new VVR needs to be created or moved, the VVDS uses the space mapping to determine where such new/expanded VVRs are to be placed within the VVDS.

The second record in the VVDS is a self-describing record for the VVDS itself.

Third and subsequent records in the VVDS are called VSAM Volume Records (VVRs). Generally, there is a one-to-one relationship between a VVR and a VSAM component (data or index) that resides on the DASD volume of the VVDS. Since an ESDS and a RRDS do not contain indexes, only one VVR will be associated with those types of VSAM datasets. For a KSDS, at *least* two VVRs (one data, one index component each) are created in the VVDS with matching RBA pointers from the BCS. If the KSDS has the IMBED or KEYRANGES on the same volume, a *third* VVR record may be associated with the KSDS, VVDS and BCS.

The contents of a VVR are the heart of the catalogs' entry for VSAM datasets. Within the VVR are *cell* types which contain, among other data elements, the physical extents (location) of where the VSAM component resides on the DASD volume, physical attributes such as key, record size, CI/CA sizes and so on and statistical related information such as number of records on file, CI/CA splits and so on. The physical extent information is a *duplication* (should be equal) of the same FORMAT 1 DSCB information in the DASD VTOC. In addition to working with the BCS, the VVDS works on a part-time basis with DADSM processing and the DASD VTOC. ICF was structured to allow DADSM to control all DASD volume allocation; VSAM sold the suballocation data space business. All requests, both initial define (new) or request for expansion (secondary), are passed through DADSM for space allocation processing. If the new or secondary space requested is available, DADSM will then allocate such requested space, creating new FORMAT 1 DSCBs and updating the DASD VTOC accordingly (fewer cylinders available and so on). This same physical extent information in the FORMAT 1 DSCBs (if KSDS, more than one DSCB) is passed and duplicated into newly created (or expanded) VVRs in the VVDS. The BCS is updated with RBA pointer(s) of the new VVRs. It may be possible that an existing VVR was moved within the VVDS to accommodate an increase in data. The BCS is updated with the new RBA pointer(s) accordingly. Even though the DASD VTOC contains the FORMAT 1 DSCB and knows the physical location of the VSAM datasets which reside on the volume, the VTOC is normally only referenced for VSAM datasets in these three instances:

1) Initial DEFINE of a VSAM dataset — to get such requested DASD space and physical extent information passed back to the VVDS for the new VVR(s)
2) The deletion of a VSAM dataset — the VTOC must free itself of the data and allow the space to be made available for other users
3) The extension of an existing VSAM dataset (secondary allocation) is treated much like the initial define process.

These are the only times VSAM normally requests access to the VTOC. VSAM relies on the VVDS (and initially the BCS) to find the VSAM datasets for OPEN and CLOSE processing. Without the VVDS, VSAM cannot find any VSAM datasets on the DASD volume. When a VSAM dataset is opened, the VVDS/VVR records are updated accordingly (open for output?) and when the VSAM dataset is closed, the statistical information (records read/CI splits) are updated accordingly in the associated VVR(s). Even the BCS is dependent on the VVDS on the same DASD volume as the BCS. Since it is a KSDS, the BCS relies on the VVDS to "find it" and allow the OPEN process to function.

If you were paying attention, you would have said that some of these things cannot be done since the VVDS is an ESDS and ESDS cannot delete records, insert or expand records and so on. You will now guess that since VSAM named each VVDS in a special format, it did so to allow the VVDS to break the house rules. This is the case; the VVDS does have special VSAM management code written for its processing to allow these normally illegal ESDS activities.

Just imagine how big the VVDS would grow if it could not reuse any of the previous CIs, since the process of deleting and redefining a VSAM file would cause nightmares to the VVDS if normal ESDS rules were followed. Due to this special access, the VVDS may remain relatively small, deleting and reusing the same control intervals as available. The ESDS may even move a VVR that needs to expand and move it to a larger empty CI within the VVDS. Note at that time the VVDS has changed the RBA pointer and must inform the BCS of the change so it can keep its RBA pointers up-to-date. Remember though, if this does not successfully occur, if VSAM detects an invalid RBA pointer from the BCS to a VVR in the VVDS, it will sequentially read the VVDS, find the new VVR entry and then update the BCS pointer accordingly.

F I G U R E 2

DISK01, BCS = BCS.CATALOG, SYS1.VVDS.VDISK01

Disk Drive # 1

```
VTOC (DISK01) — FORMAT 1 DSCB FOR SYS1.VVDS.VDISK01
              — FORMAT 1 DSCB FOR BCS.CATALOG  (DATA COMPONENT)
              — FORMAT 1 DSCB FOR BCS.CATALOG  (INDEX COMPONENT)

BCS — DATA COMPONENT   (BCS.CATALOG)
      INDEX COMPONENT   (CATINDEX . . . . . . GENERATED . . .)
      VVDS ENTRY FOR SYS1.VVDS.VDISK01
      VVR — 2 or 3 RBA POINTERS TO SYS1.VVDS.VDISK01 FOR BCS ITSELF

VVDS — SYS1.VVDS.VDISK01
       VVCR CONTAINS BACK POINTER TO BCS.CATALOG
       CONTAINS 2 OR 3 VVRS FOR BCS CATALOG  (IMBED ON ?)
       SPHERE RECORD FOR BCS

       . . . .
```

In theory, you could have or add many non-VSAM datasets here, additional VSAM datasets in BCS.CATALOG or up to 35 more VSAM BCS could reside here or contain VSAM files on this volume.

"Picture Time"

Hopefully a couple of pictures will let you finish the game and make the players a little more evident.

Figure 2 assumes the following: a disk drive (DISK01) exists; DISK01 contains a previously defined BCS named BCS. CATALOG and, of course, if a VSAM dataset exists (the BCS is a KSDS), a SYS1.VVDS.VDISK01 is also on the volume (may have been predefined or dynamically created). It is quite possible that other VSAM or non-VSAM datasets may reside on this DISK01 volume; some of the VSAM datasets may even be cataloged in different BCSes (up to 36). It is even possible to have up to 36 BCS catalogs on this same volume, although having more than one BCS on a volume is slightly suicidal; 36 BCSes on the same volume would seem to be a sure fire way of "early retirement" if the DASD volume has problems. These are just some of the many possibilities that could be colored in on the dull Figure 1 "picture."

Figure 3 shows acquiring a second DASD volume — this one is DISK02. You have just initialized the drive and created the DISK02 VTOC. You are now going to create a VSAM KSDS named VSAM.FILE in the BCS.CATALOG (resides on DISK01) and request that the VSAM.FILE be placed on the DISK02 volume. Here is what the little VSAM computer munchkins should be doing (not every detail is explained).

The access method service routines will syntax check the DEFINE CLUSTER statements. You naturally had that coded perfectly and now DADSM needs to find you space (amount passed from access method service) on DISK02. The VSAM munchkins realize that before you can create VSAM.FILE on the DISK02, you are missing a VVDS on that volume. VSAM quickly gets out its old recipe book and whips up a new VVDS with a space request of three primary tracks and two secondary tracks, placing it into the next available space on the DISK02 volume. Assume you either have a Stepcat DD statement for BCS.CATALOG or, better yet, an alias that will direct "VSAM." datasets to the BCS.CATALOG. VSAM will be hustling around: it will have to be creating the BCS data cell(s) information to write into the new sphere record for the new VSAM.FILE; and it will be awaiting the expected good news from DADSM about the location of its new allocated space on DISK02 for the data and index components for VSAM.FILE.

The VVDS will also be a busy little beaver since it must create its VVCR, a self-describing record and the VVRs for the VSAM.FILE. The VVDS will create a back pointer to the BCS.CATALOG (in VVCR) and likewise the BCS will create a second VVDS entry (it already had SYS1.VVDS.VDISK01) named SYS1.VVDS.VDISK02. Also lurking in the shadows is the DASD VTOC on DISK02; it has been updated to reflect the new FORMAT 1 DSCBs — one for a new SYS1.VVDS.VDISK02 dataset, one for the VSAM.FILE data component and a third FORMAT 1 DSCB for the INDEX component of the VSAM.FILE. It may be hard to believe, but this whole process has created many records and had several parties at once in the act and yet it is more efficient and reliable than the old VSAM catalog methods. Figure 4 reflects the circus act which has just been described.

I hope this provides some additional trivia on VSAM/ICF catalogs. My little children have dragged over the JES2 internal logic manuals and are eagerly awaiting tonight's bedtime story, so here we go. . . . Once upon a time. . . .

F I G U R E 3

Same As FIGURE 1, Add DISK02 Device

Disk Drive (another disk added)

```
VTOC   (DISK02)
   . . . NOTHING ON DEVICE EXCEPT FOR VTOC AT THIS TIME.
      . . . FREE SPACE AVAILABLE . . . .
```

F I G U R E 4

DISK01, BCS=BCS.CATALOG, SYS1.VVDS.VDISK01
WITH DISK02, NEW "VSAM.FILE" ON DISK02 ALLOCATED

Disk Drive # 1

```
VTOC (DISK01) — FORMAT 1 DSCB FOR SYS1.VVDS.VDISK01
              — FORMAT 1 DSCB FOR BCS.CATALOG   (DATA COMPONENT)
              — FORMAT 1 DSCB FOR BCS.CATALOG   (INDEX COMPONENT)

BCS — DATA COMPONENT   (BCS.CATALOG)
      INDEX COMPONENT   (CATINDEX . . . . . . GENERATED . . .)
      VVDS ENTRY FOR SYS1.VVDS.VDISK01
      VVDS ENTRY FOR SYS1.VVDS.VDISK02
      VVR — 2 or 3 RBA POINTERS TO SYS1.VVDS.VDISK01 FOR BCS ITSELF
      VVR — 2 or 3 RBA POINTERS TO SYS1.VVDS.VDISK02 FOR VSAM.FILE
        SPHERE RECORD FOR BCS ITSELF.
        SPHERE RECORD FOR VSAM.FILE

VVDS — SYS1.VVDS.VDISK01
       VVCR CONTAINS BACK POINTER TO BCS.CATALOG
       CONTAINS 2 OR 3 VVRS FOR BCS CATALOG   (IMBED ON ?)

        . . . .
```

In theory, you could have or add many non-VSAM datasets here, additional VSAM datasets in BCS.CATALOG or up to 35 more VSAM BCS could reside here or contain VSAM files on this volume.

Disk Drive # 2

```
VTOC (DISK02) — FORMAT 1 DSCB FOR SYS1.VVDS.VDISK02
              — FORMAT 1 DSCB FOR VSAM.FILE   (DATA COMPONENT)
              — FORMAT 1 DSCB FOR VSAM.FILE   (INDEX COMPONENT)

VVDS — DYNAMICALLY ALLOCATED AS 'SYS1.VVDS.VDISK02'
     — VVCR CONTAINS BACK POINTER TO BCS.CATALOG
     — CONTAIN 2 or 3 (IMBED ?) VVR RECORDS FOR VSAM.FILE, DATA
       AND INDEX COMPONENT

VSAM FILE — ACTUAL DATA COMPONENT AS PER SPACE REQUESTED
VSAM FILE — ACTUAL INDEX COMPONENT FOR VSAM.FILE

FREE SPACE FOR FUTURE BCS, VSAM DATASETS/NON-VSAM DATASETS . . .
```

ABOUT THE AUTHOR

James W. Clark is manager of technical services at Pennsylvania Higher Education Assistance Agency (PHEAA). He has been in data processing for 15 years. PHEAA, 660 Boas St., Towne House Apts. Annex, Harrisburg, PA 17102, (717) 257-2560.

Improving Mainframe COBOL/CICS Programmer Productivity

By Donald F. Tiernan and William J. Danker III

One of the more pressing problems faced by MIS managers is the *programming backlog*. This is not a new problem. It has been with MIS management since the beginning of the Information Age; the requests for service have always outpaced the capabilities of the computer programming staff.

Over the years, several methods have been advanced as a means to solve this problem. Report writers, fourth-generation languages, information centers, purchased software and so on are a few examples. Each has met with some success and some failure. None have solved the problem; the backlog still exists.

In most situations, the programming backlog results from the need for complex reports or "industry specific" on-line transactions. These represent the areas where MIS can best give a company *strategic advantage* over its competition — the worst place to have a backlog of work! The purpose of this article is to report on an advancement that can help solve the backlog problem; one that can improve programming productivity 20 to 50 percent.

Prerequisites

We believe that most well managed programming groups today use the following:

1. Third-generation languages
2. Structured programming methods
3. Shop standards for programming and documentation
4. CRT with full-screen editor
5. Management "walk-throughs"
6. An efficient test environment
7. Effective control of the program library
8. Quality assurance procedures.

If these are not being used, management should begin work to implement them. They all improve the efficiency and quality of the programming group. We believe they are also essential before implementing any new advancements into the programming group.

New Advancement

The advancement that we are reporting on is the Personal Computer (PC). We are going to describe exactly how to put together a PC workstation that will enable you to do all mainframe COBOL/CICS program development, modification and testing on a PC and then upload the completed programs to a mainframe for execution. In addition, we are going to tell you the benefits you can expect and how the workstation can be cost justified.

Building a programmer workstation is the same as any project development; you must start with a definition of requirements. Our requirements were to write, compile and test COBOL batch and COBOL-CICS programs on a PC. The completed programs were then to be uploaded to the mainframe, recompiled and placed in the program library.

From this requirements specification we developed a list of tasks:

1. Find a PC editor that functions like the mainframe's
2. Find a PC COBOL compiler that creates mainfame-compatible code
3. Find a PC-CICS product that would emulate mainframe CICS
4. Find a BMS map editor for the PC
5. Find a PC sort program that functions similar to the mainframe IBM SORT
6. Develop a method to download and upload COBOL source programs and CICS maps
7. Develop a method to download test files
8. Develop a PC configuration to support 10 people writing and testing COBOL-CICS programs.

Screen Editor

For a screen editor, we chose SPF/PC from Command Technology (Oakland, CA). It is similar to our current mainframe editor and provides several extra utilities which eliminate the need for programmers to become PC-DOS literate.

COBOL Compiler

For a PC COBOL compiler, we selected REALIA's (Chicago, IL) COBOL compiler. The code created with this compiler is completely mainframe compatible. It also contains a useful debugger (REALDBUG) and a utility that creates VSAM equivalent files (REALCOPY).

CICS And Map Editor

For the CICS translator, we chose another REALIA product, REALCICS (it also contains a BMS screen editor).

SORT

For a stand-alone sort, we selected OPT-TECH SORT from OPT-TECH Data Processing (Zephyr Cove, NV). This gives us the capability to sort large files using the same sort command format as on the mainframe.

Download/Upload Programs And Maps

Our IBM mainframe operates under VSE/SP. The Interactive Interface Facility of VSE/SP has a function that will transfer an ICCF member to a PC with an IBM 3270 emulator card (our program library is in ICCF).

Downloading BMS maps requires some development work. We are using SDF as the screen editor on the mainframe. SDF contains a utility to convert SDF maps to BMS macros. However, in the process the COBOL field names incorporated into the screens are "commented-out" to meet BMS' restriction of seven-character variable names. To solve this, we wrote a PC program that reconverts the "commented-out" field names back into the BMS macros.

Uploading programs and maps is simply a file transfer through the Interactive Interface to the mainframe. They are then recompiled or regenerated and placed into production.

Download Test Files

VSAM files that do not contain packed data are downloaded directly from the mainframe to the PC by the Interactive Interface. However, VSAM files with packed data require additional work. The Interactive Interface requires packed fields be expanded to display format before transferring to the PC. The Interactive Interface also contains an option to allow user-written CICS programs to modify records before they are moved to the host transfer file. To solve the problem of transferring files with packed fields, we wrote a CICS program (for each VSAM file with packed fields) that expands the packed fields as the file is moved to the transfer file. Then we wrote a conversion program on the PC to reconvert the fields back to packed format.

Another problem with transferring files to the PC is the ASCII collating format. The downloaded files are no longer in the correct sequence. This requires sorting the files on the PC before they can be set up as KSDS files.

PC Configuration

The PC configuration is based on a network of 12 PCs connected with PC LAN on IBM's token ring network. We selected a PS/2 Model 80 as the file server, print server and 3270 gateway to the mainframe and PS/2 Model 50Zs for the programmer workstations. The file server contains the REALIA software, SPF, SORT, COBOL source programs, BMS macros and CICS tables. The test files are kept on diskettes and loaded as needed onto each workstation's hard disk.

One difficulty with this configuration involves DOS. The DOS memory limitation on the workstations will not allow file transfer to work through the file server's 3270 gateway (that required Version 3.0 of the 3270 emulator program). As a result, 3270 emulator cards were put into each workstation PC and they are coax-attached to the mainframe.

Benefits

With this configuration in place, what are the benefits? First, the editor is much faster and more powerful than the mainframe's. Second, compiles take one minute compared to 10 to 20 minutes elapsed time on the mainframe. Third, CICS testing only affects the workstation being used — if a programmer "crashes" CICS, it only occurs on his own workstation and everyone else is unaffected. Fourth, CICS table changes can be made in a matter of minutes compared to at least a day on the mainframe.

The last benefit is in testing. This is the most significant gain. Each programmer has a copy of the test files on diskettes. If (s)he corrupts the files during testing, (s)he reloads them onto the hard disk (in many shops, it can take a day to get a test file reloaded). Also, several people can be testing programs which use the same files without impacting each other.

All of these benefits speak to improving programmer productivity. We had an efficient programming environment before we implemented the PC workstation concept. With the workstation, productivity improved at least 20 percent!

There is another benefit; the mainframe load is reduced by having all development and testing performed on PCs. This is not as large a benefit as those above, but it is still significant.

Cost Justification

Building a PC environment like the one we have described can be easily cost-justified. A productivity gain of 20 percent in a 10-person programming group is like adding two people to the group. To put a dollar value on that, assume each programmer's annual salary is $25,000. The present worth of $25,000 per year for 10 years at 10 percent is $153,600! Thus, a 20 percent gain with a five-person programming staff is worth at least $153,600. With a 10-person staff it is worth at least $307,200 and so on (if your organization's benefits are about 40 percent, as most are, these numbers go up to $215,040 and $430,080!).

Summary

Setting up the workstation was a lot of work. Was it worth it? We think so — we are getting at least a 20 percent improvement in productivity. Do our programmers like it? After the learning curve period (about two weeks of use), they do not want to go back to working on the mainframe. All new development is done on the workstation and all maintenance, unless it is extremely trivial, is downloaded and modified. We are excited about the results! We expect that our experiences will help others to accomplish the same results.

ABOUT THE AUTHORS

Donald F. Tiernan

William J. Danker III

Donald F. Tiernan is MIS Director of Garst Seed Co. and has more than 20 years experience in his field.

William J. Danker III is Programming Manager at Garst Seed Co. where he also concentrates on new technology and development. Garst Seed Co., 615 Main Street, Coon Rapids, IA 50058, (712) 684-2211.

Q We are currently running three VSE machines under VM on a 3083-J32 processor. We are about to upgrade VSE from Release 1.3.5 to 3.2. We are concerned about virtual storage limits (16 MB) on the three guest machines. One option would be to run one of the VSE machines in VAE mode in a V=R area. Another option we are considering is converting the two production machines into one and running it in VAE mode on a separate smaller processor. This would leave the remaining test VSE machine under VM on the 3083. Would it be possible to use PR/SM on a single processor as another option? Do you have any clients with a similar situation? We are running Release 5 of VM/SP HPO.

Edward J. Peck, Albany International Corp.

A I am making an assumption that you are using VM to do more than just run VSE guests, since you are considering keeping VM and a TEST VSE guest running on the 3083 and getting a smaller processor for your production machines.

Your first option to run in VAE mode is a good choice if you have real memory to dedicate to the V=R machine. Secondly, you are considering converting the two production machines to one VAE machine and running this native on a smaller processor (Pete Clark of Olan Mills swears by this). But consider your communication and terminal sharing needs. Also your costs would be greater for both hardware and software, since you will need multiple licenses for IBM and third-party vendor software. You may wish to consider combining your two production machines to one VAE V=R machine on your current processor. By combining your production machines you may also be able to eliminate Shared POWER and Shared DASD.

You also questioned the use of a PR/SM on a single processor. This would be a large scale hardware change converting to an ES/3090. To answer your question of using PR/SM as an option, yes it is an option. VM HPO and VSE/SP 3 are both supported on PR/SM processors. This is documented in the *ES/3090 Processor Complex Planning Guide GA22-7123-1*. There are some architectural differences running in Logical Partition Mode (LPAR) documented in Appendix A of this manual. These differences are documented as "rarely, if ever, affect normal operation." The differences mainly affect I/O conditions and should not affect the operation of VSE. The only problem I could see is possibly with the Missing Interupt Handler VSE.

I was able to contact one client who was running VSE with the PR/SM support and his comment was, "It works beautifully and there have been no problems — everything is quiet!"

(Answer provided by Robert E. Smith of Goal Systems International, Inc., Columbus, OH)

Q We have a CICS command-level program that uses temporary storage to pass data between pseudo conversational tasks. Most of the time this works fine, but on occasion the temporary storage records, when read, contain binary zeros. What happened to the data that was written to temporary storage?

A When you issue an EXEC CICS WriteQ TS, the command-level processor acquires a new work area and moves your data into it before requesting service from the CICS Temporary Storage Program. This move is accomplished through the use of a Move Long (MVCL) instruction that can process up to 16MB at a time.

The 370 hardware checks the operands of the MVCL instruction prior to executing it. If the execution of the MVCL would result in one or more bytes of the sending field being overlaid, the hardware refuses to process the instruction. This is termed a destructive overlap. When this happens, the hardware sets a condition code and resumes processing with the next sequential instruction in memory. It is up to the program issuing the MVCL to check the condition code to verify that the MVCL functioned properly. Unfortunately, the command-level processor does *not* check the condition code and goes merrily on its way assuming that the MVCL worked correctly.

Causing a destructive overlap is fairly easy to do in a command-level program. If you specify a length for the EXEC CICS WRITEQ TS command that is longer than the actual length, it is possible, due to the way CICS suballocates memory, that you may run into a destructive overlap situation. You must make sure that the length you specify is correct.

(Answer provided by Dave Dick of Davis, Thomas & Associates, Minneapolis, MN)

Q Every once in a while programs cancel with "INVALID ADDRESS". I have set up my VSE/SP system to use 16MB, so how can any address be invalid?

A Certain VSE Supervisor Calls (SVCs) contain restrictions on where their associated parameter lists may reside in memory — generally, system GETVIS storage. If an SVC is issued with the parameter list in the wrong place, the partition will be cancelled with an "INVALID ADDRESS" message. In a normal application program, SVC parameter list restrictions should not pose a problem. There is, however, another way in which this cancellation can occur.

In VSE/Sp 2, IBM introduced the concept of Virtual I/O (VIO). VIO is an extension to your page dataset and the paging routines of the VSE supervisor are used in maintaining this data. At IPL time, VSE sets aside a VIO buffer area (VIOPOOL) to contain the in-core portion of the VIO data space. The VIOPOOL follows the system GETVIS area and extends to the upper limit of memory (address FFFFFF). VSE treats the VIOPOOL space differently from the way it treats other address ranges within the VSE machine.

If your program attempts to reference an address contained within the VIOPOOL, it may or may not be cancelled by VSE. If the page you are referencing is currently in core, you will be allowed to access it. If the page must be paged in to satisfy your request and your partition is not the owner of that particular page within the VIOPOOL, then your program will be cancelled with an "INVALID ADDRESS". Notice that there is no way to predict whether your program will be cancelled or not — it all depends upon the current status of the page you are attempting to reference.

It is unfortunate, but VSE does not provide you with a dump to help debug the problem. In VSE/SP 3.2, IBM addressed the dump issue by performing the edit sooner in the page-in process and now provides an addressing exception with a dump instead of the "INVALID ADDRESS" cancellation — but it still hinges on the current status of the page you are attempting to reference.

To finally answer your question, in all likelihood, an index or a subscript in your program has gone "wacko" and you are attempting to access an entry well beyond the limits of your partition. You will have to check your program manually since, short of SP 3.2, IBM does not provide you with any assistance for this problem.

(Answer provided by Dave Dick of Davis, Thomas & Associates, Minneapolis, MN)

BARR/RJE Provides Fast Printing At The IRS

By Steven A. Taylor

Few dates bring forth the emotions and the instant recognition that the fateful day, April 15, brings to Americans every year. As this time of reckoning with the Internal Revenue Service approaches and everyone is faced with the Herculean task of reconciling personal accounting with that of the government, it is comforting to know that the Internal Revenue Service itself also faces such tasks at times.

In early 1987, the Remote Communications Center for the programming staff of the Internal Revenue Service (IRS) based in Washington, DC, not only faced such a task, but also it conquered the task with an efficient, cost-effective solution.

From left to right: Anthony J. Barr, President of Barr Systems, Inc.: Lawrence Riley, IRS Operations Unit Chief; and Jack Fry, President of Data Systems Hardware, Inc. are shown at the IRS' Remote Communications Center.

The Remote Communications Center is connected to the IBM 3084 mainframes at the IRS Martinsburg Computing Center (MCC) in Martinsburg, WV, via a Comten 3695 at the MCC and a Comten 3690 in Washington, DC. Program development by more than 300 programmers in the Washington area requires colossal printing capabilities. Daily printing operations at the Remote Communications Center regularly produce up to three million or more lines of print per day via RJE communications.

The printing operations were using eight 1,250 Line Per Minute (LPM) printers attached to multiple older workstations using 3780 protocol. The problem was that the printing center for the area was running out of steam. Further, maintaining these workstations was becoming an annoying and expensive problem, so a replacement was needed.

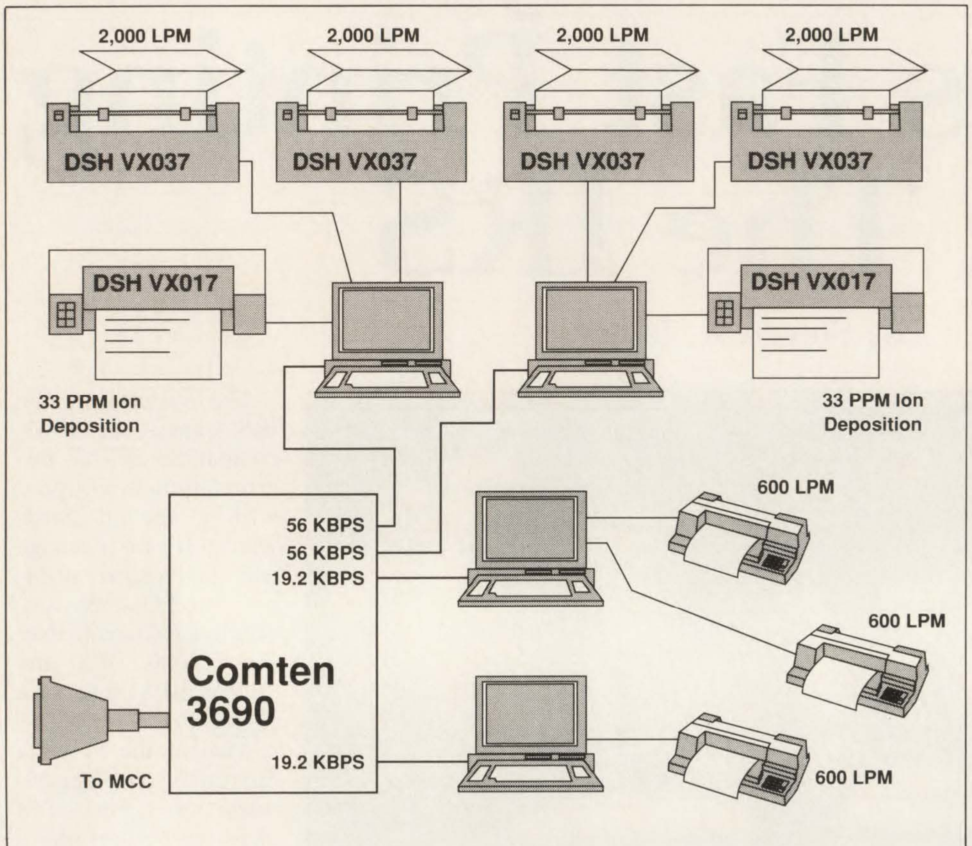
Help was found and the package it came in may surprise many readers. It was not in the form of a new specialized workstation or even in the form of a minicomputer. Rather it came in the form of high-speed printing from PC platforms with each PC capable of printing more than 6,000 LPM.

The systems at the IRS consist of IBM-compatible PC/AT microcomputers coupled with a special hardware/software package and high-speed printers. The PCs then connect to the Comten 3690 via 56,000 BPS and 19,200 BPS links (see Figure 1).

Two of the PCs are currently configured with two 1,500 LPM character printers plus a 2,000 LPM Ion Deposition page printer, totaling 5,000 LPM of output per PC. Since the line printers are capable of printing 2,000 LPM with different print bands, the potential output printing speed totals 6,000 LPM. This output printing speed is kept busy by communications lines feeding the PCs at 56,000 BPS. (A 56,000 BPS line supplies 7,000 bytes of data per second or 420,000 bytes per minute. If one assumes an average line length of 70 uncompressible characters, this is the equivalent of 6,000 LPM.)

The hardware and software package that supports this high-speed printing for the PCs, called BARR/RJE, was developed and manufactured by Barr Systems, Inc. (Gainesville, FL). This package supports a wide variety of speeds and protocols, allowing the PC to serve as a full-function RJE workstation. A special hardware interface board controls the basic communications functions, but the majority of

FIGURE 1



Each BARR/RJE PC system at the IRS installation in Washington, DC, is capable of printing more than 6,000 LPM.

the processing is done by the PC itself. This serves the dual purposes of keeping the hardware costs to a minimum and allowing upgrade of the package without hardware replacement. Several protocol options are available as well and the IRS chose to implement a HASP BSC multileaving system.

The package supports a wide variety of printers, so the printer may be matched to the users' needs. The printers can be literally anything from small, inexpensive dot-matrix printers to sophisticated Ion Deposition printers capable of printing thousands of lines per minute. The physical interfaces may be either parallel Centronics or serial RS-232 interfaces. Software drivers are available for 18 different printer types. These drivers even map the carriage control from the mainframe to each printer's own unique codes, so the user has the ability to choose virtually any type of printer.

At the IRS the printers chosen for high-speed work were 2,000 LPM Model VX037 line printers plus 33 page-per-minute Model VX017 Ion Deposition printers (similar in technology to laser printers) all from Data Systems Hardware (DSH) in Sterling, VA. The Ion Deposition printers are especially useful in this application. The programmers like the advantages over their older technology including both upper case and lower case printing capabilities, the ability to print special characters and a more convenient size of paper (8½ by 11) for listings.

The combination of the Ion Deposition printer with the package provides additional features which the IRS finds useful. Special codes may be put in the program to set different fonts and to draw lines and boxes when creating special forms and charts. Another feature is the added convenience of combining the software's option to insert a job separator page with the Ion Deposition printer's multiple paper bins. This allows automatic insertion of separator pages of a different color between jobs.

Since the system provides full multileaving capabilities on the PC, there are some added benefits beyond high-speed printing. The full console support is quite convenient. In addition to supporting standard operator commands entered from the keyboard, up to 39 function keys on the PC may be defined to perform commands which are often used such as forward spacing and back spacing the printers. There is also an automatic log-on function for easily initiating or restoring the communications line.

The color support on the system is also helpful. Lawrence Riley, the Operations Unit Chief at the Remote Communications Center and the person with primary responsibility for the operation of the system, states, "We have three systems that are using color monitors and one that is not. We find that the ones that are using the color monitors are much clearer, help us see the picture better and define the different things that you have to look at. On the one that is not (color), everything tends to run together and you do not notice things nearly as quickly."

While the package fulfills the functional requirements at the IRS, it also provides some additional capabilities and possibilities which were not anticipated. One of these is the possibility of job submission as well as job retrieval. The system is a full-function workstation, so jobs may be submitted from the PC, from disk files prepared on other PCs or from other PCs networked to a single PC. However, this function is not currently used by the IRS since most of the jobs are submitted from 327X workstations located throughout the metropolitan DC area.

Another feature that is a happy surprise

TABLE 1

Protocol	*Maximum Speed*
3780	64,000 BPS
2780	64,000 BPS
HASP BSC Multileaving	64,000 BPS
SNA RJE with SDLC Connection	256,000 BPS
SNA RJE with Token Ring Connection	4,000,000 BPS or 16,000,000 BPS

A wide variety of communications options are supported by the BARR/RJE products.

to the IRS is the ability to retrieve jobs to the hard disk on the PC for later printing. In the application at IRS, the printer speed, as shown above, is sufficient to handle the data line speed. However, if a printer were having maintenance problems, the output could be spooled directly to a hard disk on the PC and printed off-line at a later time — even if the MCC were not on-line.

Thus far I have concentrated on the technical aspects that make this solution work for the IRS. However, all solutions consist of more than pure technology, so some of the other factors should be considered as well.

For instance, two of the PCs with their printers take more than the full workload of the eight printers on the older 3780 workstations, but they take up less than half the floor space. This offers much-needed physical room for future expansion — a significant benefit since few resources are more valuable than floor space in most computer rooms.

There is room for expansion in other ways as well. The two main systems are each capable of producing 6,000 LPM of printing. On an hourly basis, this is equivalent to 720,000 lines per hour. Consequently, the workload for a busy day in excess of three million lines of print could be handled in slightly more than four hours if all of the jobs were ready for constant output.

However in real life, programmers submit the jobs randomly and the output is likewise random. Consequently, the printing load is not steady. Still, having this amount of printing power means fast turnaround. Prior to the installation of the package at the IRS, it was not unusual to have more than 100 jobs backed up in the print queues. Now, the queues are often empty by 1 p.m. This type of rapid turnaround obviously increases the productivity of the programmers. The operations staff's productivity is enhanced by the package as well.

When asked about training and installation problems, Lawrence Riley replys, "We found it easy to train the operators with this system. One of the things I like best is the ease of access to the installation process in order to train the operators in how to manipulate the system."

This ease of installation is perhaps best demonstrated by the way an additional system made its way into the IRS. After stating that the third system was installed primarily for backup to the other two systems, Riley continues, "Our fourth line went into operation because we were having some difficulties with another system entirely — one for which we had never intended to use the package." He goes on to explain that since there had been a nine-to-twelve-month delay in getting the operation under way, the Remote Communications Center staff decided to try the package. On the afternoon that the staff installed the BARR/RJE board, the new system was in operation within two hours. "It was extremely beneficial and saved the (Internal Revenue) Service a great deal of money," he notes.

Installation and maintenance usually go hand in hand. President Anthony J. Barr, the founder of Barr Systems, states that among the major reasons the IBM-compatible PC was chosen as the platform for the system were wide availability of hardware, ease of maintenance when needed and the inherent reliability and low maintenance cost of PCs.

This has certainly proved to be true of the maintenance experiences at the IRS. According to Jack Fry, President of Data Systems Hardware, "From the standpoint of reliability, the hardware board itself has never failed. We are using NEC PCs and they are very reliable. The printers require maintenance, but that is not unusual in that they are mechanical devices."

The efficiency of the communications lines is yet another area in which the IRS receives added benefits from the package. Fry continues, "From the efficiency standpoint, HASP provides more efficient use of the bandwidth than 3780 or some other less efficient protocols. I was surprised at the total (print) throughput we could get from the PC."

Barr Systems bypasses DOS in the PC to go directly to the printer and uses machine language programming to give the PC the ability to use the powerful HASP protocol. This protocol has previously been reserved for expensive devices and host-to-host transfers with systems such as the S/36 and AS/400. In addition to providing excellent transmission efficiency, the IRS found significant benefits in no host software modifications being required.

The end result of this installation at the IRS is a productive system for all involved. The programmers are happy with the system because they get their printouts in a timely fashion. They also like the increased print options like multiple fonts and a choice of portrait or landscape mode available from the Ion Deposition printer.

The operations staff is also happy with the system. In fact, they are so pleased with the performance of the system that they have asked Barr Systems and Data Systems Hardware to explore adapting the system to support some of their other systems as well. In response to this and numerous other requests, Barr Systems continues to expand into a number of different areas. These include support of SNA RJE and SNA RJE token ring networks, so the path for the support of a wide range of options is clear (see Table 1).

For now though, the IRS has a system that is productive, is saving money, is easy to install and maintain and is keeping users happy.

ABOUT THE AUTHOR

Steven A. Taylor is President of Distributed Networking Associates in Greensboro, NC. DNA offers services in telecommunications consulting and education with specialties in high-speed digital networks (T1/T3) and ISDN.

CACHE Management

Avoid The Pitfalls

By Clifford J. Goosmann and Kenneth Nethercote

Cache controllers have been designed to require few or no changes in installation and operating procedures to obtain performance improvements. Occasionally, this has been interpreted to mean that there are no special procedures to be followed when doing hardware changes or maintenance. The following article points out some of the pitfalls that can occur if cache is not managed properly. Guidelines (along with sample JCL) are provided, indicating when the tasks should be performed.

Industry standard cache controllers are designed to require little or no effort from operators or users to obtain performance improvements. To this end, some vendors recommend caching all volumes behind a cache controller. However, under certain conditions such as generally high activity and low read hit rates, performance can actually degrade. (This degradation occurs on all cache controllers but has limited impact on National Advanced Systems (NAS) controllers since they have a unique parallel processing feature. This allows the controller to transfer data from cache to the CPU while staging a track from DASD to cache.)

Operational Considerations

In addition to performance concerns, there are operational considerations that must be understood to ensure data integrity. Cache is designed to support nondisruptive operation. This means that changes to cache will not necessarily change its status.

The default condition of a cache subsystem is ACTIVE. For instance, if power to all controllers is lost and subsequently restored, each cache subsystem will start caching and every disk volume will be cached. During IPL, MVS simply interrogates the status of the cache subsystem without changing it.

Several conditions may make it necessary to stop caching one or more volumes. The most likely reason to stop caching a volume is that it is a poor cache candidate (poor read-hit rate and/or poor read/write ratio). When an HDA has been replaced, you must stop and start caching to invalidate any cache slots from the old disk pack still remaining in cache memory. Configuration changes may also require stopping and restarting cache.

When you manipulate the status of caching, you need to run the jobs on only one system. When a disk volume is being cached, the reads from all systems will go through cache. If you stop caching on a volume from any system, then you stop caching on all systems. If you stop the whole cache subsystem, you will stop caching from all systems.

Subsystem Caching Status

The status of subsystem caching is not affected by powering off one Storage Director (SD). The other member of the SD pair will continue to cache. When the first SD is powered on again, it simply interrogates the status of caching. If cache is ACTIVE, it joins the subsystem and begins caching itself without disrupting existing valid cache slots. If cache is OFFLINE, it joins the subsystem but it does not begin caching. However, if both SDs in a pair are simultaneously powered off/on (or simultaneously IMPLed), then the status of caching defaults to ACTIVE and all cache slots are invalidated.

The two SDs in a single-frame controller belong to the same cache subsystem. If power is lost to the whole controller, the cache subsystem will default to its

ACTIVE status when power is restored.

In a dual-framed pair of controllers, there are two cache subsystems and four SDs. If power is lost to one of the controllers, two SDs and one cache subsystem will become inoperative. The other frame will remain in operation and have access to all disk volumes. When power is restored to the frame, the cache subsystem will default to its ACTIVE status.

Performance and Integrity

To prevent performance and integrity problems, it is essential that cache be actively managed by the installation. The tools provided by MVS are the LISTDATA and SETCACHE subcommands of the IDCAMS utility. The NAS Cache Reporter Tool (NASCRT), as well as products from other vendors, can be used to monitor cache performance.

The following guidelines should be implemented and followed for all cache controllers.

- Monitor cache status daily by running the LISTDATA facility of the IDCAMS utility.
- Monitor cache activity and performance on a regular basis (for example, using NASCRT).
- Establish operator procedures that should be followed whenever the status of cache may change. These include, but are not limited to, component failure, maintenance, configuration changes and software changes.

Sample jobs and recommendations follow for the suggestions listed above.

Monitor Cache Status

Below is a sample job intended to list the status of four 7880-3C controllers from NAS (Santa Clara, CA). This will tell you if the cache subsystem is ACTIVE, OFF-LINE or has a SUBSYSTEM ERROR. The WTO will cause a message to be displayed on the MVS operator console. The SYSOUT listing will additionally show which disk devices are being cached.

```
//      EXEC PGM = IDCAMS
//SYSPRINT DD SYSOUT = *
//SYSIN   DD*
/*Controller #3 X'880' SD-IDs F2, F3 */-
LISTDATA STATUS UNIT(3380) VOLUME(800652) WTO
/*Controller #4 X'8A0' SD-IDs F4, F5 */-
LISTDATA STATUS UNIT(3380) VOLUME(800668) WTO
/* Controller #5 X'8C0' SD-IDs F6, F7 */-
LISTDATA STATUS UNIT(3380) VOLUME(800676) WTO
/* Controller #6 X'A00' SD-IDs F8, F9 */-
LISTDATA STATUS UNIT(3380) VOLUME(800717) WTO
```

This job should be run and the output reviewed daily and compared to existing standards. If a discrepancy exists, corrective action should be taken as described in the operations procedures. SUBSYSTEM ERRORS generally indicate that the appropriate hardware vendor should be called.

Note that this JCL must be modified for the appropriate configuration and volumes and the comments removed.

Monitor Cache Performance

A sample job to retrieve the counters maintained in the SDs is shown below. If you monitor these counters over time, you can tune your cache subsystems for optimum performance. The counters are not zeroed when you run this job. The only way to clear the counters is to IMPL or power off the SDs. NAS has a program called NASCRT, available on request, to calculate read-hit ratios from these counters. Similar programs to calculate cache performance ratios may also be available from other vendors.

```
//      EXEC PGM = IDCAMS
//SYSPRINT DD SYSOUT = *
//SYSIN   DD
/*Controller #3 X'880' SD-IDs F2, F3 */-
LISTDATA COUNTS UNIT(3380) VOLUME(800652) SUBSYSTEM
LISTDATA COUNTS UNIT(3380) VOLUME(800660) SUBSYSTEM
/* Controller #4 X'8A0' SD-IDs F4,F5 */-
LISTDATA COUNTS UNIT(3380) VOLUME(800668) SUBSYSTEM
LISTDATA COUNTS UNIT(3380) VOLUME(800720) SUBSYSTEM
/* Controller #5 X'8C0' SD-IDs F6,F7 */-
LISTDATA COUNTS UNIT(3380) VOLUME(800676) SUBSYSTEM
/* Controller #6 X'A00' SD-IDs F8, F9 */-
LISTDATA COUNTS UNIT(3380) VOLUME(800717) SUBSYSTEM
LISTDATA COUNTS UNIT(3380) VOLUME(800729) SUBSYSTEM -
   LEGEND /* Brief explanation of all the counters */
```

Note that this JCL must be modified for the appropriate volumes and configuration and the comments removed.

Establish Operator Procedures for Controlling Cache

To facilitate operator action, a series of IDCAMS commands must be made available either as started tasks from the operator console or jobs that can be submitted. Even if the operator should not normally have this capability, it should be available for emergency conditions or following scheduled changes.

Incorrect Cache Status

Describe the actions to be taken by operations when the status of cache does not match installation standards. If this is under control of performance or storage management personnel, then the only situation to be addressed is the occurrence of a cache error.

Scheduled Maintenance

Based on the previously mentioned concerns, cache must be reset prior to returning to active use. This can be done by setting cache "OFF," then "ON." The set cache "OFF" should be done prior to shutting down and turning the system over to the vendor. The vendor hardware support personnel can also reset cache. If this activity is desired, it should be requested and verified afterwards.

Unscheduled Maintenance

After any unscheduled outage, it is safer to reset cache. Performance may be degraded while cache is off, but data integrity will be maintained.

The following sample jobs are typical of those used to change the status of cache for a single volume or the entire subsystem. "ON" can be substituted for "OFF" to turn cache back on.

Single Volume

```
//      EXEC PGM = IDCAMS
//SYSPRINT DD SYSOUT = *
//SYSIN   DD
/* CONTROLLER #5 X'8C0' SD-IDS F6, F7 */
SETCACHE DEVICE UNIT(3380) VOLUME(800679) OFF
```

Subsystem

```
//      EXEC PGM-IDCAMS
//SYSPRINT DD SYSOUT = *
//SYSIN   DD *
/* CONTROLLER #5 X'8C0' SD-IDS F6, F7 */
SETCACHE DEVICE UNIT(3380) VOLUME(800676) OFF
```

Note that this JCL must be modified for the appropriate configuration and the comments removed.

The volume serial number in the subsystem job identifies the cache subsystem you want to put OFF-LINE. It could be any volume serial number in the disk strings attached to that cache controller.

If there are hardware errors in cache, they can usually be worked on independently of the SDs. In such a case, you might need to run the above job if the cache is not already showing a status of SUBSYSTEM ERROR. This will prevent MVS from issuing error message IEA454E when cache is powered off. There is a test/normal switch in the controller. While this switch is in the test position, any job trying to turn on subsystem caching will be unsuccessful.

ABOUT THE AUTHORS

Clifford J. Goosmann (CDP) is a senior systems engineer for National Advanced Systems (NAS). He has been in data processing for more than 23 years and is currently involved in marketing support, capacity planning, configuration planning and performance analysis.

Kenneth Nethercote is a customer service account representative for NAS. He has more than 16 years in the data processing field and is currently involved in customer service, hardware and software support. NAS, 11300 Cornell Park Dr., Suite 545, Cincinnati, OH 45242, (513) 489-8639.

Service Levels *from page 43*

turnaround time, system availability, accuracy and load.

What Is SLM?

After SLAs are set and agreed upon, the next and continuing step is to manage the service levels. Management requires a reporting process, contingency plans and scheduled meetings. Simple reporting will show SLAs with actual service, differences between the two and possibly trends. Some installations only report exceptions (missed SLAs) on a daily basis but show trends on a weekly or monthly basis. For example, I do not need a report that shows I am meeting my service levels every day; however, I need to know if I am getting closer to exceeding them during the month. Am I about to fail?

Follow-up on missed service levels is a requirement. Is it due to a hardware failure, operational failure, increased workloads, unrealistic service levels and so on? You can continue to provide quality service only if you know what happened when you failed. Documentation helps here.

Part of the SLA should include a sample reporting scheme that is agreed upon. Flexibility, of course, is a must. Compromise and arbitration might be needed during some periods. For example, a data center realizes that a CPU upgrade is required during the next year. The longer it can be put off, the less resultant cost. A negotiation could take place in which the users agree to decreased service levels in exchange for reduced billing, better service in the future, longer availability for the application and so on.

FIGURE 2

Service Level Agreement

Department: xxxxxxx — Date: xx/xx/xx

Contacts: xxxxxxxxxxxxxxxx — xxxxxxxxxxxxxxxx

xxxxxxxxxxxxxxxx — xxxxxxxxxxxxxxxx

Approval: ____________________ ____________________

CICS service:		Goal	Actual	Diff
Availability:	8 a.m.-5 p.m., M-F	98%	100%	2%
Response:	% of responses within 2 seconds internal	90%	95%	5%
Load:	transactions per minute during peak hours: 10-11	300	250	−50
	daily CPU hours	3.5	3.0	−.5
Accuracy:	errors due to data center problems	0	0	0
	errors due to application problems	0	0	0

Availability based on CICSPROD up and files open.
Penalties for missed service:
10% reduction in billing for 2% missed service unless missed service caused by user.
Penalties for exceeded loads:
10% increase in billing and no penalty for missed service.
Reporting: data center will provide this report by 8 a.m. each day. Weekly report will summarize service for the week.
Changes to SLAs must be negotiated with xx and yy.
Priorities: If full resources are unavailable, TSO users will be logged off to support CICS.

Batch service:	Goal	Actual	Diff
Class S: % tests in 30 minutes	95%	85%	−10%
Class T: % tests in 15 minutes	98%	100%	2%
Load . . . etc. . . . similar to CICS service . . .			
Turnaround is defined to be from job submission to job end (not including print time)			

Summary

The discussion of service levels has been around for many years and is implemented in most large installations. However, most installations, especially the smaller ones, have not yet implemented any sort of service levels. As you can see, the entire process can be fairly complex; however, it is extremely easy to start. Once you start with SLOs, SLAs can be added later and SLM will follow naturally. One step at a time!

ABOUT THE AUTHOR

Cheryl Watson has 23 years experience with IBM mainframe software, having worked in systems and developed training courses. Her expertise includes performance, capacity planning and chargeback. She is President of Watson Resources, Inc. and provides MVS courses including: SRM, SMF, RMF, capacity planing, performance measurement, IPCS and IBM products (PR/SM, ES and ESA). Watson Resources, Inc., 814 Sandringham Ln., Lutz, FL 33549, (813) 949-3673.

The Computer Resources Group

Solutions Through Commitment

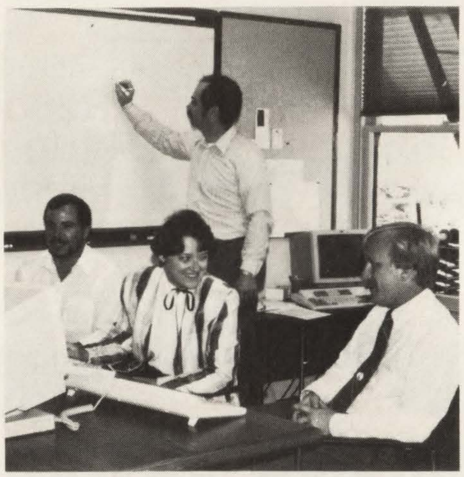
Stability and change are an apparent dichotomy of terms, yet these are the foundations upon which the growth of The Computer Resources Group is based. Formed in 1972 as a full-time placement agency with its roots in the financial communities of San Francisco, The Computer Resources Group today is a $20-million company, offering a variety of products designed to meet the needs of data processing professionals in addition to being one of Northern California's largest providers of contract computer consulting services.

Products and Services

CRG supports mainframe shops with the Information Systems Series, a documentation framework designed to help create comprehensive policies, standards, procedures and user support materials for VM and MVS data centers.

Always looking for new ways to support its clients, CRG established the Mid-Range Systems Division to support System/3X and AS/400 installations. A recent project entailed development of an E-MAIL system that will be installed on several hundred CPUs worldwide and will interface to DEC/VAX and SUN/UNIX systems. The enhancement of this system will be the basis for a new product for the AS/400. Another offering is customized, hands-on, technical training for companies preparing to migrate to the AS/400.

The Computer Resources Group provides solutions to data processing and software engineering challenges with approximately 80 percent of the business falling into the IBM/PCM category. Services range from providing consulting services for a one-week FOCUS project, to instigating an executive search for an MIS director, to managing a project consisting of 25 programmer analysts chartered with designing, coding and implementing a new system. Currently, CRG has in excess of 225 consultants on assignment in Northern California and is working with more than 100 client companies.



As an IBM Business Partner, CRG conducts customized, "hands-on" training to AS/400 clients.

Whether in a DOS, MVS or VM environment, CRG provides systems programming services to supplement its applications support. Recently, CRG performed the fastest installation of MVS/XA on the West Coast at First Nationwide Bank and, as a result, was invited by IBM to become a Business Partner.

Leadership and Philosophy

What is the key to this success? "Solutions through commitment" tells it all. A management team committed to providing both stability and opportunity for growth to its employees *plus* CRG's ability to anticipate changes in the marketplace are all part of the equation. Finally, and perhaps most importantly, CRG has always had a commitment to the highest standards of business conduct and ethics, earning both the trust and respect of its clients, consultants and candidates.

Richard D. Green, Chairman and Chief Executive Officer, is the founder of CRG and was recently called the Father of Data Processing Contract Consulting by John Dvorak of the *San Francisco Examiner.* George P. Birdsong III, President and Chief Operating Officer, joined the firm in 1979 and is largely responsible for the company's success in the contract consulting industry and its diversification into product.

Quality Assurance

A dynamic company dedicated to setting the pace for the industry, not just "keeping up," CRG ensures the level of quality service it provides. In addition to frequent client/consultant surveys, CRG has its own internal QA group, Quality Through Teamwork, dedicated to analyzing every aspect of business and making recommendations to management when possible areas of improvement are identified. Fully 95 percent of the suggestions made have been implemented, again attesting to management's openness to employees and commitment to change.

What Is Ahead?

A recent topic at a seminar hosted by CRG for its clients and consultants was "Data Processing in the 1990s — A Sneak Preview." EDI, SAA, CASE technology, 4G/Ls and Relational Database Development were presented as being the hot developments to watch. CRG has already positioned itself to support these new challenges and looks forward to many more in the future.

A commitment to stability and change: it is no dichotomy at The Computer Resources Group.

Vendor Profile is a regular forum whereby a vendor is given the opportunity to introduce the company and its products to MAINFRAME JOURNAL readers.

VSE/VTAM In A Non-Shared Address Space Really!

By Pete Clark

Perhaps the title of this should be "Virtual Storage Relief, Again" because, in essence, that is what it really is. But, this time, the scenario is somewhat different.

The idea and the original running system were done in Germany by IBM. The basic gist of the idea was to place VTAM into a private address space communicating with a CICS Terminal Owning Region (TOR) in the same address space. Then have two or more CICS Application Owning Regions (AOR) in two other address spaces communicating with the CICS TOR via Multiple Region Operation (MRO).

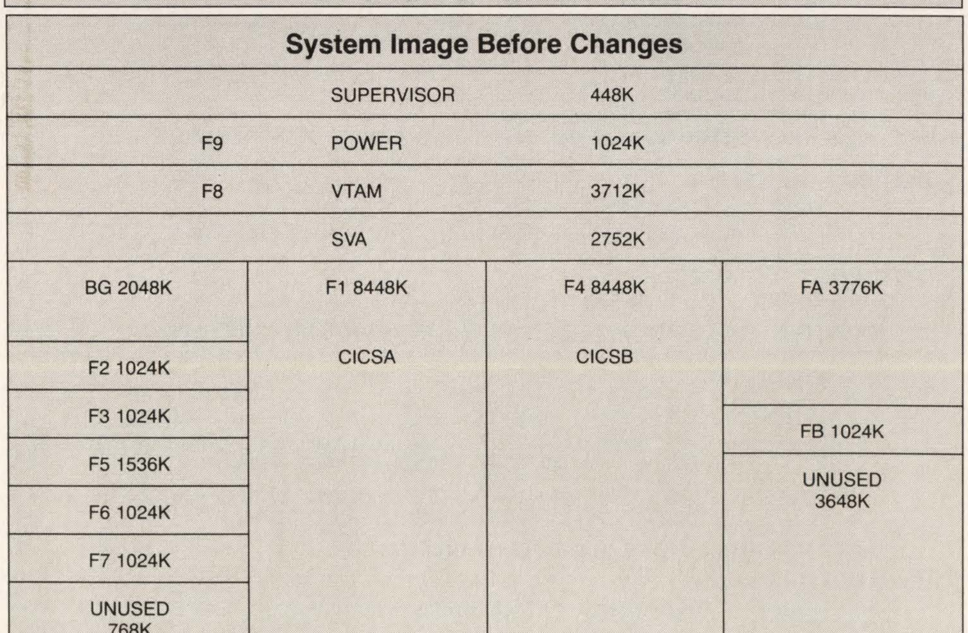
Removal of VTAM from the VSE shared area allows you to expand your private address space areas, thereby expanding the CICS areas. (See the before and after system layouts in Figures 1 and 2.) To minimize overhead, Olan Mills, Inc. decided to use transaction shipping. Various options are available, so use what is best for your environment. Basically, most transactions are defined in the TOR as remote and existing in one or the other of the AOR areas. Some transactions of course will exist in all of the CICS systems. (Examples are CEMT, CSMT and so on.)

You should only do this at the CICS 1.7 level as that is the first level of CICS/VSE that supports MRO across address spaces. With earlier versions of CICS, you could utilize Intersystem Communication (ISC) and do something similar through a communication link. Unfortunately ISC does not generally perform as well as MRO, so be careful if you decide to try this with ISC.

VSE/SP 3.2 is the preferred level of the operating system since IBM supports more than three address spaces in it. This environment will work well with any level of VSE/SP 3 and, of course, it will work with the "address space patch." *(Editor's note: The address space patch or Pete's Patch are modifications to VSE 2.1 and 3.1 that the author developed to extend addressability.)*

After implementing the initial test system in Germany, IBM enlisted the aid of two U.S. VSE customers in testing this idea in a production environment. Since Olan Mills and Carolina Steel were both storage constrained because of VTAM residing in a VSE shared address space, both were interested in participating in the test. The facility was tested and implemented

FIGURE 1



F I G U R E 2

<table>
<tr><th colspan="4">System Image After Changes</th></tr>
<tr><td colspan="4">SUPERVISOR 448K</td></tr>
<tr><td colspan="4">F9 POWER 1024K</td></tr>
<tr><td colspan="4">SVA 2816K (SEE NOTE1)</td></tr>
<tr><td rowspan="2">F1 2752K
CICSATOR</td><td rowspan="6">F2
12096K

CICSAAOR</td><td rowspan="6">F3
12096K

CICSBAOR</td><td>F4 1024K</td></tr>
<tr><td>F6 1024K</td></tr>
<tr><td>F5 1536K
BIMWNDOW</td><td>F7 1024K</td></tr>
<tr><td>BG 1664K
(NOTE2)</td><td>FA 3776K</td></tr>
<tr><td rowspan="2">F8 6144K
VTAMA</td><td>FB 2048K</td></tr>
<tr><td>UNUSED
3200K</td></tr>
</table>

F I G U R E 3

```
DFHSITOA FOR THE CICSA TOR
        DFHSIT   TYPE=CSECT,
                 SUFFIX=OA,
                 APPLID=CICSA,
                 ISC=YES,
                 IRCSTRT=YES,
                 SYSIDNT=TOR,
DFHSITAA FOR THE CICSA AOR
        DFHSIT   TYPE=CSECT,
                 SUFFIX=AA,
                 APPLID=CICSAORA,
                 ISC=YES,
                 IRCSTRT=YES,
                 SYSIDNT=AORA,
DFHSITAB FOR THE CICSB AOR
        DFHSIT   TYPE=CSECT,
                 SUFFIX=AB,
                 APPLID=CICSAORB,
                 ISC=YES,
                 IRCSTRT=YES,
                 SYSIDNT=AORB,

DFHPPTOA FOR THE CICSA TOR
         DFHPPT   TYPE=GROUP,. . . ,FN=ISC
     NOTE: ONLY PROGRAMS THAT ARE TO BE RUN IN THE TOR GO IN THIS TABLE

DFHPPTAA FOR CICSA AOR
         DFHPPT   TYPE=GROUP,. . . ,FN=ISC,
     NOTE: ONLY PROGRAMS THAT ARE TO BE RUN IN THIS AOR GO IN THIS TABLE

DFHPPTAB PPT FOR CICSB AOR
         DFHPPT   TYPE=GROUP,. . . ,FN=ISC
     NOTE: ONLY PROGRAMS THAT ARE TO BE RUN IN THIS AOR GO IN THIS TABLE

DFHTCTOA FOR THE TOR
         DFHTCT   TYPE=INITIAL,. . . ,SYSIDNT=TOR
         DFHTCT   TYPE=SYSTEM,SYSIDNT=AORA, NETNAME=CICSAORA,ACCMETH=IRC,
                  RECEIVE=(RA, 03),SEND =(SA,40),USERSEC=IDENTIFY
         DFHTCT   TYPE=SYSTEM,SYSIDNT=AORB,NETNAME=CICSAORB,ACCMETH=IRC,
                  RECEIVE=(RB,03),SEND=(SB,40),USERSEC=IDENTIFY
     NOTE: NO CHANGES TO TERMINAL DEFINITIONS FOR TOR
DFHTCTAT FOR CICSA AOR
         DFHTCT   TYPE=INITIAL,. . . , SYSIDNT=AORA
```

Figure continued

successfully at both sites.

Of course during the initial testing, some problems were encountered in several IBM software product arenas. However, during this test a signifiant situation occurred. Almost without exception, every IBM person and IBM product group contacted immediately started addressing any problems that were identified. It has been a gratifying experience being a part of this consolidated effort involving customers, IBM Germany, IBM U.S. and IBM International Trade Support Center (ITSC). It is my personal hope that this type of effort and cooperation becomes the normal process in addressing problems of this type.

To all of you who have been a part of this project, thanks and may customers and vendors always work so well together.

Details

First and foremost, from the Olan Mills side most of the work was done by Galen Smith. Galen has provided examples that are included as part of this article (see Figure 3). These examples of CICS tables document the additional entries required in the CICS tables for the MRO environment. Most are self-explanatory. If not, they should be with a little reading from the appropriate CICS manual. Also included is a temporary fix for a problem with the ASSIGN command and a system layout of the system before and after with storage sizes.

Insert the appropriate entries in the CICS tables, compile and catalog. If you wish to redistribute transactions among the CICS systems, do this as you are changing the tables.

Make any required changes in the IPL procedure to reallocate storage and to move partitions into the appropriate address spaces. The following three phases must be put in SVA for MRO support:

- DFHIRP 12120 bytes
- DFHSCTE 8 bytes
- DFHCSEOT 216 bytes.

DFHIRP uses SVA GETVIS for data buffers and control blocks. We saw an SVA increase of 25K for these phases and buffers. Create the additional JCL for the new CICS TOR and submit it to the reader queue. IPL the system and bring up the CICS systems and VTAM.

The original intention of this technique was not to replace the requirement of getting VTAM out of shared, but rather for it to be used as a short-term relief until that requirement could be accomplished.

```
         DFHTCT    TYPE=SYSTEM,ACCMETH=IRC,SYSIDNT=TOR,NETNAME=CICSA,
                   RECEIVE=(RA,40),SEND=(SA,03),USERSEC=IDENTIFY,
                   OPERSEC=(64,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,
                   18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,
                   36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,
                   54,55,56,57,58,59,60,61,62,63)
A108     DFHTCT    TYPE=REMOTE,. . . ,SYSIDNT=TOR
     NOTE: ALL BTAM DEVICES ARE IN THIS AOR (OUR CHOICE)

DFHTCTAB FOR CICSB AOR
         DFHTCT    TYPE=INITIAL,. . . ,SYSIDNT=AORB
                   SYSIDNT=AORB
         DFHTCT    TYPE =SYSTEM, ACCMETH=IRC,SYSIDNT=TOR,NETNAME=CICSA,
                   RECEIVE=(RB,40),SEND=(SB,03),USERSEC=IDENTIFY,
                   OPERSEC=(64,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,
                   18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,
                   36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,
                   54,55,56,57,58,59,60,61,62,63)
A108     DFHTCT    TYPE=REMOTE,. . . ,SYSIDNT=TOR

DFHPCTOA FOR CICSA TOR
         DFHPCT    TYPE=GROUP,. . . ,FN=ISC,
         DFHPCT    TYPE=REMOTE (WILL ROUTE THIS TRANSACTION TO CICSA AOR)
                   SYSIDNT=AORA
         DFHPCT    TYPE=REMOTE (WILL ROUTE THIS TRANSACTION TO CICSB AOR)
                   SYSIDNT=AORB
         DFHPCT    TYPE=ENTRY (THIS TRANSACTION WILL RUN IN TOR)

DFHPCTAA FOR CICSA AOR
         DFHPCT    TYPE=GROUP,
                   FN=ISC
         DFHPCT    TYPE=ENTRY (NO CHANGE TO PCT FOR AOR)

DFHPCTAB FOR CICSB AOR
         DFHPCT    TYPE=GROUP,. . . ,FN=ISC,
         DFHPCT    TYPE=ENTRY (NO CHANGE TO PCT FOR AOR)
```

However, as we progressed further into the project, we realized that perhaps some customers might have reason to remain in this environment even if VTAM is delivered to reside in a private address space.

Galen comments on the complexity of the installation, "The documentation *(VSE/SP Installation of Large CICS Partitions — GC24-3332)* provided by IBM made the CICS table changes straight forward and easy — significantly easier than using the *CICS Inter-Communications Guide*."

It is our understanding that IBM will include this technique in the documentation for VSE/SP 4.1 to be available in July 1989. Additionally, various skeletons will be supported for this environment utilizing the Interactive User Interface.

Olan Mills' basic installation occurred over a three-day period in an interruptible environment. Galen had frequent interruptions to answer questions and perform other tasks. Then the fun started as we began testing the environment looking for potential problems.

Problem 1

In our application programs we access TCT fields, specifically: TERMTYPE, PRINTTO, TRANSID and PAGESIZE. If you are also doing this, you should be aware of a change to this environment. Under CICS/MRO there are three TCT entries: skeleton, surrogate and model. Most of the information needed is contained in the surrogate that is pointed to by the skeleton. We elected to obtain the PRINTTO from our Auto Install Terminal Table. The other three fields were obtained by changing the application programs to determine if the program were looking at a skeleton. If this is true for you, retrieve the information for the surrogate TCT entry by using the surrogate address contained in the skeleton.

Problem 2

In this environment, RJE/SNA will hard wait the system. If you are operating in this environment, do not PSTART RJE,SNA. IBM is providing a fix to resolve this problem. The same fix will resolve the problem with PNET. APAR number DY37989 for POWER/VSE involving module IPW$$SN resolves the hard wait.

Problem 3

The CICS ASSIGN command produces different results in a CICS/MRO application owning area than when executing in a standard, non-MRO CICS. Galen has addressed this problem with a patch contained in this article (see Figure 4). The original CICS response to this issue was "functioning as designed." I really doubt that anyone actually designs a system to produce different results in mirror environments. Needless to say, no one was particularly impressed with that response. After IBM took the issue under review, the reply changed to "this is not correct; we should fix this." I agree with that, but so far we have not seen any code.

Problem 4

Various problems were resolved by installing the following PTF list. Several of these problems revolve around CICS security. Note that not all the PTFs on this list in Figure 5 are specifically for CICS/MRO, but they do resolve problems found during testing and probably should be installed to stabilize your system. By the time this article is in print, I am sure several of these will have been superceded. Apply the latest version.

Problem 5

If, in some environments, someone should try to perform a CEMT SET VTAM OPEN command in a CICS AOR that is not in the same address space as the TOR, the system may take a hard wait FFF. Note that in this environment, the TOR and AOR are not normally in the same address space. Since there are no actual real VTAM terminals in this area, no one should have issued the command! While working with IBM to identify and resolve this problem, the problem disappeared. We had installed some additional maintenance bringing the system to a high level of VSE/SP 3.2 when the problem could not be recreated. However, as of recent testing on May 30, 1989, this problem has recurred.

Problem 6

An exception to normal CEDF operation is CEDFing a transaction from the AOR and outputting to a printer in the TOR. This produces a message "EDF mode cannot be modified from this terminal" and prohibits CEDF controlled execution. A requirement has been submitted to IBM to correct this situation.

Problem 7

We have a menu program that transfers control to other application programs using CICS XCTL. This works in the non-

FIGURE 4

```
  JOB DY99017
*  ALTER DFHEEI SO THAT THE EXEC CICS ASSIGN OPERKEYS {OPSECURITY}
*  WORKS THE SAME IN AN MRO ENVIRONMENT AS IT DOES IN A STAND-ALONE
*  ENVIRONMENT. THIS PATCH IS ONLY FOR CICS DOS 1.7
*  IT WILL TAKE SECURITY KEYS FROM THE DFHSNNT INSTEAD OF DFHTCTTE
OPTION CATAL,LOG
EXEC MSHP,SIZE=640K
CORRECT 5746-XX300-0A69 : DY99017
AFFECTS PHASE=DFHEE I
ALTER 0512 —
  OA5C:068E
*                                                                          */
*  THIS MAKES OPERKEYS REQUEST GO TO OPSECURITY ROUTINE                    */
*  DC AL2(EIEIAS94-EIEI00): DC AL2(EIEIAS67-EIEI00)                        */
*                                                                          */
ALTER 06BE —
  D2023000A028:47F09A5E0700
*                                                                          */
*  THIS CHANGES OPSECURITY ROUTINE TO BRANCH TO OPERKEYS                   */
*  ROUTINE AFTER ESTABLISHING ADDRESSABILITY TO THE TCTTE                  */
*  MVC O(L'TCTTESK,RARG1),TCTTESK : B EIEIAS94+2                           */
*                                                                          */
ALTER OA7C —
  9101C00847809C0058A0C00854A09D20:0058D20430009A541BFFBFF7A05D4780
ALTER OA8C —
  D20430009A541BFFBFF7A05D47809A82:9A8849509A5C47709A82D2083000F010
ALTER OA9C —
  D2043000F010D2023005A028:07FE07000700D2023000F015
*                                                                          */
*  THIS CHANGES OPERKEYS ROUTINE TO SETUP ADDRESSABILITY                   */
*  TO SNTT AND GET THE SECURITY KEYS 24-64 FROM SNTTSKE IF                 */
*  OPERKEYS WAS REQUESTED AND GET SECURITY KEYS 1-23 FROM                  */
*  SNTTSK IF OPERKEYS OR OPSECURITY WAS REQUESTED.
*  EIEIAS94 DS OH                      : EIEIAS94 DS OH                    */
*  TM     TCAFCI,TCAFCTRM              : DC     H'0058'                    */
*  BZ     EIEIASVO                     : MVC    O(L'SNTTSKE,RARG1,EIEIASBL */
*  L      TCTTEAR,TCAFCAAA             : SR     RF,RF                      */
*  N      TCTTEAR,=X'OOFFFFFF'         : ICM    RF,B'0111',TCTTESNT+1      */
*  USING  DFHTCTTE,TCTTEAR             : USING  DFHSNNT,RF                 */
*  MVC    O(L'SNTTSKE,RARG1),EIEIASBL  : BZ     EIEIAS9E+6                 */
*  SR     RF,RF                        : CH     R5,EIEIAS94                */
*  ICM    RF,B'0111',TCTTESNT+1        : BNE    EIEIAS9E                   */
*  USING  DFHSNNT,RF                   : MVC    O(L'08,RARG1),SNNTSKE      */
*  BZ     EIEIAS9E                     : BR     FE                         */
*  MVC    O(L'SNNTSKE,RARG1),SNNTSKE   : NOOP                              */
*                                      : NOOP                              */
*  DROP   RF                           : DROP   RF                         */
*  EIEIAS9E DS OH                      : EIEIAS9E DS OH                    */
*  MVC    5(L'TCTTESK,RARG1),TCTTESK   : MVC    O(L'TCTTESK,RARG1),SNTTSK  */
*                                                                          */
RESOLVES 'CICS/DOS 1.7 EXEC CICS ASSIGN IN AN MRO ENVIRONMENT'
```

FIGURE 5

DY99017	5746-XX300-A69	LOCAL	02/03/89	*** CICS/DOS 1.7 EXEC CICS ASSIGN
UL36358	5746-XX300-A69		02/07/89	installed
UL37299	5746-XX300-A69		01/11/89	installed
UL37684	5746-XX300-A69		01/11/89	installed
UL38948	5746-XX300-A69		02/07/89	installed
UL39270	5746-XX300-A69		01/11/89	installed
UL39506	5746-XX300-A69		01/11/89	installed
UL39601	5746-XX300-A69		01/11/89	installed
UL39774	5746-XX300-A69		01/11/89	installed
UL40315	5746-XX300-A69		01/11/89	installed
UL41743	5746-XX300-A69		01/11/89	installed
UL43125	5746-XX300-A69		01/27/89	installed
UL43781	5746-XX300-A69		01/27/89	installed
PL35245	5746-XX300-A69			terminating task problem
PL38924	5746-XX300-A69			initial security violation
DY37989	5666-27301-A92			POWER/RJE APAR

MRO environment. For MRO, this transaction was moved to the TOR and needed to transfer control to multiple AORs. XCTLs would require PPT program entries in two or more CICSes. In some cases, this would not produce the desired results. To resolve this problem, the menu program was converted to do START TRANSACTIONS but, alas, a new problem came up. The mirror, transaction CSMI program DFHMIR, which is invoked by a start transaction, expects a terminal to already be attached. Since it is not an ABEND, AISD occurs. CICS in program DFHMIR should, after determining that a terminal is not available, perform an ATTACH for the auto-install environment. Currently an MVS APAR directed at resolving this problem intercepts the ABEND back in the TOR, does an attach and reinitiates the transaction. This APAR PL27133 has not at this time been made available for the VSE environment, but we are pursuing it. A trace of the actions of the CICS AOR and TOR during this process will furnish you with some interesting conversation areas for a few days. For example, "Why is CICS doing a conditional load of DFHSNT?"

Pluses And Minuses

Performance has been a pleasant surprise, as it did not noticeably change. I suspect that any system that has available CPU cycles will not experience any noticeable increase in response times. Most of our CICS transaction response time is in servicing I/O requests, not in CPU processing cycles. CICS/MRO implementation needs CPU cycles and does not perform any I/O. We had no noticeable paging before the implementation and have not noticed any change after implementation.

I suspect, if you are running 95 percent CPU 20 pages a second and implement this environment, you would see response slowdown. Conversely, if you are running 65 percent CPU one page a second and implement this environment, no one will realize the change occurred. CPU cycles will certainly increase depending on machine size and environment. Paging, of course, depends on how much you expand your virtual environment and your available real memory size. Paging induced just by the increased size of the MRO support would probably not be noticed.

We are now able to support a "single image look" to all of our CICS users irrespective of where their applications or combination of applications reside. Applications and data may be redistributed among CICS AORs without training, notifying or bothering users.

We have gained approximately 3.6MB of virtual storage for each of our two production CICS partitions. We have gained approximately 2.5MB of virtual storage for VTAM for improved recoverability and future terminal growth. With the additional VTAM storage, we will be able to reimplement auto logon, auto recovery and the CICS good morning message.

Of course, we are now consuming one

more partition than before, but that just gives us additional justification for needing more partitions.

It is necessary to include any VTAM applications natively accessing VTAM in the VTAM address space (that is, CICS TOR, VTAM windowing products, NetView and so on).

We found it advantageous to place some "highly active transactions" and "non-typical user applications" in the TOR (that is, editor transactions, transactions that access spooler files or system programming type data areas and so on).

You will need to develop a long and lasting relationship with the CICS routing transaction CRTE. This allows terminals to route transactions to specific CICSes when there are local AOR transactions. An example is the CEMT transaction that could be executed on any CICS. CRTE SYSID=xxxx is done first to determine which CICS to route the transaction to. The CANCEL command will stop the routing.

Another approach is to define a different TRANSID with the same program entry in the TOR for each individual AOR (that is, CEMA for CEMT on CICSA, CEMB for CEMT on CICSB). Keep in mind that hard coding a CICS RETURN TRANSID in an application program may not be a good idea. Retrieve the TRANSID from the EIB. Programs that place the transaction code in the upper left of the screen for use when the user hits enter may also be a problem, since the TRANSID becomes CICS AOR dependent.

CRTE should do an "automatic sign-on" similar to a normal remote transaction rather than requiring a sign-on for the first transaction after the CRTE transaction. Or perhaps, in some environments, all AOR defined transactions could be security code 1 (that is, default security and sign-on might not be required — you decide).

Of course, you will probably now have additional CICS tables or additional CEDA entries to maintain and the two-character table suffix may be somewhat confining. Also, you are now supporting an additional CICS with its attendant files — JCL and entries.

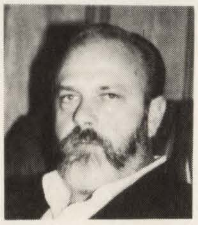
The more "vanilla" your CICS arena, the easier the task of moving to this environment. An implementation in a "vanilla" arena could be accomplished in a few days with little effort and planning. For the larger VSE user, the complexity of your environment will determine the necessary time and effort for full implementation.

I realize that there should be additional information for users of ICCF and the Interactive User Interface (IUI). Neither of the two U.S. test sites are using these products. Sorry that we cannot be of more assistance in these areas.

Is it a good temporary solution? Yes it is and, perhaps, for some it may be a more lasting solution. For us, we are looking forward to true VTAM Private Address Space support. During the interim this solution will allow us to continue to add terminals and applications.

Good luck with your implementation of VSE/VTAM in a private address space.

ABOUT THE AUTHOR



Pete Clark has been in data processing for 27 years, the last 13 with Olan Mills, Chattanooga, TN. Clark is a recognized authority on the VSE operating system. He has worked many years to extend the limits of VSE.

GFIS

GFIS *from page 35*

GPG can also be used for simple document maintenance in a stand-alone environment without the aid of GFIS. However, GPG is primarily designed to be used in conjunction with GFIS.

GFIS is a vital link in TNMP's ability to pinpoint quickly and accurately information regarding its facilities, location of the facilities and their relationship to each other. Ivy told us that before he came to TNMP in 1983, the company used a manual map system with individuals drawing the maps by hand. This archaic procedure proved to be costly with an obvious tendency to be inaccurate.

On Ivy's IBM 5081 Graphics Display Terminal red, blue and green lines criss-cross the screen indicating power lines, new additions and retirements which can easily be read by a construction crew on the printed map. Each map clearly indicates streets, lots, location codes for transformers, poles, meters, serial numbers and phase connections.

In describing the advantages of GFIS Ivy says, "We can draw a map of all the streets and show how far the poles (the circles indicate where the poles are placed) are from the edge of the street, show the lots, the easement and even if there are overhead guywires or downed guywires." After each map is updated, Ivy and his co-workers can then load it onto the IBM 6186 eight-pen plotter. The plotter is tied into the workstation through RS232 cables. It can run continuous paper or 8½ x 11 inch paper. The plotter automatically recognizes which type of paper has been inserted.

All maps are created, copied and distributed from the Fort Worth office. Heinrich informed us that because of the quantity of maps run through the Fort Worth office, TNMP will be replacing the IBM plotter with a CALCOMP electrostatic plotter.

Always on the lookout for new and better systems, Ivy says there is nothing available right now to take the place of the present system, but "with technology changing daily, more cost-effective hardware and software will, I am sure, be introduced in the near future."

ABOUT THE AUTHOR

Cecilia Coburn Perry is a freelance writer in Dallas, TX specializing in research and high-technology topics.

Memory-Resident Data *from page 63*

provides no automatic protection against concurrent updates, since this is an application defined and managed data area.

The data in the table is updated simply by moving new values to it. There is virtually no overhead involved in this. Of course, I/O overhead is incurred to keep the KSDS in sync with the on-line changes. When the updates are complete, a DEQ is issued.

It is sometimes necessary to refresh a shared data area completely. In this case, code such as that shown in Figure 13 is required.

To refresh or replace an application defined area in the shared subpool, the old area must be FREEMAINed and a new area GETMAINed. However, inflight tasks must not be corrupted. CICS has no automatic wait mechanism to prevent inflight tasks from accessing the table while it is being refreshed. The application is completely responsible for managing the area.

In Figure 13, an ENQ command is issued to prevent two tasks from attempting to refresh the area concurrently. The CWA is addressed and the pointer to the shared area is saved in WORKING-STORAGE.

At this point, a new area is obtained and loaded from the KSDS in exactly the same manner as in Figure 10. When this process is complete, the CWA contains a pointer to the new data area. The enqueue is no longer needed and a DEQ command is issued.

In case any inflight tasks are still accessing the old data area, a DELAY command is issued before the old area is freed. In this example, wait six seconds. This should be plenty of time for any inflight task to terminate. Finally, the pointer reference is set to the address of the old area that was previously saved in WORKING-STORAGE and a FREEMAIN command is issued.

Variations

The three methods discussed in this article are the most widely used for managing memory-resident data in CICS applications. Many variations on these methods are possible, such as:

- Storing the pointer to a shared data area in a TS main queue rather than in the CWA
- Issuing a single LOAD/HOLD command during post initialization and storing the module's address in the CWA or in TS for reference by subsequent tasks
- In some cases, it may be permissible to store a small amount of data directly in the CWA rather than just a pointer.

Summary

The decision to maintain data in memory should be made during the design phase of an application based on anticipated workload, service level requirements and resource limitations. The technique or techniques to be used for managing the resident data should be chosen to exploit the strengths of each method; no single method is always best.

ABOUT THE AUTHOR

David Nicolette is an independent computer consultant based in the Dallas, TX area and is President of Nicolette Consulting, Inc., 114 Creekside Lane, Coppell, TX 75019, (214) 471-1604.

Capture Ratios *from page 49*

F I G U R E 1

Sample Regression Report

DEP VARIABLE: DELCPU
ANALYSIS OF VARIANCE

SOURCE	DF	SUM OF SQUARES	MEAN SQUARE	F VALUE	PROB>F
MODEL	3	3580655.95	1193551.98	410.110	0.0001
ERROR	145	421996.21	2910.31872		
C TOTAL	148	4002652.17			

ROOT MSE	53.94737	R-SQUARE	0.9346
DEP MEAN	384.1171	ADJ R-SQ	0.9324
C.V.	14.04451		

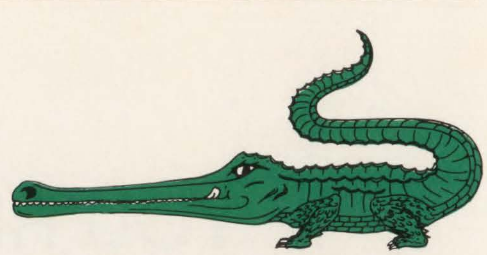
PARAMETER ESTIMATES

VARIABLE	DF	PARAMETER ESTIMATE	STANDARD ERROR	T FOR H0: PARAMETER=0	PROB > :T:
INTERCEP	1	47.160187	2.87902767	14.941	0.0001
SWAPS	1	0.05196499	0.005381872	9.656	0.0001
EXCPS	1	0.000496373	0.000027578	17.999	0.0001
PAGES	1	0.000112948	0.000048625	3.023	0.0013

A P P E N D I X 1

```
*                                                              ;
* COPYRIGHT PERFORMANCE ASSOCIATES, INC. 1987,88               ;
*                                                              ;
* PERFORMANCE ASSOCIATES, INC.                                 ;
* 72-687 SPYGLASS LANE                                         ;
* PALM DESERT, CA 92260                                        ;
*                                                              ;
* (619) 346-0310                                               ;
*                                                              ;
* MEMBER: ESTPATHL                                             ;
*                                                              ;
```

```
DATA TYPE70 (KEEP=CPU70SEC SYSTEM STARTIME);
  SET PDB.TYPE070;
  CPU70SEC=DURATM*NRCPUS-CPUWAITM;
PROC SORT; BY SYSTEM STARTIME;
DATA TYPE72;
  SET PDB.TYPE72;
  IF PERFRPGN^=. THEN DELETE;
PROC SORT; BY SYSTEM STARTIME ;
DATA TYPE72 (KEEP=SWAP PAGE EXCP CPU72TCB CPU72SRB SYSTEM STARTIME);
  SET TYPE72; BY SYSTEM STARTIME;
  RETAIN SWAP PAGE EXCP CPU72TCB CPU72SRB 0;
  CPU72SRB+CPUSRBTM;
  CPU72TCB+CPUTCBTM;
/*                                                              */
/* PGAPIN IS AN MVS/XA 2.2 DEPENDENT VARIABLE                   */
/*                                                              */
  PAGE+PGPAGEIN;
  SWAP+SWAPSEQ
/*                                                              */
/* PGAITS IS THE NUMBER OF I/O SERVICE UNITS, TO CALCULATE THE  */
/* NUMBER OF I/OS, YOU MUST DIVIDE BY THE SERVICE DEFINITION    */
/* COEFFICIENT TO GET BACK TO THE I/O COUNT. FOR THIS TO WORK,  */
/* YOU MUST HAVE SPECIFIED IOSRVC=COUNT.                        */
/*                                                              */
  EXCP+(IOUNITS/IOCCOEFF);
  IF LAST.STARTIME THEN DO;
    OUTPUT;
    CPU72SRB=0;
    CPU72TCB=0;
    PAGE=0;
    SWAP=0;
    EXCP=0;
  END;
DATA TYPE7072 (KEEP=SYSTEM STARTIME CPUDELTA CPU70SEC
                CPU72SRB PAGE SWAP EXCP);
  MERGE TYPE70 (IN=T70) TYPE 72 (IN=T72);  BY SYSTEM STARTIME;
  IF LAST.STARTIME THEN DO;
    CPUDELTA=CPU70SEC-CPU72SRB-CPU72TCB;
    IF T70 & T72 THEN OUTPUT;
  END;
PROC REG DATA=TYPE7072;
  MODEL CPUDELTA=PAGE SWAP EXCP; BY SYSTEM;
```



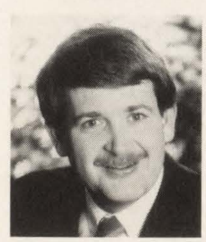
is added to the TCB and SRB time recorded for the workload, the result is a reliable estimate of the total CPU time for the workload. These values can be used for capacity planning, in performance studies to determine the true impact of a workload or in job accounting to allocate costs more fairly.

Remarks

The notion of a capture ratio is far too simple a concept to explain the complex interactions between a workload, the operating system and other competing workloads in a system. However, the new measurement variables provided by RMF for MVS/XA systems allow the path-length estimates to be calculated for critical operating system services like paging, swapping and I/Os. Using these path-length estimates, CPU overheads can be allocated in a much more reliable manner than techniques based on capture ratios for average workloads. While any estimation technique is subject to error, these path-length estimates can be used to more accurately attribute CPU overhead for accounting, capacity planning and performance studies. Until MVS actually measures and reports CPU time at the task level for operating system services, path-length estimation provides a significant comparative advantage over silver bullet capture ratios for allocating CPU overheads.

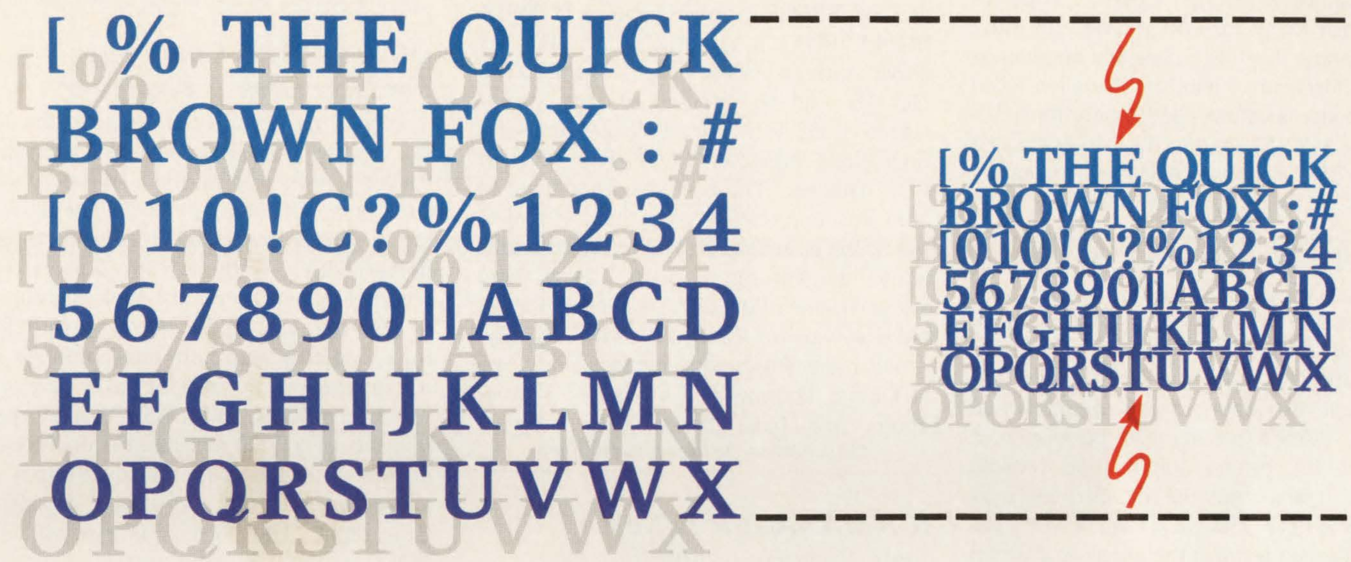
One last thought, I don't really believe that there are alligators in the New York City sewers, but that is a decision you will have to make for yourself. If you are ever down there under the city and see one, be sure to call!

ABOUT THE AUTHOR



H. Pat Artis is a recognized authority in workload characterization, forecasting simulation modeling and capacity planning. He is the author of more than 75 papers and lectures internationally. Artis is President of a private consulting firm, Performance Associates, Inc., 72-687 Spyglass Ln., Palm Desert, CA 92260, (619) 346-0310.

DATA COMPRESSION FOR DB2



With INFOPAK, this simple concept is finally simple to use

Using data compression to improve DASD storage efficiency is a technique that has traditionally been simple in concept but difficult in execution. INFOPAK from InfoTel Corporation has changed all that.

With INFOPAK you can recover as much as 70% of the space on your present disks. You can do it without compression tables and without the data analysis required to generate them. You can do it without interfering with application programs, since INFOPAK is totally transparent. And you can do it whether you're running DB2, VSAM, IMS, IDMS, or DATACOM/DB.®

INFOPAK achieves its initial compression using regular load and unload utilities. No modifications to existing programs or JCL are required and INFOPAK may be installed by your database administrator in a few easy steps.

In addition to improved DASD utilization, INFOPAK delivers important performance benefits. As significant compression yields significant I/O reductions, you will see marked improvements in response time and throughput.

TEST IT FOR YOURSELF!

Ask about our free TESTPAK program! Evaluate the data compression and performance advantages of INFOPAK against *your own* data files.

Get a realistic look at the improvements INFOPAK can make in your DASD space utilization. Ask for a free TESTPAK program. Run it against your existing data bases to determine the DASD space gains you can expect.

Contact InfoTel today!

InfoTel

InfoTel Corporation
14906 Winding Creek Court
Suite 101D
Tampa, FL 33613
Phone 800/543-1982 or
813/264-2090
FAX 813/960-5345

The Monitor For MVS Announced

Landmark Systems Corp., primarily known for its successful software product The Monitor For CICS, has just announced the development of another "Monitor." The new product is called The Monitor for MVS (TMON/MVS). Among the capabilities of TMON/MVS are: a real-time exception monitor, an I/O subsystem monitor, a delay monitor and a historical performance database with a report writer. In addition, TMON/MVS provides an on-line source of recent-past information across the complete operating system. It monitors single or multiple CPUs including: PR/SM configurations; real, auxiliary, expanded and virtual storage; DASD activity and performance; tape activity; printer activity and console activity. It also provides detailed job execution and delay analysis. TMON/MVS has been designed to meet the needs of a varied audience, from managers to systems programmers, performance analysts, applications developers and operators.

For more information contact Claudia Houston at Landmark Systems Corp., Vienna, VA, (800) 227-8911 or:

Circle #200 on the Reader Service Card

CICS SIMULCAST Improves Training & Support

CICS SIMULCAST, from Compuware Corp., is a new product designed to improve training and support for users of IBM's CICS products. With its two separate but complementary functions, Broadcast and Conference, CICS SIMULCAST provides facilities which enable an organization to provide quality user training and support with reduced effort and travel costs. The Broadcast function allows an organization to conduct training sessions for users located anywhere in the world. The Conference function allows help-desk and support personnel to view screens of individual users encountering application problems. CICS SIMULCAST features batch utilities for printing and dataset controls as well as on-line tutorials and help.

For more information contact Larry Fees at Compuware Corp., Farmington Hills, MI, (313) 737-7300 or:

Circle #201 on the Reader Service Card

DTA/QCOPY — The New VM/POWER Report Utility

DTA/QCOPY is a new software package from Davis, Thomas & Associates which transfers, archives, re-segments and re-sequences reports from any combination of VM/SP Spool Queues, POWER POFFLOAD tapes and DTA/QCOPY Archive datasets. The output from DTA/QCOPY can be directed to any combination of DTA/QCOPY Archive datasets, microfiche tapes or POWER's LST or PUN Queues. The user does not have to alter any application programs. It makes use of the new IBM XPCC Interface which allows users to run as many jobs as there are partitions in the CPU. Users no longer have to wait for their jobs to run serially.

For more information contact Don Caba at Davis, Thomas & Associates, Minneapolis, MN, (612) 591-6100 or:

Circle #202 on the Reader Service Card

ICS/MANAGER Is New Data Security Umbrella

The primary objective of ICS/MANAGER, from Integrity Solutions, Inc., is to automatically register, extract and manage all critical backup and journal information on a global Control Dataset (CDS). It serves as a single repository of all essential elements required to reconstruct lost or corrupted data. In addition, ICS/MANAGER provides the capability of automatically generating and submitting customized backup, journal and recovery JCL, saving valuable time and resources.

For more information contact Jim Grimm at Integrity Solutions, Inc., Littleton, CO, (303) 794-5505 or:

Circle #203 on the Reader Service Card

Tape Dataset Stacking Utility Maximizes Tape Efficiency

US WEST Communications Software Sales now markets TDSU (Tape Dataset Stacking Utility). Studies show that 80 percent of all datasets use less than half a reel of tape. After datasets have been created and verified, TDSU will stack tape datasets with selected similarities. This process reduces the cost of the number of tapes needed in the data center. Further benefits include less space required for both archive and on-site storage, less tape handling and an easy, cost effective transition from tape to cartridge. TDSU also generates audit trails detailing all changes.

For more information contact Angela Silva at US WEST Communications Software Sales, Denver, CO, (303) 896-9258 or:

Circle #204 on the Reader Service Card

QUICK-TALK Conference For Help Desk Automation

IBS Corp. recently announced the latest version of QUICK-TALK Conference for help desk automation. This new version now includes two new major features: Screen Image Recorder/Archive and SHOWKEY. With Screen Image Recorder/Archive, users now have the choice of recording all screens being viewed during a conference or help desk situation on a temporary or permanent basis. It works just like a VCR. With SHOWKEY, participants of a conference can actually view all the information being keyed/input by a user prior to the application accepting it as input. The product is available for MVS and ESA installations.

For more information contact Cindy Wheat-Roberts at IBS Corp., San Diego, CA, (800) 346-2894 or:

Circle #205 on the Reader Service Card

SPICE - New Productivity Aid For IMS Users

SPICE, said to be the first of its kind for the IMS market, is designed to emulate and extend IMS/VS symbolic checkpoint restart and cut the costs of developing and operating restartable applications. Westinghouse Management Systems Software recently entered the IMS software market with SPICE along with BEARS/IMS, a powerful IMS monitoring tool. Checkpoint restart programs using SPICE Checkpoint Control can checkpoint at times appropriate to the logic of the application, rather than to the IMS/VS environment in which they run. Consequently, the programmer can avoid much of the messy control logic normally associated with IMS/VS symbolic checkpoint restart.

For more information contact Mark Potenzone, Westinghouse Management Systems Software, Pittsburgh, PA, (412) 256-6183 or:

Circle #206 on the Reader Service Card

ProTerm, The Systems Utility Platform For MVS

Sequel Corp.'s ProTerm, the systems

utility platform for MVS, is used for regression testing, system stress and load, as well as benchmark testing. The platform includes a variety of other systems professional services and functions including enhancements to TSO and ISPF. The latest version of ProTerm (Version 1.3) includes VSAM KSDS file support, a CALL statement which allows direct calls to any MVS application including all native database calls. It now has direct support for all databases and all TP monitors.

For more information contact Lyle Henry at Sequel Corp., Oklahoma City, OK, (800) 776-8376 or:

Circle #207 on the Reader Service Card

Significantly Enhanced Version of CA-1 MVS Tape Management System

The latest release (Release 4.9) of Computer Associates' CA-1 tape management system for the MVS operating system addresses the important issue of long-term tape catalog maintenance by extending limited support for tape expiration dates into the 21st century. Also featured are a number of ISPF enhancements providing expanded operator screens and allowing the definition and maintenance of scratch tape sub-pools via ISPF panels. Other improvements to CA-1 include the updating of the TMSGRW Generalized Report Writer and the addition of the PARM=TEST utility program option which allows for simulation of a program prior to its actual execution.

For more information contact Dan Michaelis at Computer Associates International, Inc., Garden City, NY, (516) 227-3300 ext. 7027 or:

Circle #208 on the Reader Service Card

TIMS Provides End-User Support Information System for IMS/DC & CICS

TIMS, from 4.ST North America, Inc., is an on-demand help, support and documentation facility that can be accessed from any IMS or CICS transaction. It provides a ready-made standard framework for defining and displaying specific, in-context help, support and documentation for new and existing transactions at the application, transaction, screen and field levels. Information systems become more user friendly and require less user training and support because TIMS can provide information as to what data values can be entered and how they can be entered.

For more information contact Fred Geisendorff at 4.ST North America, Inc., Gretna, LA, (504) 366-9945 or:

Circle #209 on the Reader Service Card

HELPVTOC Consolidates VSAM & Non-VSAM File Information

Smartech Systems, Inc. has announced availability of HELPVTOC, a VSE system utility that condenses all VSAM and non-VSAM file information into five easy-to-read, customizable reports. HELPVTOC combines the information from both VTOC and LISTCAT into a simple, easy-to-read format. This gives users a concise resource to go to when looking for specific information, rather than having to pour through thick multiple-file listings. Report data is made available to the user so that installation-designed programs can be written to produce customized reports. Up to five reports can be produced with HELPVTOC.

For more information contact Susan Georgeson at Smartech Systems, Inc., Dallas, TX, (800) 537-6278 or:

Circle #210 on the Reader Service Card

SQLizer Speeds DB2 Program Development

SQLizer is a new product from Application Development Systems, Inc. that speeds the development of DB2 programs. It offers a complete set of facilities that shorten the "experimentation phase" of DB2 program development. SQLizer accomplishes this by providing the Interactive Program Development (IPD) facility that lets the programmer dynamically control the execution of the DB2 program. Using a full-screen ISPF-like interface, IPD lets the user suspend execution, make changes to the program's SQL and then execute the program with new SQL. If the inserted SQL is unsuccessful, SQLizer automatically suspends execution at that point and the appropriate DB2 return code and error message are displayed. The programmer can change the SQL and resume execution from that point. All of this can be done without recompiling, rebinding or leaving the SQLizer session.

For more information contact Doug Anderson at Application Development Systems, Inc., Minneapolis, MN, (800) 358-3048 or:

Circle #211 on the Reader Service Card

PostScript Laser Printing For IBM Mainframes

OutPost, a new product from Trax Softworks, Inc., is named after its function as a PostScript output processor. It is a printing utility program that converts any printable file into PostScript, permitting users of IBM mainframes to generate and produce their output on PostScript printers from any manufacturer including IBM, Apple Computer and numerous others. Trax believes that OutPost will allow mainframe users, particularly PROFS shops, to take advantage of the wide variety of printers which now accept PostScript.

For more information contact Cheryl Thomas at Trax Softworks, Inc., Los Angeles, CA, (213) 475-8729 or:

Circle #212 on the Reader Service Card

MVS/Quick-Ref Offers New Features

Release 2 of MVS/Quick-Ref, Chicago-Soft's quick reference aid for ISPF users under MVS, has been enhanced with several new features. The most significant reference topics added are: CICS messages and transaction abend codes, REXX language syntax, IMS status codes, cut-and-paste for text and data and support for users to add their own reference information. MVS/Quick-Ref has reportedly more than doubled its reference database in excess of 77,000 lines of documentation making it more useful to programmers, operators and technical support people.

For more information contact Peter McLaughlin at Chicago-Soft, Ltd., Chicago, IL, (312) 525-6400 or:

Circle #213 on the Reader Service Card

Almost TSO Enables Off-Loading of TSO Functions

Designed for TSO ISPF installations, Applied Software's Almost TSO enables the off-loading of TSO functions such as Edit, Utilities, Submit and JES output to the more Almost TSO multi-user VTAM environment. It is said to provide TSO-like functions for a fraction of TSO's costs and, unlike TSO's one user address space, Almost TSO supports dozens of users and with multiple address spaces (hundreds can be supported). Almost TSO provides both FSE+ and ISPF/PDF display formats.

For more information contact Ron Turner at Applied Software Inc., North Palm Beach, FL, (407) 626-4818 or:

Circle #214 on the Reader Service Card

Schools Face Computer Science Crisis

By Pat McGettigan

Isn't this an exciting profession? Technological change and the opportunities that go with it are our constant companions. No room for boredom here! And if that isn't enough, what about the salary and benefit levels that seem to hold their own, even in hard times? Then why, at a time when demand and compensation levels are at all time highs, are so many of our young people avoiding careers in the information industry?

According to a recent study by ADAPSO, the software and services industry association, enrollment in computer science courses has been declining since 1983. Many of our schools face major cutbacks in these programs that have taken years to build.

When I began my career in 1966, IBM customers were making the traumatic migration to the System/360s. All of a sudden there were things called "disks" and you could do something called "multiprogramming." Soon there were "CRTs." And when someone put a CRT on one end of a program and a disk on the other, modern commercial computing was off to the races.

In those days, there was little or no computer science, let alone computer scientists, especially in the commercial world. Just a short time before, "programmers" (as we know them today) "wired boards" on primitive accounting machines and were little more than clerical workers.

The first programmers came from all kinds of backgrounds. Some had degrees in fields like mathematics, but the majority came in the hard way, via early training schools or on-the-job training programs. To meet the demand for programmers, companies hired raw talent and filled in the gaps with employees with other backgrounds. I was one of the latter.

With the growth of computing in both the commercial and scientific sectors, major computer science programs have been developed out of necessity. But all has not been smooth, however, as many, including myself, have been critical of the content of many of these programs. After all, there are companies called IBM and DEC!

Apparently, the popularity of computer careers reached its peak in the years after the PC boom of the early 1980s. This was made clear last October at the ADAPSO conference when a distinguished panel of educators related their experiences. They were as worried and puzzled as the rest of us. I remember one questioning whether the interest generated by the PC boom was real or just a fad.

There was also discussion about the changing tide of interest away from scientific study in general with our young people opting instead for business and legal careers. However, we can take some solace in the fact that, based upon the kids' nonscientific direction, we may not know how to make anything, but we sure will know how to sell it and how to protect it within all applicable legal bounds!

We can only guess at what the effects of this trend away from computer-related professions have been. However, one serious result has been the decline in our skilled labor force.

Pat McGettigan, President and CEO of Landmark Systems Corp.

Even before my entry into the vendor world in 1983, I was aware that the demand for computer solutions seemed to be outstripping production capacities. Today, this situation has developed to the point that people refer to the use of "4GLs" and other such products as *programming!* Maybe if we had paid as much attention to our educational institutions over the last several years as we have to the quick-fix product solutions, we might have a more promising picture today.

Fortunately, there are influential people taking an active part in finally trying to forge a link between America's commercial computing world and its academic institutions.

At the meeting last October, ADAPSO, under the leadership of IBM Vice President Bob Berland, kicked off Success 2000, a program intended to promote computer science as a career. The basic game plan calls for the various vendor companies to "adopt" schools in their areas. A speakers' bureau was formed and an array of promotional materials developed. These include videos featuring John Akers of IBM, Ken Olsen of DEC and other industry leaders.

Like many other companies, Landmark enlisted in Success 2000 and our program has gotten off to a fast start. We have formed two employee committees to coordinate our activities. One is working with local high schools and the other with local colleges. We have already formed our own speakers' bureau as well and have conducted interviews and planning sessions with high school and college faculty members. Great progress has been made and we look forward to an active fall.

"What can I do to help?" you may ask. I have a couple of suggestions.

First, you and your company can adopt schools in your area. I'm not talking money. That is another, simpler way to help. However, you are rich in talent, experience and facilities. And making these assets available will not hurt that bottom line one bit. Perhaps you will be as pleasantly surprised as I was (by a 35 percent Landmark volunteer level)! It should be a great morale booster, not to mention a source of great young talent. One word of advice though — let the faculties of your adopted schools call the shots.

Second, you can keep your hiring and training practices flexible. Certainly, continue to look for those great computer science grads. But, don't make the mistake of overlooking the raw talent that undoubtedly exists in your own company. Train them. Give them the greatest business gift you have to offer: opportunity. If someone hadn't done that for me 23 years ago, I might still be selling ladies shoes!

Why ADABAS 5 is thousands of dollars faster.

The investments your organization makes in its data base technology will either cost it a fortune—or save it one. That's why you need a DBMS that assures optimum performance in a high production environment: ADABAS 5.

In a recent series of standard, fully scaled benchmarks, audited by Coopers & Lybrand, ADABAS 5 proved again that it is thousands of dollars faster. Each benchmark was conducted on a National Advanced Systems AS/EX™100 (equivalent in power to an IBM 3090 500S).

In the **standardized TP1 debit-credit benchmark,** ADABAS 5 worked with a data base containing **1 million accounts.** The results: a **record-breaking 388 transactions per second** (tps)—99.3% serviced in under one second!

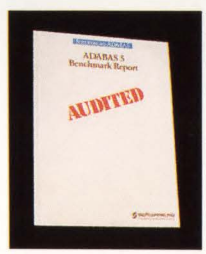
For the first time, an authentic **ET1 debit-credit benchmark** was conducted with ADABAS 5 managing **10 million accounts.** The results: **167 tps** (from terminal in/to terminal out)—99.5% serviced in under one second!

These figures represent major savings in time and money. But they're not surprising—at least not to the thousands of organizations which already use ADABAS 5.

They expect more performance. Plus the benefits that come from 18 years' experience in DBMS technology:—relational functionality which allows adaptable data structures and fast information retrieval based on multikey criteria—document management and free text retrieval—24 hour processing in a fully integrated DB/DC environment—portable applications across various hardware and operating systems—distributed processing—entity relationship data bases with recursive retrieval functions for knowledge-based systems.

Discover how much more profitable your DP organization can be. **Conduct your own ADABAS 5 benchmark,** using your own data and application profile in your own production environment. The facts will speak for themselves.

™ AS/EX is a trademark of National Advanced Systems Corporation.



Demand the performance and functionality only ADABAS 5 can offer.

To order your free copy of the ADABAS 5 Benchmark Report, call toll-free:

1-800-843-9534 today

(In Virginia or Canada, call 703-860-5050.)

SOFTWARE AG

PROGRAMMING BUSINESS SUCCESS

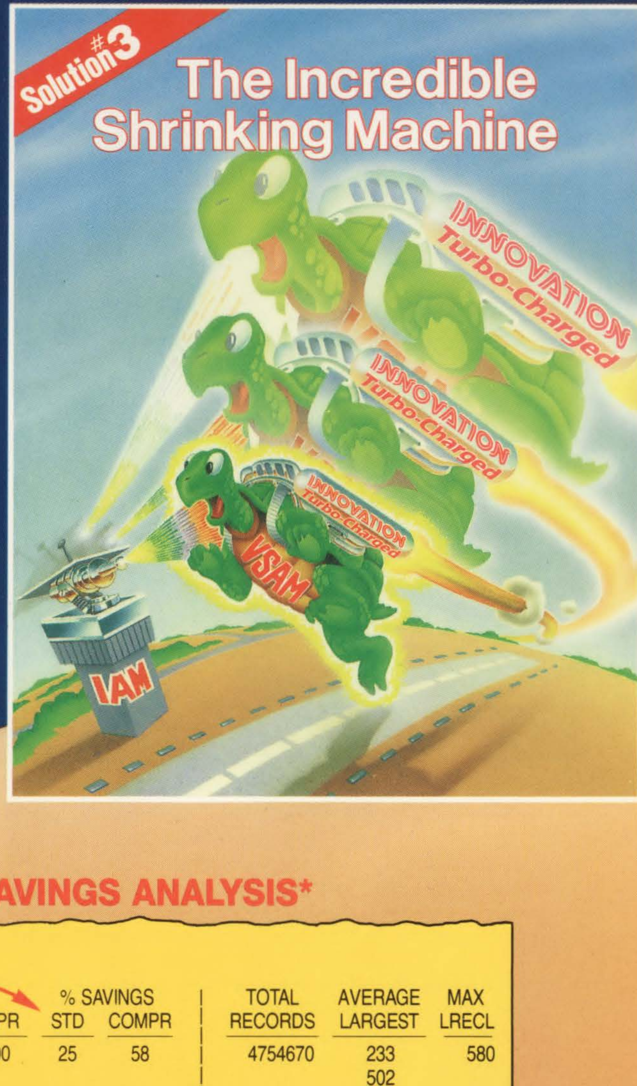
CIRCLE #16 on Reader Service Card ▲

VSAM DATA COMPRESSION

Without the CPU Overhead

IAM REDUCES THE SIZE OF YOUR VSAM FILES BY 30 TO 70%

IAM'S FILE STRUCTURE — SAVES 20 TO 40%
IAM uses an advanced file structure which is far superior to VSAM. IAM's supercompressed index, freespace concepts and blocksizes make much more efficient use of disk space.



IAM'S DATA COMPRESSION
SAVES AN ADDITIONAL 20 to 50% DASD SPACE
Most files contain records with unused fields or repeating sets of characters. When IAM applies its proprietary compression techniques, the result is an additional 20 to 50% reduction in file size.

IAM's CPU time is dramatically less than competing compression products. In fact, since IAM's CPU time is normally much less than VSAM, IAM with data compression takes less CPU time than normal VSAM processing.

TRANSPARENT
Online systems (CICS), BATCH jobs, TSO, SMP/E and other applications make extensive use of key indexed VSAM (KSDS) files.

IAM is a transparent alternative to VSAM KSDS files, which substantially reduces the impact of VSAM processing in your installation. There are no modifications to programs or JCL to use IAM files in place of VSAM.

AUTOMATIC RELEASE OF UNUSED SPACE
IAM takes the guessing game out of VSAM space allocation. Large amounts of disk space are wasted when users overestimate how much space VSAM requires or how many records a file will contain. VSAM cannot release overallocated space.

FREE VSAM SPACE SAVINGS ANALYSIS*

SPACE SAVINGS

DATA SET NAME	VSAM TRKS ALLOC	VSAM TRKS USED	IAM TRKS STD	IAM TRKS COMPR	% SAVINGS STD	% SAVINGS COMPR	TOTAL RECORDS	AVERAGE LARGEST	MAX LRECL
BIG.CLUSTER	37155	37155	27855	15600	25	58	4754670	233 502	580
CICS.FILE.MASTER. TABLE.CLUSTER	21000	19005	12720	9495	33	50	5068165	150 150	150
NAME.ADDRESS.FILE	9315	8985	6465	1875	28	79	428529	680 2090	2090

The VSAM simulation report displays the current size and data characteristics of your VSAM files and how much space IAM will save you with and without data compression.

***To see your VSAM usage, send for INNOVATION's free VSAM reporting programs.**

Call for a Free No Obligation 90 Day Trial

Supports MVS, MVS/XA and MVS/ESA

Makers of FDR and ABR

INNOVATION DATA PROCESSING
275 Paterson Avenue, Little Falls, NJ 07424 • (201) 890-7300

CIRCLE #97 on Reader Service Card ▲