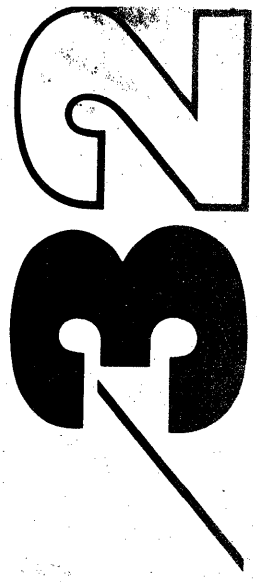
SYSTEM/32

IBM System/32
Utilities Program Product
Reference Manual
Data File Utility



IBM System/32
Data File Utility
Programming Information

SC21-7600-3
File No. S32-32

Program Number
5725-UT1

IBM System/32
Utilities Program Product
Reference Manual
Data File Utility

Fourth Edition (May 1980)

This is a major revision of and obsoletes, SC21-7600-2 and technical newsletters SN21-7935 and SN21-7997. Changes or additions to the text and illustrations are indicated by a vertical line to the left of the change or addition.

This edition applies to version 9, modification 0 of the IBM System/32 Utilities Program Product (Program 5725-UT1) and to all subsequent versions and modifications until otherwise indicated in new editions or technical newsletters.

Changes are periodically made to the information herein; changes will be reported in Technical Newsletters or in new editions of this publication.

Use this publication only for the purposes stated in the *Preface*.

Publications are not stocked at the address below. Requests for copies of IBM publications and for technical information about the system should be made to your IBM representative or to the branch office serving your locality.

This publication could contain technical inaccuracies or typographical errors. Address your comments about this publication to IBM Corporation, Publications, Department 245, Rochester, Minnesota 55901. IBM may use and distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

Purpose Of This Manual

The Data File Utility (DFU) is part of the System/32 Utilities program product, Program Number 5725-UT1. This manual is meant to be a learning tool for the new user of the Data File Utility and a reference manual for the experienced user. The new user will find it helpful to read the entire manual, paying special attention to the first seven chapters and their examples. This manual describes:

- DFU purpose and function
- DFU job setup and execution
- How to change supplied DFU format descriptions and create new ones
- How to operate DFU in each mode

Audience

The primary audience for this manual is the account programmer. The secondary audience includes third party programmers and IBM field support personnel (SEs, PSRs, PSCEs, and instructors).

How This Manual is Organized

The introductory chapter describes DFU and its capabilities, how you use DFU, and how DFU works. Chapters 2 through 7 explain DFU features, how to set up a DFU job, and how to run the DFU job.

The remaining chapters and appendixes contain additional information about DFU that you will find useful as you become more familiar with the program.

Also provided is a glossary to help you understand the terms used.

System Requirements

The data file utility portion of the IBM System/32 utilities program product runs on all models of System/32 and supports all available System/32 features.

Related System/32 Publications

The reader should have access to the following System/32 publications:

- *System/32 Operator's Guide,* GC21-7591
- *System/32 Displayed Messages Guide,* GC21-7704
- *System/32 System Control Programming Reference Manual,* GC21-7593
- *System/32 Utilities Program Product Reference Manual—Source Entry Utility,* SC21-7605
- *System/32 Utilities Program Product Reference Manual—Sort,* SC21-7633
- *System/32 RPG II Reference Manual,* SC21-7595

Contents

Chapter 1. Introduction

The data file utility (DFU) is a program product designed to process your data files. There are four basic uses of DFU:

1. Creating indexed data files.
2. Maintaining indexed data files (modifying, adding, and deleting information).
3. Displaying information from an indexed data file.
4. Preparing and printing reports from information in a direct, sequential, or indexed data file. (At your option, data can be sorted prior to printing.)

The following four examples are explained in detail later in this manual; they are briefly described here to give you an indication of how you can use DFU.

Example 1. Creating a Customer Order File

You can use DFU to create a customer order file. Your customer orders are received by a clerk who checks them for completeness, totals item quantities on an adding machine for control purposes, and clips the adding machine tape to the customer orders. Customer orders are keyed into the system and printed as they are keyed. To check that all item quantities were entered correctly, DFU can accumulate them as they are keyed and print the total at the end of the job. This total can be compared with the adding machine total. If the totals match, you have an indication that the quantities were entered correctly. If they do not match, an error was probably made when entering a quantity. The customer orders can be compared with the printed report of records entered to find where the mistake was made.

Example 2. Maintaining a File

You can use DFU to update your files. Suppose a customer calls and asks to change an order you have already recorded, to place another order, or to cancel an order. Each of these can be done easily with DFU.

Example 3. Displaying Information from a File

You can use DFU to display information from your file. Assume that you have an inventory data file. If a customer calls and places a rush order, you need to know immediately whether or not the stock is on hand to meet the request. Using DFU, item information can be displayed from the inventory file (for example, the quantity on hand) that will help you decide if the order can be filled.

Example 4. Printing Reports

Suppose that all customer payments on accounts and customer credits to accounts are keyed each day into a weekly cash receipts file. These are entered into the file as they are received. At the end of a week, DFU is used to print a report showing the cash receipts for that week. In addition, a request can be made to print the report sorted by the customer number.

Note: DFU can also be used to do the following:

1. Create, maintain, display and print a source or procedure member (explained in Chapter 11, *Source and Procedure Members).*
2. Allow you to use the sort program product to sort a data file (explained in Appendix D, *Sorting with DFU).*

HOW TO USE DFU

Using DFU requires that you:

1. Know the DFU features and your job requirements. Chapter 2, *DFU Features* explains the DFU features and when to use them.

2. Provide an RPG II *source member* (RPG II file description and input specifications) that describes the data file. RPG II programming knowledge is required to create or modify the RPG II source member. Therefore, it is assumed that it has been created for you, named, and placed in the system library. You need only supply its name in order to use it. If you need a complete description of the RPG II source member, refer to Chapter 8, *RPG II Source Member.*

3. Create a *load member* (called a format description) in the system library that describes your file and how you want DFU to process it. This step, called the setup step, is explained in Chapter 3, *DFU Setup Step*. It involves keying a command statement, then responding to a series of prompts that appear on the display screen. Your responses require an understanding of the DFU features and job requirements.

4. Do the file creating, maintaining, displaying, or printing that you set up in step 3. This is called running the DFU job. Running the job is easy once it is set up; it can be as simple as keying a command statement, then keying data as DFU requests it. Chapters 4, 5, 6, and 7 explain how you run a DFU job after it has been set up.

DFU setup sheets have been provided in Appendix F to assist you with the DFU setup.

Once DFU creates the format description in step 3, you can use it over and over again. This means that you skip steps 1 through 3 and begin with step 4.

You can create a format description for a job and then use it for another type of DFU job (for example, the same format description can be used to create and update a data file). Once you create and save the format description, it is interchangeable among the various DFU job types.

HOW DFU WORKS

Figure 1 shows how DFU works.

YOU DO THIS

Provide an RPG II *source member* that describes the data file (see note 1). Key a job setup command statement (ENTER, UPDATE, INQUIRY, or LIST), then responding to DFU prompts for additional command statement information. Chapter 3 and Appendix A explain the command statements.

DFU DOES THIS

Converts the RPG II source member into DFU attributes. DFU attributes are a simplified version of the RPG II source member (see note 2).

YOU DO THIS

Respond to a series of prompts to create DFU specifications. Your responses indicate how you want the file to be processed. Chapter 3 explains the series of prompts.

DFU DOES THIS

Creates DFU specifications (see note 3).

YOU DO THIS

Indicate the end of the setup step by pressing the EOJ command key when the following prompt appears:

HIT EOJ CMD KEY TO CONTINUE PROCESSING, OR YOU MAY NOW UPDATE DFU SPECS

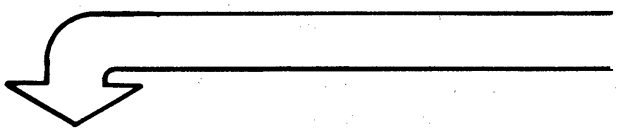
DFU DOES THIS

Builds a format description. The format description is the end result of the setup step. It describes your file and exactly how you want DFU to process it. DFU creates the format description by combining the information from the DFU attributes and DFU specifications. The format description is stored in the library as a *load member* (a library member that is in executable form) with the name you specified in the command statement.

You are now ready to create, maintain, display, or print a data file. If you are printing a data file, the report is printed automatically after the format description is built. No action is required of you to do this.

To Part 2 of Figure 1

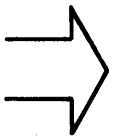
Figure 1 (Part 1 of 2). How DFU Works



YOU DO THIS

Run the job you have set up by doing one of the following:

- Key records field by field into the data file. (See *Steps for Entering a Data File* in Chapter 4.)
- Change, add, or delete records. (See *Steps for Maintaining a Data File* in Chapter 5.)
- Display records from the file. (See *Displaying the Records* in Chapter 6.)



DFU DOES THIS

Performs the requested function. Note that the format description created in the setup step for this job can be used in all subsequent runs of this job and similar jobs.

Notes:

1. The *RPG II source member* consists of RPG II file description and input specifications. It indicates the:

   Length of each record in the file
   Location and length of the record key in each record
   Identification of the various record types in the file
   Location, length, and description of the fields within each record type.

   (Record keys and record types are explained in Chapter 2, *DFU Features.*)

   When using DFU to print a report from information in two related data files, you must provide an RPG II source member for each file.

   Refer to *Related Master File* in Chapter 2 for information about printing reports using two data files.

2. *DFU attributes* are information about your file that DFU builds from an RPG II source member. The attributes are shown on the display screen as you respond to the DFU prompting sequence. For a complete description of the DFU attributes, refer to Chapter 9, *DFU Attributes.*

   If you provide two RPG II source members, DFU converts the source member you name in the command statement to DFU attributes, then prompts for the name of the second source member. DFU then converts this source member to DFU attributes.

3. *DFU specifications* are information about your file that DFU builds from the responses you made to the series of DFU prompts. They describe the file processing that you want done in the job run. These specifications are shown on the display screen when you finish responding to the prompts. They can be modified if errors have been made when responding to the prompts. If you need a detailed explanation of the DFU specifications, refer to Chapter 10, *DFU Specifications.*

Figure 1 (Part 2 of 2). How DFU Works

In order to use DFU effectively, you should know what DFU features are available for each of the DFU jobs before you begin the job setup step. Figure 2 shows the DFU features that are available for each job type. Each of these features is explained in this chapter.

	When Used			
Feature	Creating a Data File (ENTER)	Maintaining a Data File (UPDATE)	Displaying a Data File (INQUIRY)	Printing a Report from Data File Information (LIST)
Record keys	X	X	X	X
Automatic key generation	X	X		X
Record type	X	X	X	X
Editing fields	X	X	X	X
Report title	X	X	X	X
Spacing between fields	X	X	X	X
Deleting records		X		
Field names and headings	X	X	X	X
Automatic field duplication	X	X		
Field accumulation	X	X		X
Self-check fields	X	X		
Record count at end of job	X	X		X
Printing records	X	X	X	X
Sort fields				X
Control fields and accumulator fields				X
Record selection				X
Result fields				X
Related master file				X
Record list				X
Detailed summary list				X
Nondetailed summary list				X

Figure 2. DFU Feature

RECORD KEYS

A record key is a field within a record that identifies the record from all others in the file. The following are true of a record key:

- It can be a meaningful field in the record (for example, an employee number in a payroll record).
- It can be a field containing a sequential number assigned by either you or DFU.
- Its value can be either positive or negative.
- Its length and position in the record are specified in the RPG II source member. The key can be at most 29 positions long.

Keys must be assigned when the file is created and when records are added to a file. DFU requires keys to select the record when creating, maintaining, or displaying a data file. For example, to display a record, its key must be entered before DFU can retrieve and display the record.

Record keys are optional in a data file to be listed. When listing a data file that has no record keys, you can have the record numbers generated by responding yes to the following prompt that appears in the setup step:

SHOULD RECORD KEYS BE PRINTED?

The record numbers generated are not relative record numbers, but correspond to the order in which the records are printed. If no sort or select options are taken, every record in the file will be numbered in the listing.

When you respond to the prompts to set up a job to create a data file, the prompt

SHOULD DFU GENERATE KEYS?

appears if the key is five positions long. (DFU automatically determines that you will provide keys if the key length is not five positions long.) If you respond yes, DFU will generate keys for you, beginning with 00010 for the first record and incrementing by 00010 for each successive record. If you respond no, you will have to supply the key for each record entered.

The same prompt appears if you are setting up a job to maintain a file (again, only if the key is five positions long). If you add records to the end of a file, DFU will generate keys for you, the same as for entering a data file.

Note: You will be able to suspend automatic key generation to do other file maintenance (deleting or updating existing records, or inserting new records between existing ones), then return to automatic key generation.

RECORD TYPE

Files can contain one or more types of records. Records have the same type if they have the same data in the same locations in the records. For example, a file containing customer order information can have two record types; one that has name and address information, and one that has item information. These record types can have one or more common fields but, in general, they have different data in the records. DFU must know which record type it is processing in order to obtain information from the record. In order to do this, each record type is assigned a record identification indicator—two digits that DFU obtains from the RPG II source member.

When you are setting up any of the DFU jobs, the prompt

xx—ANY FIELDS FROM THIS RECORD TYPE?

will appear. The xx is the record identification indicator. If you respond yes, you are prompted for the names of the fields you want processed in that record type.

The record identification indicator is displayed when you are running the job. You will be able to change it so that you can process the record type you want.

EDITING FIELDS

DFU performs editing automatically on numeric fields. This occurs when a numeric field is printed and also when it is shown on the display screen. Editing numeric fields with decimal points and negative signs makes them easy to read. DFU does the following editing on numeric fields:

1. It inserts a decimal point in the proper place (if there are any decimal positions indicated in the RPG II source member).
2. It indicates negative numeric fields by placing a minus sign (-) to the right of the last digit in the field.

In addition, DFU blanks out unnecessary leading zeros, resulting in an uncluttered report.

For example: 00149 prints as 149 and
000.12 prints as 0.12

REPORT TITLE

During the setup step you will be able to key a job title as the response to an

ENTER TITLE

prompt. A maximum 24-character title can be keyed.

When creating, maintaining, or displaying a file, the title appears centered on the first line of the display screen. When records are printed during one of the preceding job runs or when listing a data file, the title is printed on the first line of each new page along with the date and page number.

SPACING BETWEEN FIELDS

During the setup step for creating, maintaining, displaying, and listing data files, you are able to indicate the number of spaces between fields both on the display screen and on the printed output. This allows you to separate fields and make them easier to read. The prompt that appears is:

ENTER COL SPACING VALUE (0-9,DEFAULT=1)

The spacing occurs between the longest parts of fields, either the data portion or the field headings. For example, if the first field is longer than the field heading and the second field is shorter than the field heading, spacing will occur between the end of the first field and the beginning of the second field's heading.

When determining column spacing all numeric fields are automatically made one position longer to accommodate the possible inclusion of a minus sign. This position will appear as an additional space to the right of the field. Also, when doing a summary list of a data file, accumulated fields are automatically made two positions longer (in addition to the position for the minus sign) to allow subtotals and totals to print under the column headings. If your summary list includes detail lines, two additional spaces will appear to the left of each accumulated field.

Note: The sum of the lengths of the headings and the spaces between the headings cannot exceed 396, the maximum length of three printed lines. DFU can process only those fields that can be printed on three lines. Therefore, too much space between fields can limit the number of fields that DFU can process.

DELETING RECORDS

The delete record feature allows you to mark a record for deletion while you are updating a data file. In the prompting sequence of the setup step, an

ENTER VALUES FOR 'DELETE CODE,POSITION'

prompt appears. You key the delete character and a record position for the delete character. For example, a response of X,1 indicates that the character X is to be placed in the first position of any record marked for deletion. Then when updating records, the DELETE command key can be pressed to mark a displayed record for deletion. DFU places the delete character into the position you have specified. The record is printed, followed by a RECORD DELETED message. For example:

```
          ┌─ delete character
          ↘
00020   X  XC4312 0313 0200 134311

RECORD DELETED
```

The printed list of records marked for deletion can be kept as a record of the deletions you have indicated in the file.

Note: The record still exists in the file (it is only marked for deletion by the delete character); however, it can be ignored by programs processing the data file by specifying in the program that a record with a delete character is no longer active. If you later decide that the record should not be deleted, remove the delete character from the record by using the DFU UPDATE command.

You can specify any record position for the delete code, but if the record identification code position is used, DFU displays the following message when you try to display a record that has been marked for deletion:

```
DFU 0007
UNDETERMINED RECORD TYPE
```

To keep from filling a data file with records marked for deletion, use the ORGANIZE command statement to remove the marked records. Part 2, Procedures, of the *IBM System/32 System Control Programming Reference Manual,* GC21-7593, explains the ORGANIZE command statement.

FIELD NAMES AND HEADINGS

When setting up a DFU job to create, maintain, display, or list a data file, you will be prompted for field names and headings for the fields to be processed.

The
ENTER FIELD NAME
prompt requires a response of a field name that is used in the RPG II source member. This name is restricted to six characters and is usually in abbreviated form (for example, ORDNO for customer order number). The
ENTER COLUMN HEADING
prompt appears after a field name is keyed and allows a maximum 16-character heading to be keyed for the field. Then when the job is run and the field is displayed or printed, the descriptive heading will appear instead of the field name.

These two prompts reappear, allowing you to name and assign headings to all of the fields you want DFU to process.

Note: When creating, maintaining, or displaying a data file, a maximum of 40 fields (including the key field) and associated headings are allowed for each record DFU processes. When listing a data file:

1. For a summary list, a maximum 40 fields and associated headings are allowed.

2. For a record type list, a maximum 40 fields and associated headings are allowed for each record type.

In addition, since DFU processes only those fields that can be printed on three lines (396 print positions), the length of the fields can limit the number of fields that DFU can process.

The prompt
ENTER COLUMN HEADING FOR KEYS
appears once so that you can assign a descriptive name for the record key field.

Do not enter the key field name as a response to a field name prompt. However, the key field can be used for control field, sort field, and record selection under the LIST function when redefined in part or in its entirety.

Note: DFU will use the appropriate field name from the RPG II source member as the heading (and *KEY as the heading of the record key field) if you do not enter a heading.

AUTOMATIC FIELD DUPLICATION

The auto dup (automatic field duplication) feature speeds data entry. It allows one or more fields (for example, the date) to be automatically duplicated from one record to the next as the records are keyed. You specify this feature during the prompting sequence of the setup step for either creating or maintaining a data file. By responding to the
IS THIS AN AUTO DUP FIELD?
prompt, you can define a field as an auto dup field. This field will then be automatically copied from one record to the next as the records are keyed.

For example, in a customer order file the customer number and company order number have the same value for each record entered from an individual order form. These fields could be defined as auto dup fields.

DFU has an auto dup indicator that is turned on and off by the AUTO DUP command key. In order to use the automatic duplication feature:

1. The auto dup indicator must be on.

2. The fields must be defined as auto dup fields.

3. At least one record must have been processed.

FIELD ACCUMULATION

The field accumulation feature enables from one to ten fields to be totaled as DFU processes them. Field accumulation is specified during the setup step for creating, maintaining or listing a data file.

Field Accumulation When Creating a Data File

In the customer order file example in Chapter 1, the quantity field (QTY) was summed by DFU, then compared with adding machine totals. This provided a means of checking that the quantities had been keyed correctly. QTY was defined as an accumulator field in the job setup step by responding yes to the prompt:

ACCUMULATE THIS FIELD?

From one to ten fields can be defined as accumulator fields. When the job is run, two totals are kept for each of the fields—a subtotal (batch total) and a final total.

Example 1 in Chapter 3 describes how the field accumulation feature is defined when setting up the job. The example in Chapter 4 describes how batch totals and final totals are kept when the job is run.

The batch totals are updated after each record is keyed. They can be printed using the PRINT ACCUM command key. After the batch totals are printed they are added to the final totals and then reset to zero. Both the batch and final totals are printed automatically at the end of the job.

Batch totals and final totals have a maximum 15 digits. If a batch total exceeds 15 digits during the job run, a message is printed indicating the overflow along with the batch total value before the overflow occurred. Then the batch total is set to the value of the field that caused the overflow, and the job continues.

If a final total exceeds 15 digits during the job run, a message is printed indicating the overflow along with the final total value before the overflow. The final total is reset to the value of the batch total that caused the overflow, and the job continues.

A message is printed at the end of the job (***WARNING ***-OVERFLOW IN ACCUMULATORS) if a batch or final total overflow has occurred.

Field Accumulation When Maintaining a Data File

Accumulator fields for maintaining a data file are defined in the setup step by responding yes to the prompt:

ACCUMULATE THIS FIELD?

From one to ten fields can be defined. These should be the same accumulator fields defined when the file was created (usually, the same format description used to create the file will be used to maintain the file).

When the job is run, a modified, added, or deleted accumulator field causes DFU to update the batch and final totals for the field accordingly.

Note: When a record is marked for deletion, the values of its accumulator fields are subtracted from the batch and final totals. If the record is later included in the file (by removing the delete code), the accumulator field(s) are automatically added to the batch and final totals.

Batch totals and final totals have a maximum of 15 digits. If a batch total exceeds 15 digits during the job run step, a message is printed indicating the overflow along with the batch total value before the overflow occurred. The batch total is set to the value of the field that caused the overflow, and the job continues.

If a final total exceeds 15 digits during the job run step, a message is printed indicating the overflow along with the final total value before the overflow. The final total is reset to the value of the batch total that caused the overflow, and the job continues.

A message is printed at the end of the job (***WARINING ***–OVERFLOW IN ACCUMULATORS) if a batch or final total overflow has occurred.

Example of Maintaining a Data File in Chapter 5 shows how accumulator fields are updated.

Field Accumulation When Listing a Data File

Accumulator fields to be used when listing a data file are defined in the setup step by responding yes to the prompt:

ACCUMULATE THIS FIELD?

From one to ten accumulator fields can be defined.

Accumulator fields are used in conjunction with control fields. Control fields are another type of field you define in the setup step.

Note: Control fields, accumulator fields, and how they are used in conjunction with one another are explained in *Control Fields and Accumulator Fields* in this chapter.

For a record type list, accumulator field hold areas are 15 characters long. For a summary type list, accumulator field hold areas are two characters longer than the final being accumulated (15 characters maximum).

Anytime an accumulator overflows (the accumulated sum becomes too large for DFU), a message is printed indicating that he next line printed contains an accumulator field that overflowed.

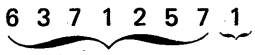
The accumulator value before the overflow occurred is also printed, then the accumulator is set to the value of the field that caused the overflow. A message is also printed at the end of the job (***WARNING***–OVERFLOW IN ACCUMULATORS).

The examples in *Types of Lists* in this chapter show the use of accumulator fields when listing a data file.

SELF-CHECK FIELDS

A practice that provides some protection against clerical and keying errors is that of using self-check fields. DFU provides a method of verifying a self-check field at the same time it is keyed.

A self-check field is composed of two parts: the base number (for example, your control or account number) and one check digit. The base number can be up to 39 digits long. The check digit is the rightmost digit in the field. The following is an example of an 8-digit self-check field:



Base Number Check Digit

When you establish your control or account numbers (the fields you want verified) you must calculate the check digit for each field and include it as the last digit of the number.

The check digit is calculated by your use of a formula (a fixed sequence of operations on the base number). When the field is keyed, DFU uses the same formula that you used to calculate the check digit. If the check digit DFU produces does not match the check digit you supplied, an error message is displayed and the field can be rekeyed.

There are two formulas for obtaining the self-check digit, called modulus 10 and modulus 11. Modulus 10 is used to detect either incorrect keying of a single digit or a single transposition. Modulus 11 is used to detect incorrect keying of a single digit, single transpositions, and double transpositions. These formulas are described in Appendix E.

You specify the self-check feature and the formula DFU should use (this must be the same formula originally used to calculate the check digit) during the prompting sequence of the setup step by responding to the prompts:

IS THIS A SELF-CHECK FIELD?

and

ENTER SELF-CHECK NUMBER (10 or 11)

PRINTING RECORDS

Printing records is a feature of DFU used when creating, maintaining, displaying, and listing files.

The record printing done when listing a file is defined during the setup step. This is explained in detail in the *Types of Lists* information in this chapter.

The record printing done when displaying a file can be requested during the job run; there is nothing to be specified in the setup step for this feature.

When a file is maintained, updated records and records marked for deletion are automatically printed. There is nothing to be specified in the setup step for this feature. Two lines are printed for each updated record: the first line shows the unchanged record and the second line shows only the updated fields. Two lines are also printed for each record marked for deletion: the first line shows the record and the second line shows the message RECORD DELETED. The sample file maintenance run in Chapter 5, *Maintaining a Data File,* shows how these records are printed.

When creating or maintaining a data file, records can be printed when they are keyed. This feature simplifies finding errors that have been made when entering data. During the prompting sequence in the setup step, you specify that records should be printed when they are keyed by responding yes to the prompt:

SHOULD RECORDS BE PRINTED?

DFU will also print headings when:

- The record to be printed is the first of a record type,
- The record to be printed is the first of a new page, or
- The record to be printed is the first after a batch total has been printed.

Note: When any DFU print operations are performed on a matrix printer, the forms mode selector switch must be set in the continuous forms mode or unpredictable results may occur.

RECORD COUNT AT END OF JOB

DFU gives you a record count at end of job. The format of the information varies depending on the mode of processing being used. When listing a file, DFU prints the number of records processed as the last line of the listing. When DFU is creating or updating a file, the number of records entered, updated, and deleted is displayed on the CRT screen when the EOJ command key is pressed. In addition, the record count is printed as the last line of the listing if printing has previously occurred.

SORT FIELDS

Sort fields are a feature of DFU used when listing a data file. They are fields upon which a data file is sorted when it is listed, resulting in a well-organized, easily read report.

The following are true of sort fields:

- They are defined in the setup step by responding yes to the prompt
  SHOULD FILE BE SORTED?
  The prompts
  ENTER SORT FIELD NAME
  and
  IS THIS SORT FIELD ASCENDING?
  then appear.
- One, two, three, four, or five sort fields can be defined. Each is specified to be sorted in ascending or decending order.
- If you define a sort control field, it must be in each of the records included in the list.
- The order in which you specify the sort fields determines the order of the final output.

For example, NUMBER and DEPARTMENT are the two sort fields defined. The following shows the result of defining NUMBER first, then DEPARTMENT (in this example, both are sorted in ascending order):

NUMBER	DEPARTMENT
1234	17
1357	16
1368	18
1426	18
1980	15
1999	18

The following show how the result changes when DEPARTMENT is defined first, followed by NUMBER:

DEPARTMENT	NUMBER
15	1980
16	1357
17	1234
18	1368
18	1426
18	1999

Note: DFU sort produces an addrout sort. Three bytes per record are required. The DFU sort routine allocates 10 blocks of space. If you get sort operator message 7732, the disk space assigned to the sort output file in the // FILE NAME-OUTPUT statement on the DFU procedure #DFUSRT is too small. Increase the size of the BLOCKS parameter on the FILE statement.

CONTROL FIELDS AND ACCUMULATOR FIELDS

Control fields are a feature of DFU used when listing a data file. They are used to control the totaling and printing of accumulator fields that you have defined (the *Field Accumulation When Listing a Data File* information in this chapter explains accumulator field definition).

One, two, three, four, or five control fields can be defined in the setup step by responding yes to the prompt:

ANY CONTROL FIELDS?

then naming the control field(s) in the

ENTER CONTROL FIELD NAME

prompt(s) that appear. How the control fields are used when the data file is listed depends upon the number of control fields that you define, and the order in which you define them.

One Control Field

If one control field is defined for a list, DFU keeps two separate sums for each accumulator field when the data file is being listed: a subtotal and a final total. When the control field changes value, the subtotals are printed, added to the final totals, then reset to zero. A single asterisk is printed to the right of the last (rightmost) subtotal. DFU resumes summing until the next change in the control field occurs.

The final total for each field is printed at the end of the report. Two asterisks (**) are printed to the right of the last total. The examples in *Types of Lists* in this chapter show the use of one control field.

Two Control Fields

If two control fields are defined for a list, the first one defined is called the major control field; the second one defined is called the minor control field. DFU keeps three separate sums for each accumulator field as the data file is listed. Two subtotals and a final total are kept for each accumulator field.

When the minor control field changes its value, its subtotals are printed, added to the major subtotals, then reset to zero. A single asterisk is printed to the right of the last subtotal.

When the major control field changes its value, the following occur:

1. The subtotals for the minor control field are printed, added to the subtotals for the major control field, then reset to zero. A single asterisk is printed to the right of the last subtotal on this line.

2. The subtotals for the major control field are printed, added to the final total, then reset to zero. Two asterisks are printed to the right of the last subtotal on this line.

The final total for each field is printed at the end of the report. Three asterisks (***) are printed to the right of the last total.

Three Control Fields

If three control fields are defined for a list, the first one defined is called the major control field, the second one defined is called the intermediate control field, and the third one defined is called the minor control field. DFU keeps four separate sums as the data file is listed. Three subtotals and a final total are kept for each accumulator field.

When the minor control field changes its value, its subtotals are printed, added to the intermediate subtotals, then reset to zero. A single asterisk is printed to the right of the last subtotal.

When the intermediate control field changes its value, the following occur:

1. The subtotals for the minor control field are printed, added to the subtotals for the intermediate control field, then reset to zero. A single asterisk is printed to the right of the last subtotal on this line.

2. The subtotals for the intermediate control field are printed, added to the subtotals for the major control field, then reset to zero. Two asterisks are printed to the right of the last subtotal on this line.

When the major control field changes its value, the following occur:

1. The subtotals for the minor control field are printed, added to the subtotals for the intermediate control field, then reset to zero. A single asterisk is printed to the right of the last subtotal on this line.

2. The subtotals for the intermediate control field are printed, added to the subtotals for the major control field, then reset to zero. Two asterisks are printed to the right of the last subtotal on this line.

3. The subtotals for the major control field are printed, added to the final totals, then reset to zero. Three asterisks are printed to the right of the last subtotal on this line.

The final total for each accumulator field is printed at the end of the report. Four asterisks (****) are printed to the right of the last total.

Four Control Fields

If four control fields are defined for a list, DFU keeps five separate sums as the data file is listed. Four subtotals and a final total are kept for each accumulator field.

When the fourth control field that you defined changes its value, its subtotals are printed, added to the third control field subtotals, then reset to zero. A single asterisk is printed to the right of the last subtotal.

When the third control field that you defined changes its value, the following occur:

1. The subtotals for the fourth control field are printed, added to the subtotals for the third control field, then reset to zero. A single asterisk is printed to the right of the last subtotal line.

2. The subtotals for the third control field are printed, added to the subtotals for the second control field, then reset to zero. Two asterisks are printed to the right of the last subtotal on this line.

When the second control field that you defined changes its value, the following occur:

1. The subtotals for the fourth control field are printed, added to the subtotals for the third control field, then reset to zero. A single asterisk is printed to the right of the last subtotal line.

2. The subtotals for the third control field are printed, added to the subtotals for the second control field, then reset to zero. Two asterisks are printed to the right of the last subtotal on this line.

3. The subtotals for the second control field are printed, added to the subtotals for the first control field, then reset to zero. Three asterisks are printed to the right of the last subtotal on this line.

When the first control field that you defined changes its value, the following occur:

1. The subtotals for the fourth control field are printed, added to the subtotals for the third control field, then reset to zero. A single asterisk is printed to the right of the last subtotal line.

2. The subtotals for the third control field are printed, added to the subtotals for the second control field, then reset to zero. Two asterisks are printed to the right of the last subtotal on this line.

3. The subtotals for the second control field are printed, added to the subtotals for the first control field, then reset to zero. Three asterisks are printed to the right of the last subtotal on this line.

4. The subtotals for the first control field are printed, added to the final totals, then reset to zero. Four asterisks are printed to the right of the last subtotal on this line.

The final total for each accumulator field is printed at the end of the report. Five asterisks are printed to the right of the last total.

Five Control Fields

If five control fields are defined for a list, DFU keeps six separate sums as the data file is listed. Five subtotals and a final total are kept for each accumulator field.

When the fifth control field that you defined changes its value, its subtotals are printed, added to the fourth control field subtotals, then reset to zero. A single asterisk is printed to the right of the last subtotal.

When the fourth control field that you defined changes its value, the following occur:

1. The subtotals for the fifth control field are printed, added to the subtotals for the fourth control field, then reset to zero. A single asterisk is printed to the right of the last subtotal line.

2. The subtotals for the fourth control field are printed, added to the subtotals for the third control field, then reset to zero. Two asterisks are printed to the right of the last subtotal on this line.

When the third control field that you defined changes its value, the following occur:

1. The subtotals for the fifth control field are printed, added to the subtotals for the fourth control field, then reset to zero. A single asterisk is printed to the right of the last subtotal line.

2. The subtotals for the fourth control field are printed, added to the subtotals for the third control field, then reset to zero. Two asterisks are printed to the right of the last subtotal on this line.

3. The subtotals for the third control field are printed, added to the subtotals for the second control field, then reset to zero. Three asterisks are printed to the right of the last subtotal on this line.

When the second control field that you defined changes its value, the following occur:

1. The subtotals for the fifth control field are printed, added to the subtotals for the fourth control field, then reset to zero. A single asterisk is printed to the right of the last subtotal line.

2. The subtotals for the fourth control field are printed, added to the subtotals for the third control field, then reset to zero. Two asterisks are printed to the right of the last subtotal on this line.

3. The subtotals for the third control field are printed, added to the subtotals for the second control field, then reset to zero. Three asterisks are printed to the right of the last subtotal on this line.

4. The subtotals for the second control field are printed, added to the subtotals for the first control field, then reset to zero. Four asterisks are printed to the right of the last subtotal on this line.

When the first control field that you defined changes its value, the following occur:

1. The subtotals for the fifth control field are printed, added to the subtotals for the fourth control field, then reset to zero. A single asterisk is printed to the right of the last subtotal line.

2. The subtotals for the fourth control field are printed, added to the subtotals for the third control field, then reset to zero. Two asterisks are printed to the right of the last subtotal on this line.

3. The subtotals for the third control field are printed, added to the subtotals for the second control field, then reset to zero. Three asterisks are printed to the right of the last subtotal on this line.

4. The subtotals for the second control field are printed, added to the subtotals for the first control field, then reset to zero. Four asterisks are printed to the right of the last subtotal on this line.

5. The subtotals for the first control field are printed, added to the final totals, then reset to zero. Five asterisks are printed to the right of the last subtotal on this line.

The final total for each accumulator field is printed at the end of the report. Six asterisks are printed to the right of the last total.

RESULT FIELDS

Result fields are a feature of DFU that you can use when printing reports from data file information. A result field is a field that DFU calculates (when the report is printed) according to operations you specify in the job setup step. For example, if a customer order file contains the quantity of an item ordered in addition to the price per item, you could define a result field to be calculated as follows:

Item quantity x item price = sales amount

DFU calculates the sales amount for each transaction and prints it in the report.

The following are true of result fields:

- You can specify up to six result fields in a job setup step.
- You can specify a name under which the result will be saved for use in future calculations.
- Each result field can be a combination of up to four constants or fields from either the list file or a related master file.
- If you specify a constant, the constant:
  1. Must be 15 positions or less.
  2. Can have at most 9 decimal positions.
  3. Must be numeric.
  4. Can have one decimal point.
  5. Can be negative. If so, the minus sign must follow the last digit.
  6. Cannot have embedded blanks.
- You indicate the addition, subtraction, division, and multiplication of the fields or constants that DFU must do to calculate the result field.
- You can specify that DFU accumulates a result field. If so, the result field is treated as an accumulator field as explained in *Control Fields and Accumulator Fields* in this chapter.

When you are calculating a result field using a division operation and the divisor (factor 2) is zero, the result field is filled with slashes (/). If the result is accumulated, a zero is added to the accumulator.

In the job setup step, you must respond to the following prompts to specify a result field:

Prompt	Response
ENTER FIELD NAME	Press the RESULT FIELD command key when this prompt appears. This causes the prompts in which you can specify how DFU calculates the result field to appear.
ENTER RESULT NAME (IF DESIRED)	Key the name under which the result will be saved if you want to use it in future calculations. The name cannot be the name of a field being listed.
ENTER LENGTH OF RESULT FIELD (1-15)	Key the length (including decimal positions) of the result field. The maximum length is 15 positions.
ENTER DEC POS IN RESULT FIELD (0-9)	Key the number of decimal positions in the result field. The maximum number of decimal positions is nine.
ENTER COLUMN HEADING	Key the heading you want DFU to print above the result field column on the report.
ACCUMULATE THIS FIELD?	Press the YES command key if the result field is an accumulator field. Press the NO command key if the result field is not an accumulator field.
ENTER FIRST FACTOR IN RESULT FIELD	Key the first factor DFU uses in computing the result field. This can be a constant or the name of a field. DFU automatically places this factor in the result field with an add operation.
ENTER NEXT FACTOR IN RESULT FIELD	Key the next factor DFU should use in computing the result field. This can be a constant or the name of a field. When you make a null entry by pressing the ENTER key without keying a factor, DFU stops prompting for result field factors.
ENTER OPERATION (ADD, SUB, MULT, DIV)	Key the operation DFU must perform on the result field and the factor you named in the previous prompt. DFU performs the operation as follows: ADD – DFU adds the factor that you named in the previous prompt to the result field and places the answer in the sum in the result field. SUB – DFU subtracts the factor that you named in the previous prompt from the result field and places the difference in the result field. MULT – DFU multiplies the factor that you named in the previous prompt and the result field, then places the product in the result field. DIV – DFU divides the result field with the factor that you named in the previous prompt and places the result in the result field.

For example, assume you want DFU to generate and accumulate a result field named SLSAM (sales amount) that is calculated as follows:

QTY x PRICE - DSCNT = SLSAMT

(quantity) (item price) (discount) (sales amount)

The prompts and responses that define the result field are:

Prompt	Response
ENTER FIELD NAME	RESULT FIELD command key
ENTER RESULT NAME (IF DESIRED)	ENTER key
ENTER LENGTH OF RESULT FIELD (1-15)	7
ENTER DEC POS IN RESULT FIELD	2
ENTER COLUMN HEADING	SALES AMOUNT
ACCUMULATE THIS FIELD?	YES command key
ENTER FIRST FACTOR IN RESULT FIELD	QTY
ENTER NEXT FACTOR IN RESULT FIELD	PRICE
ENTER OPERATION (ADD, SUB,MULT,DIV)	MULT
ENTER NEXT FACTOR IN RESULT FIELD	DSCNT
ENTER OPERATION (ADD, SUB,MULT,DIV)	SUB
ENTER NEXT FACTOR IN RESULT FIELD	ENTER key

Example 2 in Chapter 3 shows an additional example that describes how you respond to prompts to specify a result field.

RECORD SELECTION

The DFU feature of record selection is available when printing reports from data file information. Record selection can be based on the following comparisons:

- Field to field (if field attributes are the same)
- Field to constant
- A combination of field to field and field to constant

If you specify an alphameric field to constant comparison, the constant must be 20 positions or less. If you specify a numeric field to constant comparison, the constant:

- Must be 15 positions or less.
- Can have at most 9 decimal positions.
- Can have one decimal point.
- Can be negative. The minus sign must follow the last digit.
- Cannot have embedded blanks.

You can specify up to 10 comparisons in either an *or* or *and* relationship with one another when you set up the job to list the file.

For example, if the customer balance is a field in the record, you could have DFU make the following comparisons to include all customers with balances from $50 to $100 and all customers with balances above $250:

Include the record if the balance is:

Greater than or equal to $50
AND
Less than or equal to $100
OR
Greater than $250

In the job setup step, you must respond to the following prompts to specify record selection:

Prompt	Response
SELECT RECORDS BASED ON FIELD VALUES?	Press the YES command key to continue with the selection criteria prompts.
ENTER NAME OF SELECT FIELD (FACTOR 1)	The field you name in response to this prompt is the field you want tested to determine if DFU should print the record in the report. Notice that the field you name first is called the select field and is also referred to as factor 1. *Note:* You can specify up to 10 select fields in a single job setup step.
ENTER CONDITION (EQ, NE, GT, LT, GE, LE)	Your response indicates the type of condition you want to be met. The following indicates your choices for a response (factor 1 is field you named in the previous prompt; factor 2 is the field or constant that you want compared with factor 1): EQ Factor 1 is equal to factor 2. NE Factor 1 is not equal to factor 2. GT Factor 1 is greater than factor 2. LT Factor 1 is less than factor 2. GE Factor 1 is greater than or equal to factor 2. LE Factor 1 is less than or equal to factor 2.
IS FACTOR 2 A CONSTANT?	Press the YES command key if factor 2 is a constant. Press the NO command key if factor 2 is a field (the next prompt is skipped).
ENTER CONSTANT (FACTOR 2 VALUE)	Your response is the constant that DFU compares with the select field. The constant must be from 1 to 20 characters.
ENTER FIELD NAME (FACTOR 2 VALUE)	Your response names the field DFU compares with the select field.
ENTER NEXT SELECT FIELD RELATION (OR,AND)	Your response indicates how the next record selection criteria relates to the one you have just defined. If both conditions must be met in order to select the record, use AND. If either condition being met satisfies the criteria, use OR. The prompt ENTER NAME OF SELECT FIELD (FACTOR 1) reappears so that you can specify another selection criteria. If you have defined all of the selection criteria, press the ENTER key.

Example 2 in Chapter 3 shows how you respond to prompts to specify record selection.

RELATED MASTER FILE

The related master file is a file of relatively permanent information (for example, an inventory file or a customer name and address file) that DFU uses in combination with a list file (a customer order file for example) to produce a printed report. The master file is an indexed file. The list file relates to the master file by containing a field that is the same as the key field in the master file. When DFU prints the report, information is available from both files.

In order to prepare a job to use both a list file and a related master file, you do the following:

- Provide two RPG II source members; one that describes the list file, and one that describes the master file. Chapter 8 provides information about the RPG II source members.
- Do the job setup step as described in Figure 8 of Chapter 3.

When you respond to the prompts in the job setup step, you can specify fields from either the list file or the master file to be the following:

- Fields to be printed on the report (Refer to *Field Names and Headings* in this chapter for further information.)
- Accumulator fields (Refer to *Field Accumulation when Listing a Data File* in this chapter for further information.)
- Control fields (Refer to *Control Fields and Accumulator Fields* in this chapter for further information.)
- Factors used to calculate a result field (Refer to *Result Fields* in this chapter for further information.)

Master file fields cannot be specified as sort or select fields.

If you specify a field name that exists in both the list file and the master file, DFU selects the field from the list file.

Example 2 in Chapter 3 shows an example of preparing a job that uses a list file and a related master file.

TYPES OF LISTS

There are three types of lists that you can define in the set-up step:

- Record
- Detailed summary
- Nondetailed summary

Record List

In the prompting sequence for the list setup step, the following prompt appears:

IS THIS A SUMMARY LIST?

If you respond no, you indicate that a record list will be printed.

A record list contains all of the record types and fields that you specify in the setup prompting sequence. Headings are printed each time a record type changes and at the top of each new page. Figure 3 shows and explains a sample record list.

The title line consists of the date, job title, and page number, and is printed at the top of each page.

- The date **A** begins in position 1 of the title line.
- The job title **B** is in the center of the title line.
- The page number **C** is at the right of the title line.

Following the title line are headings and detail records.

A line of headings is printed after the title line on each page **D** and after each time the record type changes **E** . (Note that in this example there are two record types: one containing the fields CUSTOMER NUMBER, DATE, INVOICE NUMBER, and AMOUNT; and one containing the fields CUSTOMER NUMBER,DATE, INVOICE NUMBER,CREDIT, and CREDIT REASON.) Each heading line has two blank lines preceding it and one blank line following it. The detail records are single spaced.

In this example, AMOUNT and CREDIT have been defined as accumulator fields. The file has been sorted by CUSTOMER NUMBER and DATE before it was printed. The CUSTOMER NUMBER has also been defined as a control field. Each time CUSTOMER NUMBER changes, the accumulator fields **F** and **G** and their headings **H** are printed.

Subtotals are printed starting in position 40 with two spaces between each field.

An asterisk is also printed after the rightmost subtotal **I** . The final totals are printed at the end of the job **J** .

```
      A                          B                        C
      01/18/74               WEEKLY CASH RECEIPTS         PAGE  1

    D CUSTOMER NUMBER  DATE      INVOICE NUMBER  AMOUNT

            1313       10/21/7-     XC1231      124.29
            1313       10/23/7-     XC2114       78.25
            1313       10/30/7-     XC4312       35.16


      CUSTOMER NUMBER  DATE      INVOICE NUMBER  CREDIT  CREDIT REASON

            1313       10/31/7-     IN1319       28.76   UNUSED MATERIAL
                                           H ----> AMOUNT          CREDIT
                                               F 237.70        G 28.76 * I


      CUSTOMER NUMBER  DATE      INVOICE NUMBER  AMOUNT

            14121      10/30/7-     XC4314      705.00
                                                   AMOUNT          CREDIT
                                                   705.00           0.00 *


  E ---> CUSTOMER NUMBER  DATE   INVOICE NUMBER  AMOUNT

            21884      10/21/7-     XC1259       28.30
            21884      10/30/7-     XC4313       95.23
                                                   AMOUNT          CREDIT
                                                   123.53           0.00 *


      CUSTOMER NUMBER  DATE      INVOICE NUMBER  AMOUNT

            34129      10/26/7-     XC2815       12.95


      CUSTOMER NUMBER  DATE      INVOICE NUMBER  CREDIT  CREDIT REASON

            34129      10/28/7-     IN1308        6.50   WRONG PART NUMBER
                                                   AMOUNT          CREDIT
                                                    12.95           6.50 *
                                                  1079.18  <- J ->  35.26 **
```

Figure 3. Sample Record List

Detailed Summary List

If you respond yes to the prompt
   IS THIS A SUMMARY LIST?
the following prompt appears:
   SHOULD DETAIL RECORDS BE LISTED?
Detail records are the transaction records in your file. They contain the fields you named in the job setup prompting sequence.

If you respond yes to the preceding prompt, you indicate that a detailed summary list will be printed. If you respond no, you indicate that a nondetailed summary list will be printed.

A detailed summary list contains all of the record types and fields that you specify in the setup step prompting sequence. Headings are printed once at the top of each new page. Figure 4 shows and explains a sample detailed summary list.

The title line consists of the date, job title, and page number, and is printed at the top of each page.

- The date A begins in position 1 of the title line.
- The job title B is in the center of the title line.
- The page number C is at the right of the title line.

Following the title line are headings and detail records.

A line of headings is printed after the title line of each page D. The detail records are single spaced following the column headings E.

There is *one* column heading line for each page of the report for all of the record types listed. If a detail record does not contain the field indicated in the column heading, blanks appear in the detail record F.

In this example, there are two record types: one containing the fields CUSTOMER NUMBER, DATE, INVOICE NUMBER, and AMOUNT; and one containing the fields CUSTOMER NUMBER, DATE, INVOICE NUMBER, CREDIT, and CREDIT REASON.

AMOUNT and CREDIT have been defined as accumulator fields. The file has been sorted by CUSTOMER NUMBER and DATE before it was printed. The CUSTOMER NUMBER has also been defined as a control field. Each time CUSTOMER NUMBER changes, a line of subtotals is printed G. (Notice that the subtotals are printed directly under their headings. The headings for the subtotals are not printed each time the subtotals are printed.) An asterisk is also printed after the rightmost subtotal.

The final totals are printed at the end of the job H.

```
A                               B                                   C
10/30/7-                   WEEKLY CASH RECEPTS                      PAGE   1

D CUSTOMER NUMBER  DATE      INVOICE NUMBER    AMOUNT     CREDIT  CREDIT REASON
        1313       10/21/7-      XC1231        124.29     <----F---->
        1313       10/23/7-      XC2114         78.25
        1313       10/30/7-      XC4312         35.16
        1313       10/31/7-      IN1319                   28.76   UNUSED MATERIAL

                                               237.70     28.76 * G

        14121      10/30/7-      XC4314        705.00

  E                                            705.00      0.00 *

        21884      10/21/7-      XC1259         28.30
        21884      10/30/7-      XC4313         95.23

                                               123.53      0.00 *

        34129      10/26/7-      XC2815         12.95
        34129      10/28/7-      IN1308                    6.50    WRONG PART NUMBER

                                                12.95      6.50 *
                                              1079.18  H  35.26 **
```

Figure 4. Sample Detailed Summary List

Nondetailed Summary List

If you respond yes to the prompt
IS THIS A SUMMARY LIST?
the following prompt appears:
SHOULD DETAIL RECORDS BE LISTED?
If you respond no to the preceding prompt, you indicate that a nondetailed summary list will be printed.

A nondetailed summary list provides a summarization or a composite report of file activity. It lists only subtotals of accumulator fields without printing the transaction (detail) records. Subtotals are printed each time a control field changes. A final total is printed at the end of the job. Control fields can also be printed in the summary list if so defined in the setup step prompting sequence. If control fields are not defined for the job, the accumulated totals are printed only at the end of the job.

Figure 5 shows and explains a sample nondetailed summary list.

The title line consists of the date, job title, and page number, and is printed at the top of each page.

- The date **A** begins in position 1 of the title line.
- The job title **B** is in the center of the title line.
- The page number **C** is at the right of the title line.

Following the title line are headings and accumulator field totals.

A line of headings **D** is printed after the title line on each page.

In this example, AMOUNT and CREDIT have been defined as accumulator fields. The file has been sorted by customer number before it was printed. The customer number has also been defined as a control field. Each time the customer number changes, the subtotals are printed and an asterisk is printed after the rightmost subtotal **E** . Final totals are printed at the end of the job **F** .

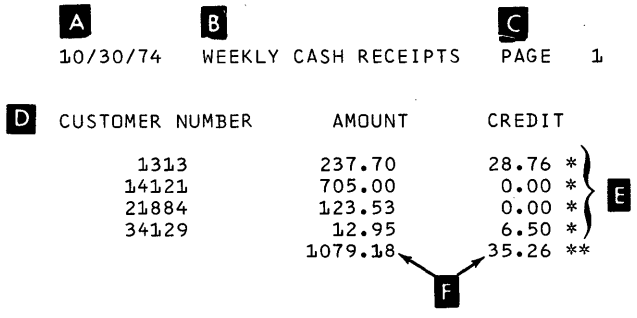


Figure 5. Sample Nondetailed Summary List

This chapter explains the purpose of the DFU setup step, the command statements you use, and the prompting sequences that occur. It also provides two examples of the DFU job setup step. Refer to Appendix F for a sample form to use when creating a format description with the DFU setup step.

PURPOSE OF THE DFU SETUP STEP

The single purpose of the DFU setup step is to create a format description. The following are true of the format description:

- The format description is a *load member.*
- You cannot run a DFU job (do the actual creating, maintaining, displaying, or printing of a data file) until there is a format description that describes the file processing.
- The format description will be saved on disk if you specify a name for it in the command statement.
- The job run begins immediately after the format description is built. The next time you run the job, this format description will already exist and you will skip the setup step and begin with the job run (explained in Chapter 4, 5, 6, or 7). To prevent the job from running at this time, press the EOJ command key when the prompt for the first record appears.
- The format description built for one job can be used for another similar job. For example, once you build a format description to create a data file, you can use it for updating, displaying, and listing the same file.

COMMAND STATEMENTS AND PROMPTING SEQUENCES

The setup step requires that you interact with DFU using the keyboard to enter the proper command statement, then respond to prompts that appear on the display screen.

Figures 6, 7, and 8 show the command statement and prompting sequences for:

- Creating and maintaining a data file—Figure 6. (Since the prompting sequence is nearly identical for these two functions, the prompts are shown in the same figure.)
- Displaying a data file—Figure 7.
- Printing reports from information in a data file—Figure 8.

The *lettered* prompts in Figures 6, 7, and 8 appear if you key the one-word command statement (ENTER, UPDATE, INQUIRY, or LIST) and have DFU prompt you for the necessary command statement information.

They can be skipped if you initially key the complete command statement. Appendix A lists and explains these command statements.

The *numbered* prompts in Figures 6, 7, and 8 are those in which you supply information about your file and how you want DFU to process it.

As shown in the example below, the prompts appear on line 5 of the display screen. The cursor is positioned on line 6 for your response. Lines 1 through 4 initially show the first four DFU attributes that DFU has built from the RPG II source member.

```
1
2          *FILE    SALESORD     50
3          *KEY                   5       50
4    01    *RECORD
5          *CODE    C       H              1
6    SHOULD RECORDS BE PRINTED?
     _
```

Prompt Sequence	Prompt	Response	Notes	Next Prompt
A	←READY	ENTER if you are creating a data file UPDATE if you are maintaining a data file		B
B	ENTER FILENAME FOR FILE TO BE CREATED or ENTER FILENAME FOR FILE TO BE MAINTAINED	Filename	Your response specifies the name of the file to be created or maintained. When creating a file, this name must not be the same as an existing file on disk. When maintaining a file, this name must be that of a file that already exists on disk. The name must be eight characters or less and must begin with an alphabetic character (A through Z, #, $, or @).	C
C	ENTER NAME OF FORMAT DESCRIPTION (THE DEFAULT NAME IS #DFUOBJ)	Format description name	Your response specifies the name of the format description that DFU will build and save as a *load member* in the system library. The name must be eight characters or less and must begin with an alphabetic character. Since the format description is saved and given the name you specify, you need only remember this name in order to skip the setup step the next time the job is run. If you do not want to save the format description, press the ENTER key without keying a response for the prompt (this is called a null response). DFU then builds the format description in a system *load member* called #DFUOBJ. DFU removes this format description from the system after you run the job.	D
D	ENTER RPG II SOURCE MEMBER NAME	RPG II source member name	Your response specifies the name of the RPG II *source member* in the system library. The name must be eight characters or less and must begin with an alphabetic character.	1
1	SHOULD RECORDS BE PRINTED?	YES command key	Records are printed when they are entered or updated.	2
		NO command key	Records are not printed when they are entered; however, any updated or deleted record will still be printed.	2

Figure 6 (Part 1 of 4). Prompting Sequence for Creating and Maintaining a Data File

Prompt Sequence	Prompt	Response	Notes	Next Prompt
2	ENTER VALUES FOR 'DELETE CODE, POSITION'	Delete code, Position	Specifies the delete code and position that DFU uses when updating records and the DELETE command key is pressed. The record displayed is marked for deletion by putting the delete code in the position specified.	3, 4, or 5
		Null response	Assumes the default of a blank in position one of the record.	3, 4, or 5
3	SHOULD DFU GENERATE RECORD KEYS FOR YOU?	YES command key	DFU automatically generates the record key for each record you enter or add to the end of a file.	5
	Prompt 3 occurs only if the record key field is defined as five positions long in the RPG II source member.	NO command key	The key must be supplied for each record entered or added to the end of a file.	4 or 5
4	ARE YOUR RECORD KEYS ALL NUMERIC?	YES command key	DFU right-justifies and inserts leading zeros in the keys that are entered.	5
	Prompt 4 does not occur if the record key field is packed (DFU assumes the key is numeric), if the key length is greater than 15 (DFU assumes the key is alphameric), or if a yes response was made to prompt 3.	NO command key	The complete record key must be entered to process a record.	5
5	ENTER COLUMN HEADING FOR KEYS	Heading	Key the heading desired for the record key field. The maximum length is 16 characters.	6
		Null response	DFU uses *KEY as the heading for the record key field.	6
6	ENTER COL SPACING VALUE (0-9, DEFAULT=1)	Number	Specifies the number of spaces between the fields displayed and printed.	7
		Null response	DFU assumes a spacing value of 1.	7
7	ENTER TITLE	Title	Specifies the title of the job. The maximum title length is 24 characters.	8
		Null response	Specifies that there is no title for the job.	8
8	xx—ANY FIELDS FROM THIS RECORD TYPE?	YES command key	Specifies that fields from this record type are to be processed. You will describe the field(s) in prompts 9 through 14.	9
	xx is a record identification indicator from the RPG II source member.	NO command key	Specifies that no fields are to be processed from this record type. DFU repeats this prompt for the next record type defined in the RPG II source members.	8 or 15

Prompts 9, 10, 11, 12, 13, and 14 appear in sequence, allowing you to specify the name, heading, and features for each field of this record type you want DFU to process. The prompting sequence can be interrupted by pressing the REC ADV key. This will cause prompt 9 to reappear.

Figure 6 (Part 2 of 4). Prompting Sequence for Creating and Maintaining a Data File

Prompt Sequence	Prompt	Response	Notes	Next Prompt
9	ENTER FIELD NAME	Field name	Key the name of the next field DFU should process. The maximum field name length is six characters. If REC ADV is pressed after keying the name, the field name is used as the heading and no DFU features are specified for this field. Prompt 9 reappears.	9 or 10
		DUP key	The field name on line 4 of the display screen appears on line 6. If the ENTER key is pressed, this field name is entered and prompt 10 appears. If REC ADV is pressed, the field name is used as the heading and no DFU features (auto dup, accumulation, or self-checking) are specified. Use the ROLL↑ or ROLL↓ key to display the desired field on line 4 before pressing DUP.	10
		Null response	Specifies there are no more fields to describe in this record type.	8 or 15
10	ENTER COLUMN HEADING	Heading	Specifies the heading for the field. The maximum heading length is 16 characters.	9 or 11
			If REC ADV is pressed after keying the heading, it causes prompt 9 to appear.	9 or 11
		Null response (ENTER or REC ADV)	For either key, the field name is used as the heading. ENTER causes prompt 11 to appear. REC ADV causes prompt 9 to appear.	
11	IS THIS AN AUTO-DUP FIELD?	YES command key	Specifies that this field is an auto dup field.	12
		NO command key	Specifies that this field is not an auto dup field.	12
		REC ADV	Specifies that this field is not an auto dup field.	9
12	ACCUMULATE THIS FIELD?	YES command key	Specifies that this field is an accumulator field.	13
		NO command key	Specifies that this field is not an accumulator field.	13
		REC ADV	Specifies that this field is not an accumulator field.	9
13	IS THIS A SELF-CHECK FIELD?	YES command key	Specifies that this field is a self-check field.	14
		NO command key	Specifies that this field is not a self-check field.	9
		REC ADV	Specifies that this field is not a self-check field.	9

Figure 6 (Part 3 of 4). Prompting Sequence for Creating and Maintaining a Data File

Prompt Sequence	Prompt	Response	Notes	Next Prompt
14	ENTER SELF-CHECK NUMBER (10 or 11)	10	Specifies that the modulus 10 formula will be used to verify the self-check digit.	**9**
		11	Specifies that the modulus 11 formula will be used to verify the self-check digit.	**9**
15			Prompt 15 appears when there are no more record types to describe. At the left is an example of the display that occurs at this time. Lines 1, 2, 3, and 4 of the display show the first four lines of the DFU specifications that DFU created from your prompt responses. The DFU specifications can be displayed or modified at this time. (Chapter 10, *DFU Specifications,* explains this). The PRINT REC command key can be pressed to list the DFU attributes and specifications on the printer.	
		EOJ command key	Press the EOJ command key to end the setup step and build the format description.	
			If you are creating a data file, prompt E appears. If you are maintaining a data file, **the job run begins** (see *Steps for Maintaining a File* in Chapter 5).	**E**
E	ENTER NUMBER OF RECORDS TO BE IN FILE	Number of records	Your response specifies the maximum number of records that will be keyed into the file. **The job run begins** (see *Steps for Creating a Data File* in Chapter 4).	

```
        *ENT/UPD*LIST   X,1
        *KEY    *GENKEY *KEY
2       *TITLE  DAILY SALES ORDERS
01      *RECORD
HIT EOJ CMD KEY TO CONTINUE PROCESSING,
OR YOU MAY NOW UPDATE DFU SPECS
```

Figure 6 (Part 4 of 4). Prompting Sequence for Creating and Maintaining a Data File

Prompt Sequence	Prompt	Response	Notes	Next Prompt
A	←READY	INQUIRY		B
B	ENTER FILENAME OF INQUIRY FILE	Filename	Your response specifies the name of the file to be displayed. This name must be that of a file that already exists on disk. The name must be eight characters or less and must begin with an alphabetic character (A through Z, #, $, or @).	C
C	ENTER NAME OF FORMAT DESCRIPTION (THE DEFAULT NAME IS #DFUOBJ)	Format description name	Your response specifies the name of the format description that DFU will build and save as a *load member* in the system library. The name must be eight characters or less and must begin with an alphabetic character. Since the format description is saved and given the name you specify, you need only remember this name in order to skip the setup step the next time the job is run. If you do not want to save the format description, press the ENTER key without keying a response for the prompt (this is called a null response). DFU then builds the format description in a system *load member* called #DFUOBJ. DFU removes this format description from the system after you run the job.	D
D	ENTER RPG II SOURCE MEMBER NAME	RPG II source member name	Your response specifies the name of the RPG II *source member* in the system library. The name must be eight characters or less and must begin with an alphabetic character.	1 or 2
1	ARE YOUR RECORD KEYS ALL NUMERIC?	YES command key	DFU right-justifies and inserts leading zeros in the keys that are entered.	2
	Prompt 1 does not occur if the record key field is packed (DFU assumes the key is numeric) or if the key length is greater than 15 (DFU assumes the key is alphameric). Prompt 2 appears instead.	NO command key	The complete record key must be entered to display a record.	2
2	ENTER COLUMN HEADING FOR KEYS	Heading	Key the column heading desired for the record key field. The maximum length is 16 characters.	3
		Null response	DFU uses *KEY as the heading for the record key field.	3

Figure 7 (Part 1 of 3). Prompting Sequence for Displaying a Data File

Prompt Sequence	Prompt	Response	Notes	Next Prompt
3	ENTER COL SPACING VALUE (0-9,DEFAULT=1)	Number	Specifies the number of spaces between the fields displayed.	4
		Null response	DFU assumes a spacing value of one.	4
4	ENTER TITLE	Title of job	Specifies the title of the job. The maximum title length is 24 characters.	5
		Null response	Specifies that there is no title for the job.	5
5	xx—ANY FIELDS FROM THIS RECORD TYPE?	YES command key	Specifies that fields from this record type are to be displayed. You will describe the fields in prompts 6 and 7.	6
	xx is a record identification indicator from the RPG II source member.	NO command key	Specifies that no fields are to be displayed from this record type. DFU repeats this prompt for the next record type defined in the RPG II source member.	5 or 8
	Prompts 6 and 7 appear, allowing you to specify the name and heading for each field in this record type you want to display.			
6	ENTER FIELD NAME	Field name	Key the name of the field DFU should display. If REC ADV is pressed after keying the name, the field name is used as the heading and prompt 6 reappears.	6 or 7
		DUP key	The field named on line 4 of the display screen appears on line 6. If the ENTER key is pressed, this field name is entered and prompt 7 appears. If REC ADV is pressed, the field name is used as the heading. Use the ROLL↑ or ROLL↓ key to display the desired field on line 4 before pressing DUP.	7
		Null response	Specifies there are no more fields to display in this record type.	5 or 8

Figure 7 (Part 2 of 3). Prompting Sequence for Displaying a Data File

Prompt Sequence	Prompt	Response	Notes	Next Prompt
7	ENTER COLUMN HEADING	Column heading	Specifies the column heading for the field to be displayed.	6
		Null response	The field name is used as the column heading.	6
8			If there are no more record types to describe, prompting ends. At the left is an example of the display that occurs at this time. Lines 1, 2, 3, and 4 of the display show the first four lines of the DFU specifications that DFU created from your prompt responses. The DFU specifications can be displayed or modified at this time (Chapter 10, *DFU Specifications,* explains this). The PRINT REC command key can be pressed to list the DFU attributes and specifications on the printer.	

```
         *INQUIRY
         *KEY     *NUMERIC  *KEY
2        *TITLE   DAILY SALES ORDERS
01       *RECORD
HIT EOJ CMD KEY TO CONTINUE PROCESSING,
OR YOU MAY NOW UPDATE DFU SPECS
```

		EOJ command key	Press the EOJ command to end the setup step and build the format description.	
			The job run begins (see *Displaying the Records* in Chapter 6).	

Figure 7 (Part 3 of 3). Prompting Sequence for Displaying a Data File

Prompt Sequence	Prompt	Response	Notes	Next Prompt
A	←READY	LIST or LIST ,,,,,,,master file name	The master file name parameter specifies the name of the second file from which DFU prints information.	B
B	ENTER FILENAME OF LIST FILE	Filename of list file	Your response specifies the name of the file to be printed. This name must be that of a file that already exists on disk. The name must be eight characters or less and must begin with an alphabetic character (A through Z, #, $, or @).	C
C	ENTER NAME OF FORMAT DESCRIPTION (THE DEFAULT NAME IS #DFUOBJ)	Format description name	Your response specifies the name of the format description that DFU will build and save as a *load member* in the system library. The name must be eight characters or less and must begin with an alphabetic character. Since the format description is saved and given the name you specify, you need only remember this name in order to skip the setup step the next time the job is run. If you do not want to save the format description, press the ENTER key without keying a response for the prompt (this is called a null response). DFU then builds the format description in a system *load member* called #DFUOBJ. DFU removes this format description from the system after you run the job.	D
D	ENTER RPG II SOURCE MEMBER NAME	RPG II source member name for the list file	Your response specifies the name of the RPG II *source member* in the system library that describes the list file. The name must be eight characters or less and must begin with an alphabetic character.	E or 1
	Note: Prompts E and F appear only if you have keyed the initial command statement as: LIST ,,,,,,,master file name			
E	KEY RPG II SOURCE NAME FOR ABOVE FILE	RPG II source member name for master file	Your response specifies the name of the RPG II *source member* for the master file. The name must be eight characters or less and must begin with an alphabetic character.	F

Figure 8 (Part 1 of 8). Prompting Sequence for Printing Reports from Information in a Data File

Prompt Sequence	Prompt	Response	Notes	Next Prompt
F	ENTER FIELD NAME FOR MASTER FILE KEY	Field name	Your response specifies the name of the field in the list file's RPG II source member that DFU uses to retrieve master file records. The field you name must have the same attributes as the key field in the master file.	1
1	IS THIS A SUMMARY LIST?	YES command key	DFU prints a summary list of your file. DFU prints a common column heading for all record types. (Chapter 2, *DFU Features,* explains the summary list.)	2
		NO command key	DFU prints a record list of your file. DFU prints a new line of column headings each time the record type changes. (Chapter 2, *DFU Features,* explains the record list.)	3
2	SHOULD DETAIL RECORDS BE LISTED? Prompt 2 appears only if the response to prompt 1 was yes.	YES command key	DFU prints a detailed summary list of your file in which all selected records are printed. (Chapter 2, *DFU Features,* explains the detailed summary list.)	3
		NO command key	DFU prints a nondetailed summary list of your file in which only totals and control fields are printed at the time of control breaks. (Chapter 2, *DFU Features,* explains the nondetailed summary list.)	6
3	SHOULD RECORD KEYS BE PRINTED?	YES command key	DFU prints a record key as the first field for each record in the file.	4
		NO command key	DFU does not print a record key in the list.	6
4	ARE YOUR RECORD KEYS ALL NUMERIC? Prompt 4 does not occur if the record key field is packed (DFU assumes the key is numeric) or if the key length is greater than 15 (DFU assumes the key is alphameric). Prompt 5 appears instead.	YES command key	Leading zeros are suppressed, and if the key is negative, the sign is printed.	5
		NO command key	Record key is printed without editing.	5
5	ENTER COLUMN HEADING FOR KEYS Prompt 4 appears only if the response to prompt 3 was yes.	Heading	Key the heading desired for the record key field. The maximum length is 16 characters.	6
		Null response	DFU uses *KEY as the heading for the record key field.	6
6	ENTER COL SPACING VALUE (0-9,DEFAULT=1)	Number	Specifies the number of spaces between the printed columns.	7
		Null response	DFU assumes a spacing value of 1.	7
7	ENTER TITLE	Title	Specifies the title of the report that DFU prints at the top of each page. The maximum title length is 24 characters.	8
		Null response	Specifies that there is no title for the report.	8

Figure 8 (Part 2 of 8). Prompting Sequence for Printing Reports from Information in a Data File

Prompt Sequence	Prompt	Response	Notes	Next Prompt
8	xx–ANY FIELDS FROM THIS RECORD TYPE? xx is a record identification indicator from the RPG II source member. For a summary list, this prompt reappears until all record types in the file have been prompted. For a record list, prompts 9, 10, and 11 appear after prompt 8 allowing you to specify the fields for this record type before prompt 8 reappears.	YES command key	Specifies that fields from this record type are to be printed. You describe the fields in prompts 9, 10, or 11.	8 or 9
		NO command key	Specifies that no fields are to be printed from this record type. DFU repeats this prompt for the next record type defined in the RPG II source member.	8, 9, 20 or 22
		ALL	Specifies that all record types are to be included in the summary list. This response is valid only for a summary list the first time prompt 8 appears.	9
	Prompts 9, 10, and 11 appear in sequence, allowing you to specify the name and heading for each field of this record type that you want listed. You also can specify as many as ten fields to be used as accumulator fields. The prompting sequence can be interrupted by pressing the REC ADV key. This causes prompt 9 to reappear.			
9	✓ ENTER FIELD NAME	Field name	Name of the next field DFU prints. If you are printing a report from two data files, the list file and related master file, your response can be the name of a field in either file. If you press REC ADV after keying the name, the field name is used as the heading and the field is not used as an accumulator field. Prompt 9 reappears.	9 or 10
		DUP key	The field named on line 4 of the display screen appears on line 6. If you press the ENTER key, this field name is entered and prompt 10 appears. If you press REC ADV, the field name is used as the heading and the field is not used as an accumulator field. Prompt 9 then reappears. For a summary list, the attribute list associated with the last record type will be displayed on lines 1 through 4 when this prompt first appears. Field names included in any record type can be entered. Use of the DUP key to enter field names associated with prior record types is possible if preceded by ROLL↓. Use the ROLL↑ or ROLL↓ key to display the desired field on line 4 before pressing DUP.	9 or 10
		Null response	Specifies there are no more fields to describe in this record type (or in the file if you are defining a summary list).	8, 20, or 22
		RESULT FIELD command key	Specifies that you want to define a calculated result field. You can define a maximum of six result fields.	12

Figure 8 (Part 3 of 8). Prompting Sequence for Printing Reports from Information in a Data File

Prompt Sequence	Prompt	Response	Notes	Next Prompt
10	✓ ENTER COLUMN HEADING	Heading	Specifies the heading DFU prints for the field. The maximum heading length is 16 characters. If you press REC ADV after keying the heading, prompt 9 reappears.	9 or 11
		Null response (ENTER or REC ADV)	For either key, the field name is used as the heading. ENTER causes prompt 11 to appear. REC ADV causes prompt 9 to reappear.	9 or 11
11	ACCUMULATE THIS FIELD?	YES command key	Specifies that this field is an accumulator field.	9
		NO command key	Specifies that this field is not an accumulator field.	9
		REC ADV key	Specifies that this field is not an accumulator field.	9
12	ENTER RESULT NAME (IF DESIRED)	Name	Specifies name under which result is saved for future calculations.	13
		Null response	Specifies that the result is not saved for future calculations.	
13	ENTER LENGTH OF RESULT FIELD (1-15)	Number	Specifies the length of the result field. The maximum length is 15 positions.	14
14	ENTER DEC POS IN RESULT FIELD (0-9)	Number	Specifies the number of decimal positions to appear in the result field. The maximum number of decimal positions is 9.	15
		Null response	Specifies that no decimal positions appear in the result field.	15
15	ENTER COLUMN HEADING	Heading	Specifies the heading DFU prints for the result field. The maximum heading length is 16 characters.	16
		Null response	Specifies the heading DFU prints for the result field is *RESULT.	16
16	ACCUMULATE THIS FIELD?	YES command key	Specifies that DFU accumulates this field.	17
		NO command key	Specifies that DFU does not accumulate this field.	17
17	ENTER FIRST FACTOR IN RESULT FIELD	Field name	Specifies the field DFU uses as the first factor in calculating the result field.	18
	DFU places this factor in the result field.	DUP key	The field named on line 4 of the display screen appears on line 6. Press ENTER to specify this field as the first factor in calculating the result field.	18
		Constant	Specifies the constant DFU uses as the first factor in calculating the result field. The *Result Fields* information in Chapter 2 lists the rules for entering a constant.	18

Figure 8 (Part 4 of 8). Prompting Sequence for Printing Reports from Information in a Data File

Prompt Sequence	Prompt	Response	Notes	Next Prompt
18	ENTER NEXT FACTOR IN RESULT FIELD	Field name	Specifies the field DFU uses as the next factor in calculating the result field.	19
	You can specify a maximum of four factors.	DUP key	The field named on line 4 of the display screen appears on line 6. Press ENTER to specify this field as the next factor in calculating the result field.	19
		Constant	Specifies the constant DFU uses as the next factor in calculating the result field. The *Result Fields* information in Chapter 2 lists the rules for entering a constant.	19
		Null response	Specifies that there are no more factors used to calculate the result field.	9
19	ENTER OPERATION (ADD, SUB, MULT, DIV)	ADD, SUB, MULT, or DIV	The factor you specified in prompt 17 is combined with the result field by the operation you specify: • ADD: Add the factor to the result field. • SUB: Subtract the factor from the result field. • MULT: Multiply the factor and the result field. • DIV: Divide the result field by the factor.	18
20	ANY CONTROL FIELDS?	YES command key	Specifies that you want control fields to be used in this list. (Chapter 2, *DFU Features*, explains the use of control fields.)	21
	This prompt appears only if one or more accumulator fields were defined in prompts 9, 10, and 11.	NO command key	Specifies that control fields are not used.	22
21	ENTER CONTROL FIELD NAME	Field name	Specifies the name of the field that you want used as a control field. The name must be six characters or less.	21 or 22
	This prompt appears only if you responded yes to prompt 20. This prompt is repeated until a null response is made or until five control fields have been named. The order in which you name the control fields determines how DFU uses them.	Null response	Specifies that there are no more control fields to name.	22
		DUP key	The field named on line 4 of the display screen appears on line 6. Press ENTER to continue with the prompts. Use the ROLL↑ or ROLL↓ key to display the proper field name on line 4 of the display screen before pressing the DUP key.	21 or 22

Figure 8 (Part 5 of 8). Prompting Sequence for Printing Reports from Information in a Data File

Prompt Sequence	Prompt	Response	Notes	Next Prompt
22	SHOULD FILE BE SORTED?	YES command key	Specifies that DFU sorts the file before listing it.	23
		NO command key	Specifies that DFU does not sort the file.	25
23	ENTER SORT FIELD NAME	Field name	Specifies the name of the field that you want used as a sort field. The name must be six characters or less.	24
	This prompt appears only if you responded yes to prompt 22. This prompt is repeated until a null response is made or until five sort fields have been named. The order in which you name the sort fields determines how DFU will use them.	Null response	Specifies that there are no more sort fields to name.	25
		DUP key	The field named on line 4 of the display screen appears on line 6. Press ENTER to continue with the prompts. Use the ROLL↑ or ROLL↓ key to display the proper field name before pressing the DUP key.	24
24	IS THIS SORT FIELD ASCENDING?	YES command key	DFU sorts the field in ascending sequence.	23 or 25
		NO command key	DFU sorts the field in descending sequence.	23 or 25
25	SELECT RECORDS BASED ON FIELD VALUES?	YES command key	Specifies that DFU selects records based on field values. You specify the record selection criteria in prompts 26 through 31.	26
		NO command key	Specifies that DFU does not select records based on field values.	32
26	ENTER NAME OF SELECT FIELD (FACTOR 1)	Field name	Specifies that DFU selects records based on the value of this field.	27
		DUP key	The field named on line 4 of the display screen appears on line 6. Press ENTER to specify this field as one DFU should use to select records.	27
27	ENTER CONDITION (EQ, NE, GT, LT, GE, LE)	EQ, NE, GT, LT, GE, or LE	Specifies the comparison DFU makes between the field you named in prompt 26 and the factor you will specify in prompt 29 or 30. One of the following comparisons is made, depending on your response: ● EQ: Factor 1 equals factor 2. ● NE: Factor 1 does not equal factor 2. ● GT: Factor 1 is greater than factor 2. ● LT: Factor 1 is less than factor 2. ● GE: Factor 1 is greater than or equal to factor 2. ● LE: Factor 1 is less than or equal to factor 2.	28

Figure 8 (Part 6 of 8). Prompting Sequence for Printing Reports from Information in a Data File

Prompt Sequence	Prompt	Response	Notes	Next Prompt
28	IS FACTOR 2 A CONSTANT?	YES command key	Specifies that DFU compares the select field with the constant you name in prompt 29.	29
		NO command key	Specifies that DFU compares the select field with the field you name in prompt 30.	30
29	ENTER CONSTANT (FACTOR 2 VALUE)	Constant	Specifies the constant DFU compares with the select field to determine if the record is listed. The *Record Selection* information in Chapter 2 lists the rules for entering a constant.	31 or 32
30	ENTER FIELD NAME (FACTOR 2 VALUE)	Field name	Specifies the field DFU compares with the select field to determine if the record is selected.	31 or 32
		DUP key	The field named on line 4 of the display screen appears on line 6. Press ENTER to specify this field as the one DFU compares with the select field.	31 or 32
31	ENTER NEXT SELECT FIELD RELATION (OR, AND)	OR	Specifies that DFU selects the record if either the previous criteria or the next criteria you specify in prompts 26 through 31 is met.	26
		AND	Specifies that DFU selects the record if both the previous criteria and the next criteria you specify in prompts 26 through 31 is met.	26
		Null response	Specifies that there is no more record selection criteria.	32

Figure 8 (Part 7 of 8). Prompting Sequence for Printing Reports from Information in a Data File

Prompt Sequence	Prompt	Response	Notes	Next Prompt
32			When there are no more record types, control fields for accumulation, control fields for sorting, and records to select to describe, the prompting ends. At the left is an example of the display that occurs at this time. Lines 1, 2, 3, and 4 of the display show the first four lines of the DFU specifications that DFU created from your prompt responses. The DFU specifications can be displayed or modified at this time (Chapter 10, *DFU Specifications*, explains this). The PRINT REC command key can be pressed to list the DFU attributes and specifications on the printer.	
		EOJ command key	Press the EOJ command key to end the setup step and build the format description.	E
E	INDICATE IF THE FILE IS TO BE SORTED BEFORE LISTING ('NOSORT',SORT)	SORT	Specifies that DFU prints a sorted report.	
		NOSORT	Specifies that DFU does not sort the report before printing it.	
		Null response	Specifies that DFU does not sort the report before printing it.	
			DFU then prints the report.	

```
         *LIST   *SUMMARY*DETAIL
         *KEY
1        *TITLE  CUSTOMER SALES ANALYSIS
02       *RECORD
HIT EOJ CMD KEY TO CONTINUE PROCESSING,
OR YOU MAY NOW UPDATE DFU SPECS
```

Figure 8 (Part 8 of 8). Prompting Sequence for Printing Reports from Information in a Data File

JOB SETUP EXAMPLES

The following are two examples of DFU job setup steps. Example 1 describes how you prepare a job that allows an operator to create and maintain a customer order file. The examples in Chapters 4 and 5 describe how the operator creates and maintains the file.

Example 2 describes how you set up a job to list the file that the operator creates in the example in Chapter 4. In example 2, there is no operator action required. As soon as you complete the job setup step and create the format description, DFU lists the file.

Example 1 — Job Description

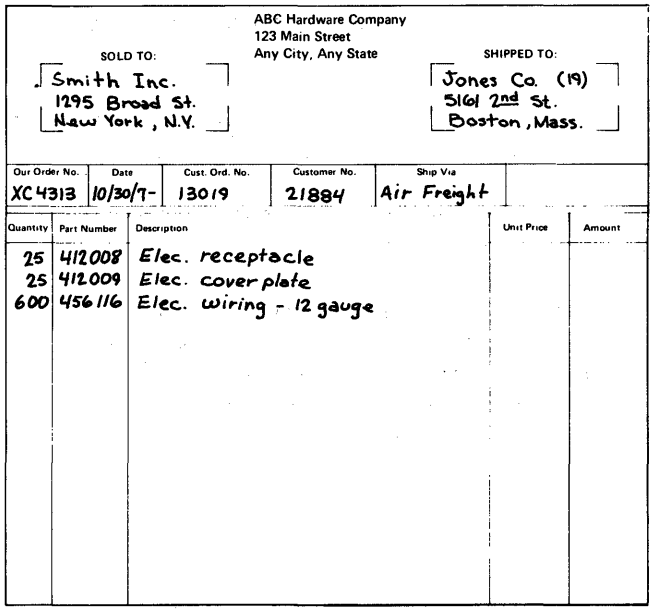
This example describes how to set up a DFU job that allows an operator to create and maintain a customer order file. There are four basic steps in this example:

1. The clerk receives customer orders by phone, writes them on forms, then verifies that each form is complete. Figure 9 shows a sample order form. The order forms constitute the customer order file that the operator creates in the *Example of Creating a Data File* in Chapter 4 and maintains in the *Example of Maintaining a Data File* in Chapter 5.

2. The clerk totals the quantities from each order form on an adding machine and clips the adding machine tape to the customer order.

3. An operator keys the customer order into the customer order file. DFU lists each order as it is keyed, adds the quantities on the order, and prints the quantity total for each order form.

4. The operator compares the quantity totals printed in step 3 with the totals on the adding machine tape. If they match, the operator enters the next customer order. If they do not match, the operator must correct the error before continuing.



ABC Hardware Company
123 Main Street
Any City, Any State

SOLD TO:
Smith Inc.
1295 Broad St.
New York, N.Y.

SHIPPED TO:
Jones Co. (19)
5161 2nd St.
Boston, Mass.

Our Order No.	Date	Cust. Ord. No.	Customer No.	Ship Via
XC4313	10/30/7-	13019	21884	Air Freight

Quantity	Part Number	Description	Unit Price	Amount
25	412008	Elec. receptacle		
25	412009	Elec. cover plate		
600	456116	Elec. wiring - 12 gauge		

Figure 9. Sample Order Form

Example 1 — Setting Up the Job

This example assumes the following:

- The required RPG II source member, called SALESRPG in this example, describes the customer order file and is now on disk for your use. Figure 10 shows the file description and input specifications of the RPG II source member. From the RPG II source member, you see:

  1. The file name is SALESORD (columns 7-14 of the file description specifications).

  2. The customer order file is an indexed file (column 15 of the file description specifications).

  3. The key field is five positions long (columns 29-30 of the file descriptions specifications).

  4. The key field begins at position 46 in each record of the file (columns 35-38 of the file description specifications).

  5. The customer order file contains two record types. Record type 01 is a heading record. It is identified by an H in position 1 of the record and contains the following constant information about the order form:

Field Name	Heading on the Order Form
CODE	This is the record identification code. It does not appear on the order form.
CUSTNO	Customer No.
ORDNO	Our Order No.
CUSORD	Cust. Ord. No.
DATE	Date
SHPTO	Shipped to
SHPVIA	Ship Via

Record type 02 is a detail record. It is identified by a D in position 1 of the record and contains information for each line on the order form. Record type 02 has the following fields:

Field Name	Heading on the Order Form
CODE	This is the record identification code. It does not appear on the order form.
CUSTNO	Customer No.
ORDNO	Our Order No.
QTY	Quantity
PARTNO	Part Number

Note: Chapter 8 describes the RPG II source member.

- You set up the job so DFU:

  1. Prompts the operator for the header record fields both:
     a. In the order that they appear on the order form, and
     b. With the headings as they appear on the order form.

  2. Prompts the operator for the detail record fields both:
     a. In the order that they appear on the order form, and
     b. With the headings as they appear on the order form.

  These steps increase the operator's order entry speed and decrease the chance of error.

- DFU prints each record after the operator keys it from the order form.

- In this example, the delete code is an X in position 1 of the record, the position reserved for the record identification code.

- DFU generates keys for the records.

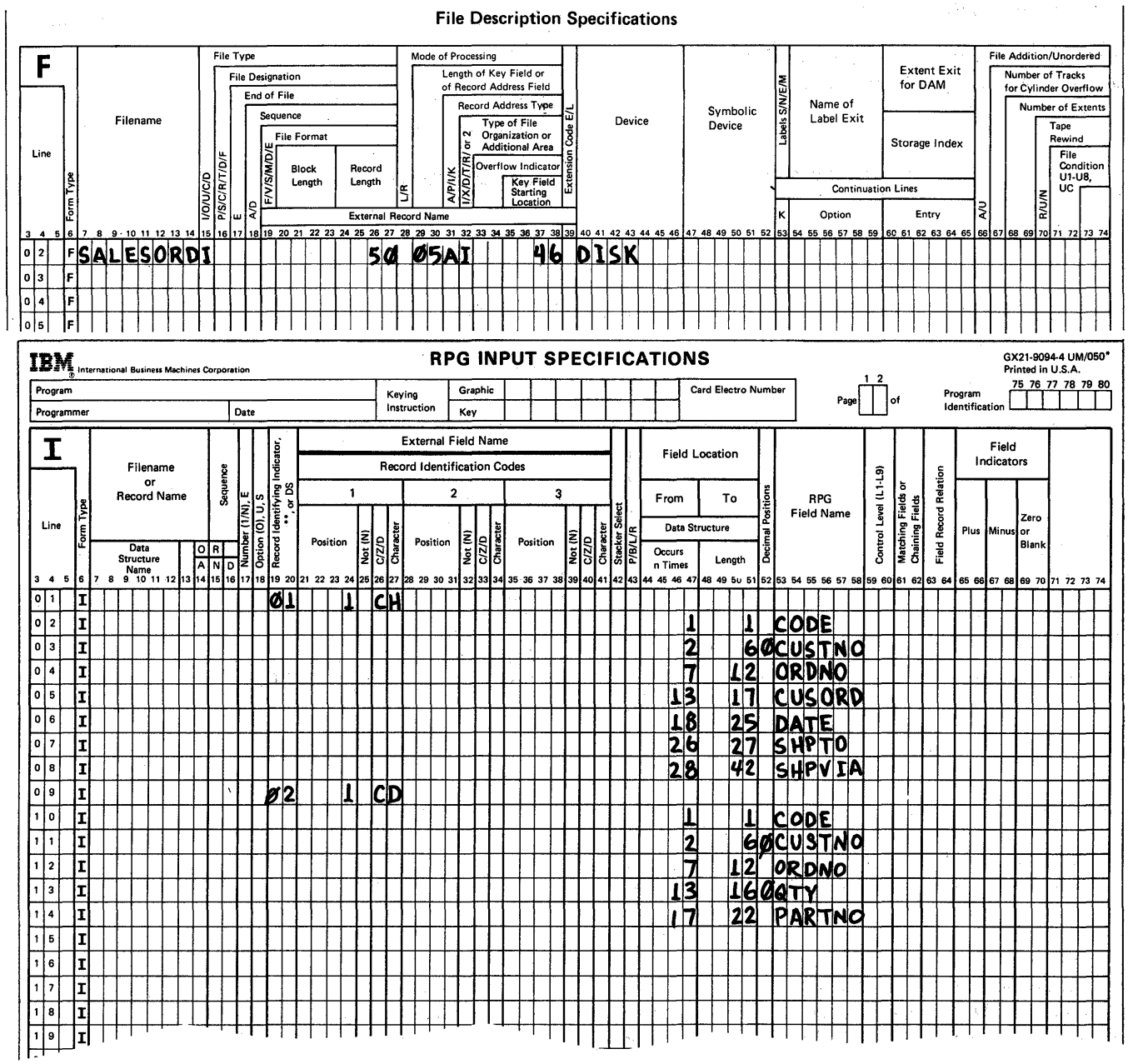
- The title of the job is Daily Sales Orders.

- DFU duplicates the ORDNO and CUSTNO fields from the header record into each detail record of an order form.

- DFU accumulates a total for the QTY field on each order form and prints the total for each order form.

- You make the file big enough to contain a maximum of 50 records.

- The format description you build in this setup step is called ORDERFMT. DFU saves it for you on disk where the operator can use it over and over for creating and maintaining the customer order file.



Note: The command statement you key to use the source entry utility to create this RPG II source member is:

```
SEU SALESRPG,R
```

Figure 10. The RPG II Source Member, SALESRPG

Example 1 – Prompting Sequence

Begin the job setup step by keying the word ENTER and pressing the ENTER key. The following three prompts appear:

```
ENTER FILENAME FOR FILE TO BE CREATED
_
```

Key SALESORD and press ENTER.

```
ENTER NAME OF FORMAT DESCRIPTION
(THE DEFAULT NAME IS #DFUOBJ)
_
```

Key ORDERFMT and press ENTER.

```
ENTER RPG II SOURCE MEMBER NAME
_
```

Key SALESRPG and press ENTER.

The prompt

DFU ATTRIBUTES BEING BUILT

appears momentarily on the display screen. DFU is converting the RPG II source member into DFU attributes. When DFU completes the conversion, the following display appears:

```
        *FILE   SALESORD    50
        *KEY                 5       50
01      *RECORD
        *CODE   C       H                1
SHOULD RECORDS BE PRINTED?
_
```

This display is the first of a series of prompts in the job setup step. Note that four lines of the DFU attributes are displayed on lines 1 through 4, the prompt is on line 5, and the cursor is positioned on line 6 to accept your response. Respond with the YES command key since you want DFU to print each record after the operator keys it.

Figure 11 shows the entire list of DFU attributes that have been built from the RPG II source member. You can use the ROLL ↑ or ROLL ↓ keys to scan the attributes on the display screen.

For information about DFU attributes, refer to Chapter 9.

```
****  DFU ATTRIBUTES  ****

         *FILE    SALESORD     50
         *KEY                   5       50
01       *RECORD
         *CODE    C       H                1
                  CODE          1          1
                  CUSTNO        5.0        6
                  ORDNO         6         12
                  CUSORD        5         17
                  DATE          8         25
                  SHPTO         2         27
                  SHPVIA       15         42
02       *RECORD
         *CODE    C       D                1
                  CODE          1          1
                  CUSTNO        5.0        6
                  ORDNO         6         12
                  QTY           4.0       16
                  PARTNO        6         22
```

Figure 11. DFU Attributes Created for Example 1

As you read the following prompts, you may find it helpful to follow the explanation of the prompting sequence for creating and maintaining a data file in Figure 6 of this chapter. This provides a complete explanation of each prompt, the responses allowed, and the results of keying a response.

```
         *FILE    SALESORD     50
         *KEY                   5       50
01       *RECORD
         *CODE    C       H                1
ENTER VALUES FOR 'DELETE CODE,POSITION'
_
```

If you have scanned the DFU attributes, return them as shown in this display screen to their original position.

Key X,1 on line 6 and press ENTER to indicate a delete code of X in position 1 of the record.

```
         *FILE    SALESORD     50
         *KEY                   5       50
01       *RECORD
         *CODE    C       H                1
SHOULD DFU GENERATE RECORD KEYS FOR YOU?
_
```

Respond with the YES command key. (This prompt occurs because the record key field is defined as five positions long in the RPG II source member.)

```
        *FILE   SALESORD    50
        *KEY                 5       50
01      *RECORD
        *CODE   C       H            1
ENTER COLUMN HEADING FOR KEYS
_
```

Press the ENTER key to indicate that the default heading, *KEY, should be used.

```
        *FILE   SALESORD    50
        *KEY                 5       50
01      *RECORD
        *CODE   C       H            1
ENTER COL SPACING VALUE (0-9,DEFAULT=1)
_
```

Key 2 and press ENTER to indicate that you want two spaces to separate field columns on the display screen and printed output.

```
        *FILE   SALESORD    50
        *KEY                 5       50
01      *RECORD
        *CODE   C       H            1
ENTER TITLE
_
```

Key DAILY SALES ORDERS, your title for the job, then press the ENTER key to continue with the prompts.

```
01      *RECORD
        *CODE   C       H            1
                CODE            1    1
                CUSTNO          5.0  6
01-- ANY FIELDS FROM THIS RECORD TYPE?
_
```

Respond with the YES command key. (The first record type specified in the RPG II source member is 01. The first four lines of the display show the DFU attributes for this record type.)

```
01      *RECORD
        *CODE   C       H            1
                CODE            1    1
                CUSTNO          5.0  6
ENTER FIELD NAME
_
```

Prompting now begins to allow you to key information for the fields to be entered from the header record. Key ORDNO, the name of the first field the operator will enter, and press the ENTER key.

Note that the order in which you name the fields does not have to be the same as the order of the fields in the DFU attributes. You should key the field names as they appear on the order form—not necessarily in the same order as in the DFU attributes.

```
01      *RECORD
        *CODE   C       H               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER COLUMN HEADING
-
```

Key OUR ORDER NO., the heading for the ORDNO field, and press the ENTER key. Assign headings so they correspond to the headings used on the order forms. Then, the headings on the order form will match the headings that appear on the display screen when the file is being created. This results in less confusion and less chance for a keying mistake when creating the file.

```
01      *RECORD
        *CODE   C       H               1
                CODE            1       1
                CUSTNO          5.0     6
IS THIS AN AUTO-DUP FIELD?
-
```

Respond with the YES command key to define ORDNO as an auto dup field.

```
01      *RECORD
        *CODE   C       H               1
                CODE            1       1
                CUSTNO          5.0     6
ACCUMULATE THIS FIELD?
-
```

Press the REC ADV key to indicate that ORDNO is not an accumulator field and you want to skip the prompts for the self check feature.

```
01      *RECORD
        *CODE   C       H               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER FIELD NAME
-
```

Key DATE, the next field, and press the ENTER key to continue with the prompts.

```
01      *RECORD
        *CODE   C       H               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER COLUMN HEADING
-
```

Since the heading is also DATE, press REC ADV without keying a response. This causes (1) the field name, DATE, to be used as the heading, and (2) the

ENTER FIELD NAME

prompt to appear, skipping the prompts for the auto-dup, accumulator, and self check features.

```
01      *RECORD
        *CODE   C       H               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER FIELD NAME
_
```

You can use the DUP key to help speed keying field names. CUSORD is the field name that should be entered next. Before you press the DUP key, CUSORD must be on line 4 of the display screen. Press the ROLL↑ key twice to bring the CUSORD field of the DFU attributes to line 4. Press the DUP key.

```
                CODE            1       1
                CUSTNO          5.0     6
                ORDNO           6       12
                CUSORD          5       17
ENTER FIELD NAME
CUSORD_
```

CUSORD appears on line 6. Press ENTER to continue with the prompts.

```
                CODE            1       1
                CUSTNO          5.0     6
                ORDNO           6       12
                CUSORD          5       17
ENTER COLUMN HEADING
_
```

Key the heading CUST. ORD. NO. and press REC ADV to return to the
ENTER FIELD NAME
prompt.

```
                CODE            1       1
                CUSTNO          5.0     6
                ORDNO           6       12
                CUSORD          5       17
ENTER FIELD NAME
_
```

Press the ROLL↓ key twice to bring CUSTNO to line 4 of the display screen. Press the DUP key.

```
01      *RECORD
        *CODE   C       H               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER FIELD NAME
CUSTNO_
```

CUSTNO appears on line 6.
Press the ENTER key to continue with the prompts.

```
01      *RECORD
        *CODE   C       H               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER COLUMN HEADING
_
```

Key the heading, CUSTOMER NO., then press the ENTER key to continue with the prompts.

```
01      *RECORD
        *CODE   C       H               1
                CODE            1       1
                CUSTNO          5.0     6
IS THIS AN AUTO-DUP FIELD?
_
```

Respond with the YES command key because CUSTNO is an auto dup field.

```
01      *RECORD
        *CODE   C       H               1
                CODE            1       1
                CUSTNO          5.0     6
ACCUMULATE THIS FIELD?
_
```

Press the REC ADV key because CUSTNO is not an accumulator field.

```
01      *RECORD
        *CODE   C       H               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER FIELD NAME
_
```

Press the ROLL↑ key four times to bring SHPTO, the next field to describe, to line 4 of the display. Press the DUP key.

```
                ORDNO           6       12
                CUSORD          5       17
                DATE            8       25
                SHPTO           2       27
ENTER FIELD NAME
SHPTO_
```

SHPTO appears on line 6. Press the ENTER key to continue with the prompts.

```
                ORDNO           6       12
                CUSORD          5       17
                DATE            8       25
                SHPTO           2       27
ENTER COLUMN HEADING
_
```

Key the heading, SHIP TO, and press REC ADV to return to the

ENTER FIELD NAME

prompt.

```
                ORDNO          6       12
                CUSORD         5       17
                DATE           8       25
                SHPTO          2       27
ENTER FIELD NAME
_
```

Press the ROLL↑ key once to bring SHPVIA, the next field to describe, to line 4 of the display. Then press the DUP key.

```
                CUSORD         5       17
                DATE           8       25
                SHPTO          2       27
                SHPVIA        15       42
ENTER FIELD NAME
SHPVIA_
```

SHPVIA appears on line 6. Press ENTER to continue with the prompts.

```
                CUSORD         5       17
                DATE           8       25
                SHPTO          2       27
                SHPVIA        15       42
ENTER COLUMN HEADING
_
```

Key the heading SHIP VIA, then press REC ADV.

```
                CUSORD         5       17
                DATE           8       25
                SHPTO          2       27
                SHPVIA        15       42
ENTER FIELD NAME
_
```

Since all fields of record type 01 have been described, press the ENTER key.

```
02      *RECORD
        *CODE   C       D               1
                CODE           1        1
                CUSTNO         5.0      6
02-- ANY FIELDS FROM THIS RECORD TYPE?
_
```

Respond with the YES command key. You will describe the ORDNO, CUSTNO, QTY, and PARTNO fields in record type 02. You will define ORDNO and CUSTNO as auto dup fields and QTY as an accumulator field.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER FIELD NAME
_
```

Press the ROLL↑ key once, then press the DUP key instead of keying ORDNO. Press the ENTER key to continue.

```
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
                ORDNO           6       12
ENTER COLUMN HEADING
_
```

Key the heading, OUR ORDER NO., then press the ENTER key to continue with the prompts.

```
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
                ORDNO           6       12
IS THIS AN AUTO-DUP FIELD?
_
```

Respond with the YES command key.

```
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
                ORDNO           6       12
ACCUMULATE THIS FIELD?
_
```

Press the REC ADV key.

```
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
                ORDNO           6       12
ENTER FIELD NAME
_
```

Press the ROLL↓ key once, then press the DUP key instead of keying CUSTNO. Press the ENTER key to continue.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER COLUMN HEADING
-
```

Key the heading, CUSTOMER NO., then press the ENTER key to continue with the prompts.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
IS THIS AN AUTO-DUP FIELD?
-
```

Respond with the YES command key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ACCUMULATE THIS FIELD?
-
```

Press the REC ADV key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER FIELD NAME
-
```

Press the ROLL↑ key twice, then press the DUP key instead of keying QTY. Press the ENTER key to continue.

```
                CODE            1       1
                CUSTNO          5.0     6
                ORDNO           6       12
                QTY             4.0     16
ENTER COLUMN HEADING
-
```

Key the heading QUANTITY, then press the ENTER key.

```
                CODE          1        1
                CUSTNO        5.0      6
                ORDNO         6        12
                QTY           4.0      16
IS THIS AN AUTO-DUP FIELD?
_
```

Respond with the NO command key.

```
                CODE          1        1
                CUSTNO        5.0      6
                ORDNO         6        12
                QTY           4.0      16
ACCUMULATE THIS FIELD?
_
```

Respond with the YES command key.

```
                CODE          1        1
                CUSTNO        5.0      6
                ORDNO         6        12
                QTY           4.0      16
IS THIS A SELF-CHECK FIELD?
_
```

Respond with the NO command key.

```
                CODE          1        1
                CUSTNO        5.0      6
                ORDNO         6        12
                QTY           4.0      16
ENTER FIELD NAME
_
```

Press the ROLL↑ key once, then press the DUP key instead of keying PARTNO. Press the ENTER key to continue.

```
                CUSTNO        5.0      6
                ORDNO         6        12
                QTY           4.0      16
                PARTNO        6        22
ENTER COLUMN HEADING
_
```

Key the heading, PART NUMBER, and press the REC ADV key to return to the
ENTER FIELD NAME
prompt.

```
                CUSTNO        5.0      6
                ORDNO         6        12
                QTY           4.0      16
                PARTNO        6        22
ENTER FIELD NAME
_
```

Since there are no more fields to describe, press the ENTER key.

```
         *ENT/UPD*LIST    X,1
         *KEY     *GENKEY *KEY
2        *TITLE   DAILY SALES ORDERS
01       *RECORD
HIT EOJ CMD KEY TO CONTINUE PROCESSING,
OR YOU MAY NOW UPDATE DFU SPECS
```

DFU determines that there are no more record types defined in the RPG II source member and shows the first four DFU specifications on lines 1 through 4 of the display screen. As explained in lines 5 and 6 of the display, you can modify the specifications at this time (refer to Chapter 10, *DFU Specifications*, for an explanation of how to do this) or you can end the setup step. In this example, you do not need to modify the specifications. Press the PRINT REC command key to list the DFU attributes and specifications. Then press the EOJ command key to build the format description. Figure 12 shows the DFU attributes and specifications built in this setup step.

An

ENTER NUMBER OF RECORDS TO BE IN FILE

prompt appears. Key 50 and press ENTER, since you anticipate a maximum of 50 header and detail records to be in this file.

This ends the setup step for this job. You have created the format description, ORDERFMT, that allows the operator to create and maintain the customer order file. The job run automatically begins. For this example, the operator is required to create the SALESORD file and key three order forms into it; the example in Chapter 4 shows how this is done. The operator is also required to maintain the file by adding, deleting, and changing information in it; the example in Chapter 5 shows how this is done.

```
****  DFU ATTRIBUTES  ****

         *FILE    SALESORD     50
         *KEY                   5        50
01       *RECORD
         *CODE    C       H                1
                  CODE           1         1
                  CUSTNO         5.0       6
                  ORDNO          6         12
                  CUSORD         5         17
                  DATE           8         25
                  SHPTO          2         27
                  SHPVIA         15        42
02       *RECORD
         *CODE    C       D                1
                  CODE           1         1
                  CUSTNO         5.0       6
                  ORDNO          6         12
                  QTY            4.0       16
                  PARTNO         6         22


****   DFU SPECIFICATIONS   ****

         *ENT/UPD *LIST      X,1
         *KEY     *GENKEY    *KEY
2        *TITLE   DAILY SALES ORDERS
01       *RECORD
         *D       ORDNO      OUR ORDER NO.
         *        DATE       DATE
         *        CUSORD     CUST. ORD. NO.
         *D       CUSTNO     CUSTOMER NO.
         *        SHPTO      SHIP TO
         *        SHPVIA     SHIP VIA
02       *RECORD
         *D       ORDNO      OUR ORDER NO.
         *D       CUSTNO     CUSTOMER NO.
         *ADD     QTY        QUANTITY
         *        PARTNO     PART NUMBER
```

Figure 12. DFU Attributes and DFU Specifications Created for Example 1.

Example 2—Job Description

This example explains how you prepare a job to print a report from the SALESORD customer order file. (Example 1 in this chapter shows how you create the format description for creating SALESORD; the example in Chapter 4 shows how you create SALESORD.)

Customers frequently call and request up-to-date information on the items they have purchased during the month. They need to know quantities ordered for each item and the sales price. You have decided that the one-time report in Figure 13 would satisfy their requests:

To print this report, DFU needs the following:

Part number
Part description
Quantity ordered
Price per item
Dollar value of sales

The part number and quantity ordered are fields in SALESORD. The part description and price per item are fields in a related inventory master file named INVENTRY. The dollar value of sales is a result field generated by DFU when it prints the report.

CUSTOMER SALES ANALYSIS

Date_____ Page Number _____

Customer Number	Part Number	Description	Quantity	Price	Sales Amount
____	_____	_____	_____	____	_____
____	_____	_____	_____	____	_____
			Accumulated quantity for each customer		Accumulated sales amount for each customer
____	_____	_____	_____	____	_____
			Total Quantities		Total Sales Amount

Figure 13. Report Format

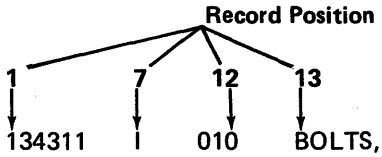
Example 2—Setting Up the Job

You require the following information to set up the job:

- Two RPG II source members must be on disk before you begin: SALESRPG and INVENRPG. Figure 10 in this chapter describes SALESRPG. Figure 14 describes INVENRPG. The SEU command statements that allow you to enter these source members are:

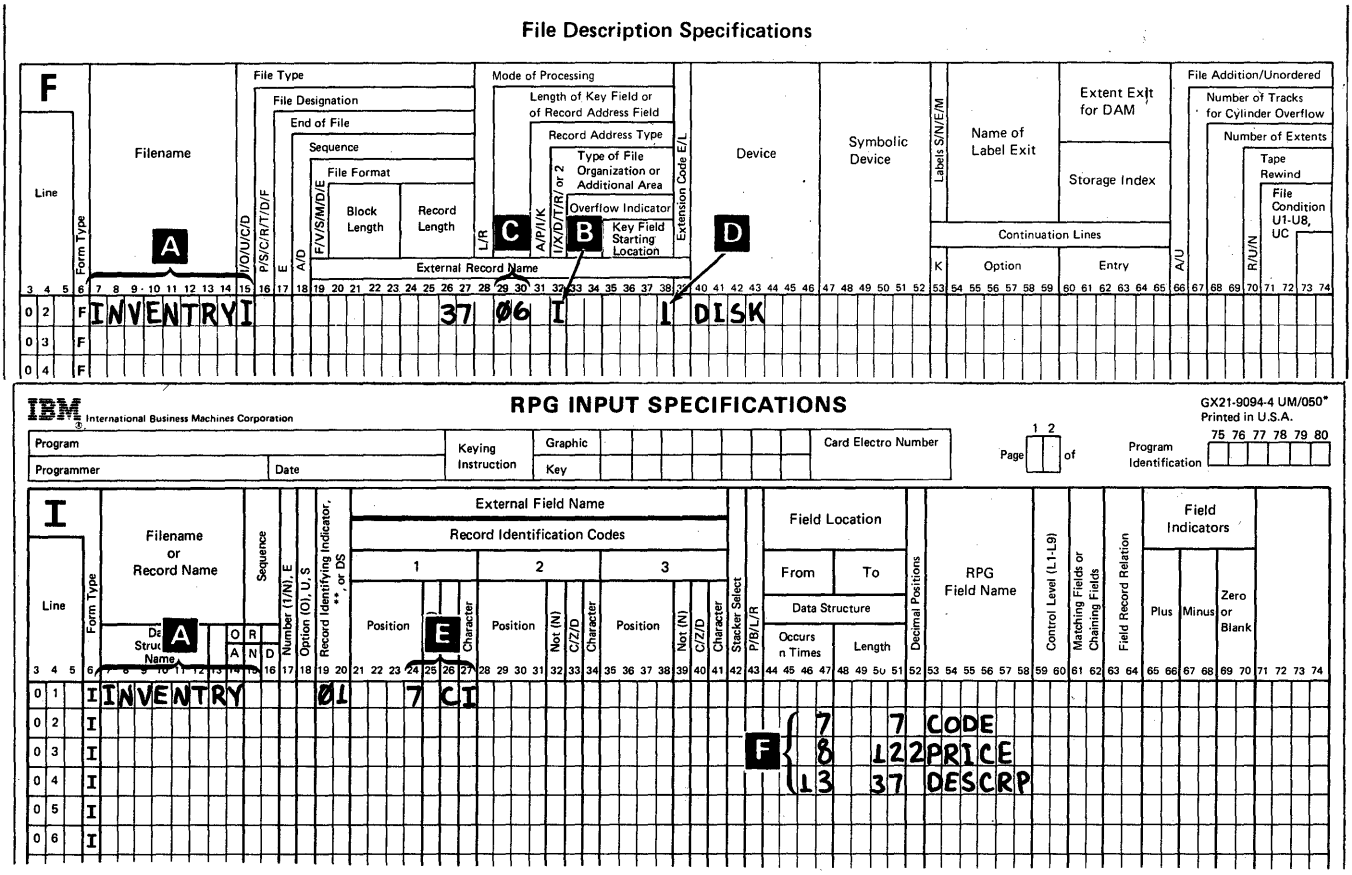
```
SEU  SALESRPG,R
SEU  INVENRPG,R
```

- Two data files must be on disk before you begin: SALESORD and INVENTRY. Example 1 in this chapter describes how you set up the job to create SALESORD; the example in Chapter 4 describes how you create it. The following shows the INVENTRY file:




```
134311  I  010  BOLTS, LAG 1/4 x 3 INCHES
159715  I  549  FLASHLIGHT, HEAVY-DUTY
213157  I  086  SHELF BRACKETS, 4 x 4
412008  I  075  ELEC. RECEPTACLE
412009  I  039  ELEC. COVER PLATE
456116  I  098  ELEC. WIRING-12 GAUGE
612695  I  5000 SINK, OVAL YELLOW
612723  I  5500 SINK, OVAL BLUE
612783  I  6000 SINK, OVAL GREEN
```

Note: Use DFU to create the file by (1) setting up the job as explained in Chapter 3, then (2) creating the file as explained in Chapter 4.



Note: The command statement you key to use the source entry utility to create this RPG II source member is SEU INVENRPG,R.

From the RPG II source member, you see:

- The file name is INVENTRY A .
- The inventory file is an indexed file B .
- The key field is six positions C and begins in position 1 of the record D . The part number is the record key. DFU uses it to retrieve records from this file.
- There is one record type identified by an I in position 7 E .
- The fields in each record are F :

  CODE — Record identification code
  PRICE — Price per item
  DESCRP — Description of the item

Figure 14. RPG II Source Member Named INVENRPG that Describes the Inventory File

- In the prompting sequence of the setup step, you indicate the following:

  1. The report is a summary list in which only one record type is listed.

  2. The title of the report is Customer Sales Analysis.

  3. The report headings are:
     Customer number
     Part number
     Description
     Quantity
     Price
     Sales amount

  4. DFU accumulates the item quantities.

  5. For each transaction, DFU calculates a result field, sales amount, by multiplying the item quantity by the price per item.

  6. DFU accumulates the sales amounts.

  7. DFU sorts the report by customer number and part number and prints accumulated quantities and sales amounts each time the customer number changes.

  8. For this report, you are only interested in printing transactions from certain customers: customer numbers 01313, 14121, and 21884.

Example 2—Prompting Sequence

Begin the job setup step by keying:

```
LIST ,,,,,,,INVENTRY
```

The following prompts appear:

```
LIST ,,,,,,,INVENTRY
ENTER FILENAME OF LIST FILE
_
```

Key SALESORD, the name of the transaction file, then press the ENTER key.

```
LIST ,,,,,,,INVENTRY
ENTER FILENAME OF LIST FILE
SALESORD
ENTER NAME OF FORMAT DESCRIPTION
 (THE DEFAULT NAME IS #DFUOBJ)
```

Press the ENTER key. There is not an existing format description for this list, and since it is a one-time report, you do not want to name and save the format description DFU creates.

```
SALESORD
ENTER NAME OF FORMAT DESCRIPTION
 (THE DEFAULT NAME IS #DFUOBJ)

ENTER RPG II SOURCE MEMBER NAME
_
```

Key SALESRPG, the name of the RPG II source member that describes SALESORD, then press the ENTER key.

The prompt

DFU ATTRIBUTES ARE BEING BUILT

appears momentarily on the display screen. DFU is converting the RPG II source member SALESRPG into DFU attributes. When DFU completes the conversion, the two following displays appear:

```
INVENTRY
KEY RPG II SOURCE NAME FOR ABOVE FILE
_
```

Key INVENRPG, the name of the RPG II source member that describes the related master file, then press the ENTER key.

```
ENTER FIELD NAME FOR MASTER FILE KEY
_
```

Key PARTNO, the name of the field in SALESORD that DFU uses to retrieve records from INVENTRY, then press the ENTER key.

The prompt

DFU ATTRIBUTES ARE BEING BUILT

appears momentarily on the display screen. DFU is converting the RPG II source member INVENRPG into DFU attributes and adding them to the end of the DFU attributes created from SALESRPG. Figure 15 shows the entire list of DFU attributes that have been built from the RPG II source members. You can use the ROLL↑ or ROLL↓ keys to scan the attributes on the display screen.

For information about DFU attributes, refer to Chapter 9.

When DFU completes the conversion, the following display appears:

```
         *FILE   SALESORD    50
         *KEY                 5      50
01       *RECORD
         *CODE   C      H              1
IS THIS A SUMMARY LIST?
_
```

This display is the first of a series of prompts in the job setup step. Note that four lines of the DFU attributes are displayed on lines 1 through 4, the first prompt is on line 5, and the cursor is positioned on line 6 to accept your response.

If you scanned the DFU attributes, return them as shown in this display screen to their original position. Press the YES command key. DFU lists only one record type — the transaction detail records.

```
****  DFU ATTRIBUTES  ****

         *FILE   SALESORD    50
         *KEY                 5      50
01       *RECORD
         *CODE   C      H              1
                 CODE          1       1
                 CUSTNO        5.0     6
                 ORDNO         6      12
                 CUSORD        5      17
                 DATE          8      25
                 SHPTO         2      27
                 SHPVIA       15      42
02       *RECORD
         *CODE   C      D              1
                 CODE          1       1
                 CUSTNO        5.0     6
                 ORDNO         6      12
                 QTY           4.0    16
                 PARTNO        6      22
         *FILE   INVENTRY    37
         *KEY    PARTNO        6       6
                 CODE          1       7
                 PRICE         5.2    12
                 DESCRP       25      37
```

Figure 15. DFU Attributes for Example 2

```
        *FILE   SALESORD    50
        *KEY                 5       50
01      *RECORD
        *CODE   C       H            1
SHOULD DETAIL RECORDS BE LISTED?
_
```

Press the YES command key. DFU prints all transactions for a customer.

```
        *FILE   SALESORD    50
        *KEY                 5       50
01      *RECORD
        *CODE   C       H            1
SHOULD RECORD KEYS BE PRINTED?
_
```

Press the NO command key. The keys are not meaningful in this report, since they were generated by DFU for this file; therefore, they are not printed.

```
        *FILE   SALESORD    50
        *KEY                 5       50
01      *RECORD
        *CODE   C       H            1
ENTER COL SPACING VALUE (0-9,DEFAULT=1)
_
```

Press the ENTER key.

```
        *FILE   SALESORD    50
        *KEY                 5       50
01      *RECORD
        *CODE   C       H            1
ENTER TITLE
_
```

Key CUSTOMER SALES ANALYSIS, the title DFU prints on each page of the report, then press the ENTER key.

```
01      *RECORD
        *CODE   C       H               1
                CODE            1       1
                CUSTNO          5.0     6
01-- ANY FIELDS FROM THIS RECORD TYPE?
-
```

Press the NO command key, since DFU requires no information from the header record (type 01) to print the report.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
02-- ANY FIELDS FROM THIS RECORD TYPE?
-
```

Press the YES command key, since DFU lists detail records (type 02) in the report.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER FIELD NAME
-
```

DFU has begun prompting for the fields from the detail record to print in the report. Press the DUP key, then press the ENTER key to indicate that the customer number field, CUSTNO, is printed first in the report.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER COLUMN HEADING
-
```

Key CUSTOMER NUMBER, the heading for the customer number field, and press the REC ADV key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER FIELD NAME
-
```

Press the ROLL↑ key three times to bring PARTNO to line 4 of the display. Press the DUP key, then the ENTER key to indicate that the part number field, PARTNO, is printed following the customer number field in the report.

```
                CUSTNO        5.0      6
                ORDNO         6        12
                QTY           4.0      16
                PARTNO        6        22
ENTER COLUMN HEADING
_
```

Key PART NUMBER, the heading for the part number field, and press the REC ADV key.

```
                CUSTNO        5.0      6
                ORDNO         6        12
                QTY           4.0      16
                PARTNO        6        22
ENTER FIELD NAME
_
```

Press the ROLL↑ key five times to bring DESCRP to line 4 of the display. Press the DUP key, then press the ENTER key.

```
       /
        *KEY    PARTNO        6        6
                CODE          1        7
                PRICE         5.2      12
                DESCRP        25       37
ENTER COLUMN HEADING
_
```

Key DESCRIPTION, the heading for the description field, and press the REC ADV key.

```
        *KEY    PARTNO        6        6
                CODE          1        7
                PRICE         5.2      12
                DESCRP        25       37
ENTER FIELD NAME
_
```

Press the ROLL↓ key six times to bring QTY to line 4 of the display. Press the DUP key, then press the ENTER key.

```
                CODE          1        1
                CUSTNO        5.0      6
                ORDNO         6        12
                QTY           4.0      16
ENTER COLUMN HEADING
_
```

Key QUANTITY, the heading for the quantity field, and press the ENTER key.

```
                CODE          1        1
                CUSTNO        5.0      6
                ORDNO         6        12
                QTY           4.0      16
ACCUMULATE THIS FIELD?
_
```

Press the YES command key. DFU totals the quantity each customer purchased during the month.

```
                CODE          1        1
                CUSTNO        5.0      6
                ORDNO         6        12
                QTY           4.0      16
ENTER FIELD NAME
_
```

Press the ROLL↑ key five times to bring PRICE to line 4 of the display. Press the DUP key, then press the REC ADV key.

```
        *FILE   INVENTRY    37
        *KEY    PARTNO       6        6
                CODE         1        7
                PRICE        5.2      12
ENTER FIELD NAME
_
```

Press the RESULT FIELD command key. This allows you to specify the result field (the amount of each transaction) you want DFU to calculate in the report.

DFU calculates this field by multiplying the QTY field by the PRICE field for each transaction.

```
        *FILE   INVENTORY   37
        *KEY    PARTNO       6        6
                CODE         1        7
                PRICE        5.2      12
ENTER RESULT NAME (IF DESIRED)
_
```

Press the ENTER key (null response). This indicates that you do not wish to save the result for use in future calculations.

```
        *FILE   INVENTRY    37
        *KEY    PARTNO       6        6
                CODE         1        7
                PRICE        5.2      12
ENTER LENGTH OF RESULT FIELD (1-15)
_
```

Key 7, since the largest expected sales amount is $10,000.00, and press the ENTER key.

```
        *FILE   INVENTRY    37
        *KEY    PARTNO       6        6
                CODE         1        7
                PRICE        5.2      12
ENTER DEC POS IN RESULT FIELD (0-9)
_
```

Key 2, then press the ENTER key.

```
        *FILE   INVENTRY    37
        *KEY    PARTNO       6        6
                CODE         1        7
                PRICE        5.2     12
ENTER COLUMN HEADING
_
```

Key SALES AMOUNT, the heading for the result field, then press the ENTER key.

```
        *FILE   INVENTRY    37
        *KEY    PARTNO       6        6
                CODE         1        7
                PRICE        5.2     12
ACCUMULATE THIS FIELD?
_
```

Press the YES command key.

```
        *FILE   INVENTRY    37
        *KEY    PARTNO       6        6
                CODE         1        7
                PRICE        5.2     12
ENTER FIRST FACTOR IN RESULT FIELD
_
```

Press the ROLL↓ key five times to bring QTY to line 4 of the display. Press the DUP key, then press the ENTER key.

```
                CODE         1        1
                CUSTNO       5.0      6
                ORDNO        6       12
                QTY          4.0     16
ENTER NEXT FACTOR IN RESULT FIELD
_
```

Press the ROLL↑ key five times to bring PRICE to line 4 of the display. Press the DUP key, then press the ENTER key.

```
        *FILE   INVENTRY    37
        *KEY    PARTNO       6        6
                CODE         1        7
                PRICE        5.2     12
ENTER OPERATION (ADD,SUB,MULT,DIV)
_
```

Key MULT, then press the ENTER key.

```
        *FILE   INVENTRY    37
        *KEY    PARTNO       6        6
                CODE         1        7
                PRICE        5.2     12
ENTER NEXT FACTOR IN RESULT FIELD
_
```

Press the ENTER key since there are no other factors DFU uses to calculate the result field.

```
        *FILE   INVENTRY    37
        *KEY    PARTNO       6        6
                CODE         1        7
                PRICE        5.2     12
ENTER FIELD NAME
_
```

Press the ENTER key since there are no other fields DFU prints in the report.

```
        *FILE   INVENTRY    37
        *KEY    PARTNO       6        6
                CODE         1        7
                PRICE        5.2     12
ANY CONTROL FIELDS?
_
```

Press the YES command key since DFU prints separate sales amounts and quantity totals for each customer.

```
        *FILE   INVENTRY    37
        *KEY    PARTNO       6        6
                CODE         1        7
                PRICE        5.2     12
ENTER CONTROL FIELD NAME
_
```

Press the ROLL↓ key seven times to bring CUSTNO to line 4 of the display. Press the DUP key, then press the ENTER key. This causes DFU to print accumulated totals each time the customer number changes.

```
02      *RECORD
        *CODE   C       D            1
                CODE         1       1
                CUSTNO       5.0     6
ENTER CONTROL FIELD NAME
_
```

Press the ENTER key since there are no other control fields to define.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
SHOULD FILE BE SORTED?
_
```

Press the YES command key. DFU sorts the report by customer number, then sorts part number within customer number.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER SORT FIELD NAME
_
```

Press the DUP key, then press the ENTER key to define CUSTNO as the first sort field.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
IS THIS SORT FIELD ASCENDING?
_
```

Press the YES command key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER SORT FIELD NAME
_
```

Press the ROLL↑ key three times to bring PARTNO to line 4 of the display. Press the DUP key, then press the ENTER key. This defines PARTNO as the second sort field.

```
                CUSTNO          5.0     6
                ORDNO           6       12
                QTY             4.0     16
                PARTNO          6       22
IS THIS SORT FIELD ASCENDING?
_
```

Press the YES command key.

```
                CUSTNO        5.0      6
                ORDNO         6        12
                QTY           4.0      16
                PARTNO        6        22
ENTER SORT FIELD NAME
_
```

Press the ENTER key since there are no other sort fields in the report.

Now you have supplied all of the information DFU needs to format and print the report. The prompts that follow allow you to select certain records you want DFU to include in the report. In this example, DFU prints transactions for customer numbers 01313, 14121, and 21884.

```
                CUSTNO        5.0      6
                ORDNO         6        12
                QTY           4.0      16
                PARTNO        6        22
SELECT RECORDS BASED ON FIELD VALUES?
_
```

Press the YES command key.

```
                CUSTNO        5.0      6
                ORDNO         6        12
                QTY           4.0      16
                PARTNO        6        22
ENTER NAME OF SELECT FIELD (FACTOR 1)
_
```

Press the ROLL↓ key three times to bring CUSTNO to line 4 of the display. Press the DUP key, then press the ENTER key to define CUSTNO as a select field.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER CONDITION (EQ,NE,GT,LT,LE,GE)
_
```

Key EQ, then press the ENTER key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
IS FACTOR 2 A CONSTANT?
_
```

Press the YES command key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER CONSTANT (FACTOR 2 VALUE)
_
```

Key 01313, then press the ENTER key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER NEXT SELECT FIELD RELATION(OR,AND)
_
```

Key OR, then press the ENTER key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER NAME OF SELECT FIELD (FACTOR 1)
_
```

Press the DUP key, then press the ENTER key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER CONDITION (EQ,NE,GT,LT,LE,GE)
_
```

Key EQ, then press the ENTER key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
IS FACTOR 2 A CONSTANT?
_
```

Press the YES command key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER CONSTANT (FACTOR 2 VALUE)
-
```

Key 14121, then press the ENTER key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER NEXT SELECT FIELD RELATION(OR,AND)
-
```

Key OR, then press the ENTER key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER NAME OF SELECT FIELD (FACTOR 1)
-
```

Press the DUP key, then press the ENTER key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER CONDITION (EQ,NE,GT,LT,LE,GE)
-
```

Key EQ, then press the ENTER key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
IS FACTOR 2 A CONSTANT?
-
```

Press the YES command key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER CONSTANT (FACTOR 2 VALUE)
_
```

Key 21884, then press the ENTER key.

```
02      *RECORD
        *CODE   C       D               1
                CODE            1       1
                CUSTNO          5.0     6
ENTER NEXT SELECT FIELD RELATION(OR,AND)
_
```

Press the ENTER key.

```
        *LIST   *SUMMARY*DETAIL
        *KEY
1       *TITLE  CUSTOMER SALES ANALYSIS
02      *RECORD
HIT EOJ CMD KEY TO CONTINUE PROCESSING,
OR YOU MAY NOW UPDATE DFU SPECS.
```

You have finished responding to the prompts. The first four lines of the display now show DFU specifications that DFU created from your responses.

Press the PRINT REC command key. DFU prints the attributes and specifications shown in Figure 16.

```
****  DFU ATTRIBUTES  ****

        *FILE   SALESORD    50
        *KEY                 5      50
01      *RECORD
        *CODE   C       H            1
                CODE          1      1
                CUSTNO        5.0    6
                ORDNO         6      12
                CUSORD        5      17
                DATE          8      25
                SHPTO         2      27
                SHPVIA        15     42
02      *RECORD
        *CODE   C       D            1
                CODE          1      1
                CUSTNO        5.0    6
                ORDNO         6      12
                QTY           4.0    16
                PARTNO        6      22
        *FILE   INVENTRY    37
        *KEY    PARTNO        6      6
                CODE          1      7
                PRICE         5.2    12
                DESCRP        25     37
```

```
****  DFU SPECIFICATIONS  ****

        *LIST   *SUMMARY*DETAIL
        *KEY
1       *TITLE  CUSTOMER SALES ANALYSIS
02      *RECORD
        *       CUSTNO  CUSTOMER NUMBER
        *       PARTNO  PART NUMBER
        *       DESCRP  DESCRIPTION
        *ADD    QTY     QUANTITY
        *       PRICE   PRICE
7.2     *ADD    *RESULT SALES AMOUNT
ADD             QTY
MULT            PRICE
        *TOTAL  CUSTNO
        *SORTA  CUSTNO
        *SORTA  PARTNO
        *SELECT CUSTNO  EQ
                01313
OR      *SELECT CUSTNO  EQ
                14121
OR      *SELECT CUSTNO  EQ
                21884
```

Figure 16. DFU Attributes and Specifications Created for Example 2

Scan the DFU specifications using the ROLL↑ key to check that they match those in Figure 16. If changes are required, refer to *Updating DFU Specifications* information in Chapter 10.

Press the EOJ command key.

The following prompt appears:

INDICATE IF THE DATA FILE IS TO BE SORTED BEFORE LISTING ('NOSORT', SORT)

Key SORT and press the ENTER key.

Figure 17 shows the report that DFU prints.

```
7-/09/04                        CUSTOMER SALES ANALYSIS                          PAGE    1

CUSTOMER NUMBER  PART NUMBER  DESCRIPTION                    QUANTITY    PRICE  SALES AMOUNT

          1313        134311  BOLTS, LAG 1/4 X 3 INCHES           200     0.10         20.00
          1313        159715  FLASHLIGHT, HEAVY-DUTY                2     5.49         10.98
          1313        213157  SHELF BRACKETS, 4 X 4                16     0.86         13.76
          1313        412008  ELEC. RECEPTACLE                     13     0.75          9.75
          1313        412009  ELEC. COVER PLATE                    13     0.39          5.07

                                                                  244                  59.56 *

         14121        612695  SINK, OVAL YELLOW                    48    50.00       2400.00
         14121        612723  SINK, OVAL BLUE                      48    55.00       2640.00
         14121        612783  SINK, OVAL GREEN                     48    60.00       2880.00

                                                                  144                7920.00 *

         21884        412008  ELEC. RECEPTACLE                     25     0.75         18.75
         21884        412009  ELEC. COVER PLATE                    25     0.39          9.75
         21884        456116  ELEC. WIRING-12 GAUGE               600     0.98        588.00

                                                                  650                 616.50 *
                                                                 1038                8596.06 **
    11 RECORDS PROCESSED
```

Figure 17. Report Printed for Example 2

This chapter explains how to create a data file. It is assumed that the format description already exists. If the format description does not exist, refer to Figure 6 in Chapter 3 for assistance in creating the format description before proceeding. Refer to Appendix F for a sample form to use when creating a data file.

ENTER COMMAND STATEMENT

Note: If you have just completed the job setup step, you will not be required to key the ENTER command statement. Refer directly to *Steps for Entering a Data File* in this chapter.

When creating a file, key the word ENTER and press the ENTER key. A prompt is then issued for the name of the file:

ENTER FILENAME OF FILE TO BE CREATED

Your response gives a name to the data file being created. The filename is a maximum of eight characters in length, and must start with an alphabetic character. The filename must not be a duplicate of an existing file.

The next prompt is:

ENTER NAME OF FORMAT DESCRIPTION
(THE DEFAULT NAME IS #DFUOBJ)

Your response specifies the name of the format description for processing this job.

The next prompt is:

ENTER NUMBER OF RECORDS TO BE IN FILE

Key the number of records the file will contain.

The preceding prompting sequence can be skipped by initially keying the complete command statement or the name of the procedure that contains the following complete command statement:

ENTER filename,format description name,,number of records

Note: See Appendix A for a description of all possible command statement parameters.

A preprinted template is available for your use (form number GX21-7638 or equivalent) that shows the DFU command key assignments. Insert the template when you are using DFU.

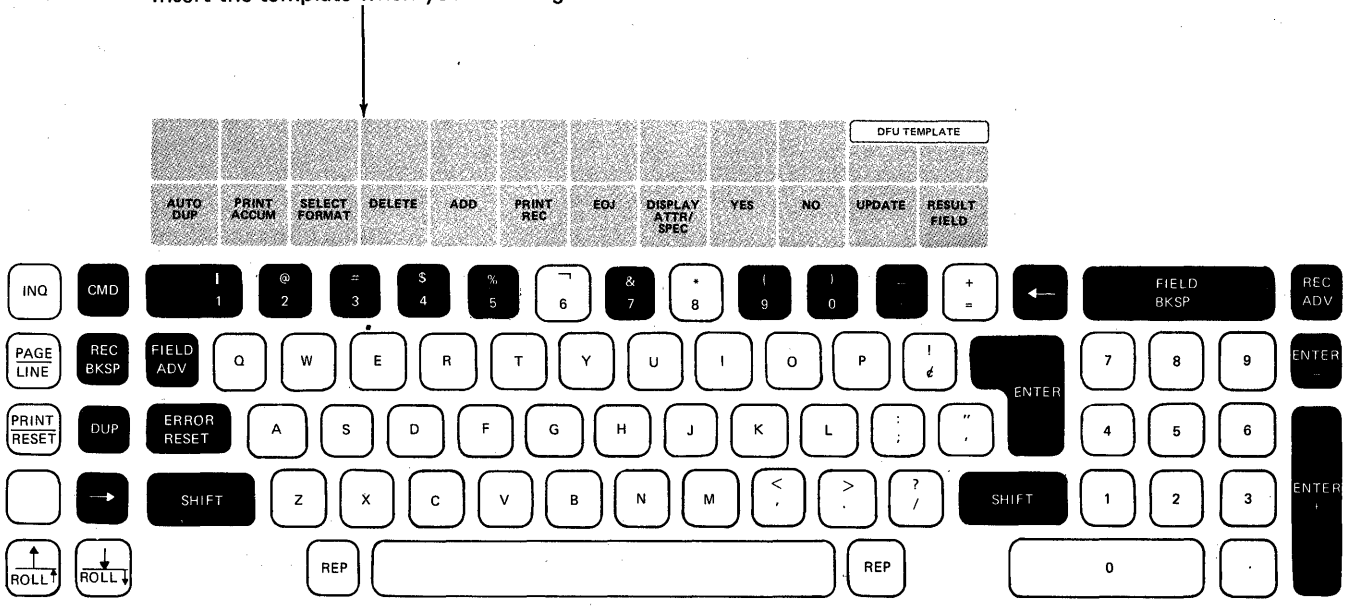


Figure 18. Function and Command Keys for Creating a Data File

FUNCTION AND COMMAND KEYS FOR CREATING A DATA FILE

Note: Function and command keys are disabled during printing.

Figure 18 shows the function and command keys that can be used when creating a data file. The following explains these keys:

Function Key	Use
◄— (cursor left)	The cursor moves left one position.
► (cursor right)	The cursor moves right one position.
REC BKSP	The current record is ignored and the cursor returns to the first field that can be entered.
REC ADV	Processing for this record is complete. All fields that have been entered are written to the file.
FIELD ADV	The field is entered (rightmost slashes are blanked out). The cursor position has no effect on what is entered. For a numeric field, the data is right-adjusted and padded with leading zeros.
FIELD BKSP	If the cursor is in the first position of a field, it is repositioned at the start of the preceding field. If the cursor is not in the first position of the field, it is reset to the start of the field. If the cursor is within the key field, it is repositioned at the start of the field. FIELD BKSP cannot be used to return to the key field once the key field has been entered. *Note:* FIELD BKSP cannot be used to access auto dup fields unless the auto dup indicator is turned off.
ENTER ENTER	Blanks the field from the cursor position to the end of the field, then enters the keyed data into the record. If the field is numeric, the data is right-adjusted and padded with leading zeros.

Function Key	Use
ENTER	Performs the same tasks as ENTER and ENTER+ except the field is edited with a minus (-) sign.
DUP (lowercase—field duplication)	The character that the cursor is positioned under and all characters to its right are duplicated from the corresponding field in the previous record, and the field is entered into the record. This key is active only if: (1) the preceding record was the same record type as the current record, or (2) the auto dup indicator is off, but the field is specified as an auto dup field and at least one record has been processed.
DUP (uppercase—single character duplication)	The character that the cursor is positioned under is duplicated from the corresponding character in the previous record, and the cursor moves to the next character position. If the cursor is currently in the last position of a field, it duplicates that character and performs a field enter function. The DUP key is active only if: (1) the preceding record was the same record type as the current record, or (2) the auto dup indicator is off, but the field is specified as an auto dup field and at least one record has been processed.
ERROR RESET	When pressed, the display stops flashing and the keyboard is freed, allowing an appropriate response to the message displayed on the screen.

Command Key	Use
AUTO DUP	Reverses the status of the auto dup indicator. If off, it is turned on; if on, it is turned off.

Command Key	Use
PRINT ACCUM	Prints the batch accumulators, adds them to the total accumulators, and resets them to zero.
SELECT FORMAT	Requests a new record type.
DELETE	Inserts a delete code into the record displayed on line 4.
ADD	Changes processing mode from update to add. New records can be entered in add mode. If DFU is generating record keys, add mode allows the entering of record keys from the keyboard (automatic key generation is suspended). Only keys lower than the next DFU-generated key can be entered. To resume automatic key generation, respond with a null entry when prompted for the next record key.
EOJ	Prompts for end of job. In enter/update mode, record counts are displayed on the display screen.
YES	Positive response to the prompt END OF JOB? Final totals are printed, record counts are printed if printing has previously occurred, and the job is terminated.
NO	Negative response to the prompt END OF JOB? Processing continues.
UPDATE	Changes processing mode to update. In update mode, only existing records can be altered.

USING DFU FEATURES WHILE CREATING A DATA FILE

The following features are explained to help you use and understand DFU while creating a data file:

Duplicating Fields

Field duplication from the previous record into the current record can be done in one of two ways:

1. Use the auto dup feature as explained in Chapters 2 and 3. Auto dup fields are specified during the job setup step, then are automatically duplicated from the previous record to the current record if the auto dup indicator is on.

   (Pressing the AUTO DUP command key reverses the status of the auto dup indicator: if off, it is turned on; if on, it is turned off.)

2. Press the DUP key. This duplication works if either of the following are true:
   a. The previous record type was the same as the current record type.
   b The previous and current fields have been defined as auto dup fields, the auto dup indicator is off, and at least one record has been processed.

   Note: The uppercase DUP key allows single character duplication from the previous record.

Checking the Record Type

After each record is entered, its record identification indicator is checked to ensure that it agrees with the record type currently being processed (defined in the RPG II specifications). This is done before the record is written to disk. If the record code does not agree, DFU examines the record code positions; if they are all blank, DFU forces the correct record code into the record and writes the record to disk (see note). If the record code positions are not all blank, the following warning message is displayed:

DFU 0019
INVALID CODE FOR RECORD ID INDICATOR

The following options are available:

- Press the REC BKSP key, then rekey the record.
- Press the REC ADV key to write the record to disk as it is.
- Press the ENTER key to force the correct code into the record and write the record to disk.

Note: If you describe the record code as a not field in the RPG II specifications, the following will occur:

- The record code forced by DFU will be a blank (hexadecimal 40) whenever a character or number (1 through 9) is specified.
- The record code forced by DFU will be a character 1 (hexadecimal F1) whenever a blank or zero is specified.

Printing the Record

A record is printed if the print option was specified in the setup step. A heading line is printed before: (1) the first record on a new page, (2) the first record of a record type, (3) the first record following the accumulator printout. Records are single spaced when printed.

Editing Fields

Editing is performed only on numeric fields. A decimal point is inserted into the edited field to show the number of decimal positions (if decimal placement is specified in the RPG II source member). A minus sign is placed to the right of the last character of a field if the field is negative. Leading zeros are suppressed.

The following is an example of editing a decimal field:

```
SALESORD  DAILY SALES ORDERS      A 01 02
ENTRY                               00020
KEY  OUR ORDER NO. CUSTOMER NO. COST PER
  20      XC4312         1313         0.00
                                     200//
```

In the preceding display, the amount 200 is keyed into the COST PER field without a decimal point. When one of the function keys is pressed to insert the field into the record, the following display will appear:

```
SALESORD  DAILY SALES ORDERS      A 01 02
ENTRY                               00020
ER NO. CUSTOMER NO. COST PER PART NUMBER
312         1313         2.00
                                    //////
```

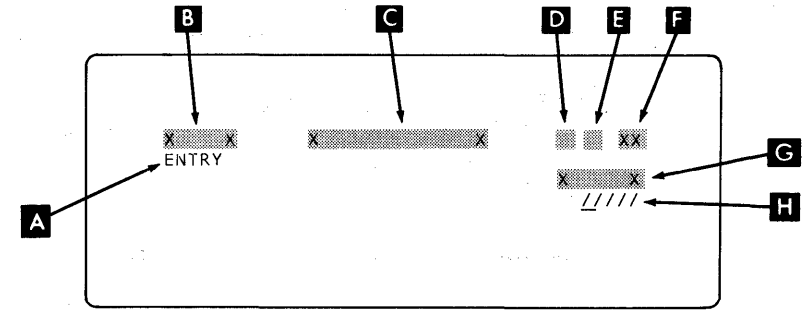
Notice the COST PER field on this display is right-adjusted, the decimal point is correctly positioned, and the field is zero filled. If the ENTER- (minus) key had been pressed in the preceding example, the following would appear:

```
SALESORD  DAILY SALES ORDERS      A 01 02
ENTRY                               00020
ER NO. CUSTOMER NO. COST PER PART NUMBER
312         1313         2.00-
                                    //////
```

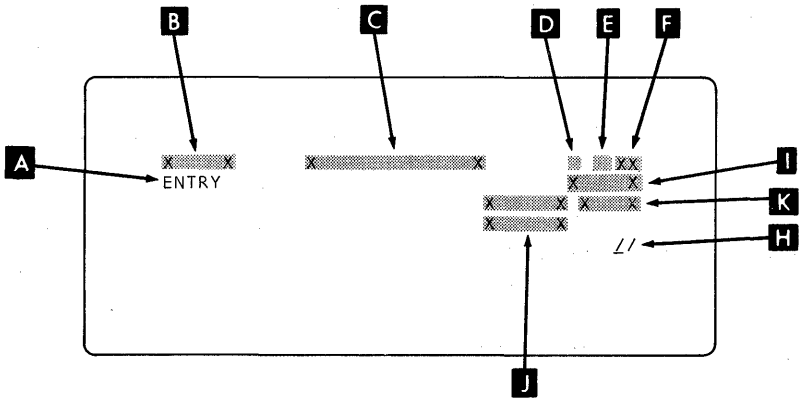
Note: Only numeric fields will be edited. If leading zeros are desired, code the fields as alphameric.

STEPS FOR ENTERING A DATA FILE

One of two initial displays appears: one if you supply the record keys, and one if DFU is generating record keys for you. Figure 19 shows the format of both displays.



You are supplying keys.



DFU is generating keys.

A The word ENTRY indicates that you will enter records in the data file.

B Name of the file you are now creating.

C Job title (may be blank).

D Automatic duplication indicator (either A or blank).

E Record type of the previous record (blank for the initial display).

F Record type of the record to be entered.

G Heading of key field.

H The cursor is positioned to accept your entry. The number of slashes (/) indicates the maximum number of positions in the field.

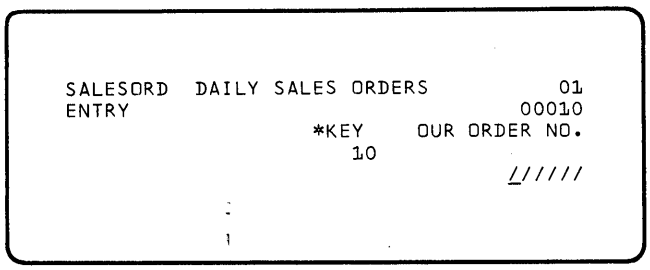
I Key of the record to be entered.

J Key and its heading for the record to be entered.

K Heading of the field to be entered.

Figure 19. Initial Displays for Creating a Data File

The following shows a sample initial display when DFU is generating the record keys:



Perform the following steps to enter your data file:

1. Check the status of the auto dup indicator. If necessary, press the AUTO DUP command key to reverse the status. (A indicates on, blank indicates off.)

2. Check that the record type on the first line of the display is the same as the selected record type being entered. If the record types are different, use the SELECT FORMAT command key to select the proper type.

3. The rightmost heading on line 3 of the display names the field to be keyed (OUR ORDER NO.). Line 5 of the display contains as many slashes (/) as there are positions in the field. Key in the data for this field. As each field is keyed, press one of the following function keys to enter the field into the record or write the record to disk:

   ENTER, ENTER+, FIELD ADV enters a field into a record. If you press any of these keys after the last defined field of a record, the record is written to disk and printed (if the print option was specified in the format description built in the setup step).

   ENTER- places a negative field in a record. If you press ENTER- after keying the last defined field of the record, the record is written to disk and printed (if print was specified in the setup).

   REC ADV writes the record to disk, prints the record (if specified), and skips the prompts for remaining fields in the record. (For example, if you have a 7-field record and only the first four fields are required, you can press the REC ADV key after keying the fourth field. This allows you to skip the prompts for the remaining fields.)

4. When the batch totals are to be printed, press the PRINT ACCUM command key.

5. a. If more records are to be entered, return to step 1.
   b. If DFU is generating record keys and you want to change (correct) records that you have already entered, press the UPDATE command key and return to step 1. This allows you to enter the key of an existing record, then update that record.
   c. If there are no more records to be entered or changed, press the EOJ command key, and the YES command key to end the program.

EXAMPLE OF CREATING A DATA FILE

This example describes how to create a data file; assume that the job was prepared for you and that the format description was built in example 1 of Chapter 3.

As an operator, you need to know the following about the job before you begin:

- The four steps of the customer order entry are:

  1. A clerk receives the customer orders and verifies that they are complete. The following shows a sample order form:

ABC Hardware Company
123 Main Street
Any City, Any State

SOLD TO:
Smith Inc.
1295 Broad St.
New York, N.Y.

SHIPPED TO:
Jones Co. (19)
5161 2nd St.
Boston, Mass.

Our Order No.	Date	Cust. Ord. No.	Customer No.	Ship Via
XC4313	10/30/7-	13019	21884	Air Freight

Quantity	Part Number	Description	Unit Price	Amount
25	412008	Elec. receptacle		
25	412009	Elec. cover plate		
600	456116	Elec. wiring - 12 gauge		

  2. The clerk totals all quantities on the order form using an adding machine, then clips the adding machine tape to the customer order.

  3. You key the customer order. DFU lists each order when it is keyed and prints the quantity totals.

  4. You compare the quantities accumulated in step 3 with the total on the adding machine tape from step 2. If these totals match, key the next customer order. If the totals do not match, correct the error before continuing.

- The name of the data file you will create is SALESORD.

- The file will be able to contain 50 records.

- The format description for this job already exists on disk with the name ORDERFMT. This format description was created in example 1 in Chapter 3.

- The data file has two record types:

  1. Record type 01 contains constant information for the order (such as customer number, customer order number, our order number, and date). This record is called the header record.

  2. Record type 02 contains transaction information about the order, including such fields as part numbers and quantities. This record is called a detail record and it requires a header record to provide necessary constant information.

- The fields that you key from the order forms are:

For Record Type 01 (header)	For Record Type 02 (detail)
OUR ORDER NO.	OUR ORDER NO.
DATE	CUSTOMER NO.
CUST ORD. NO.	QUANTITY
CUSTOMER NO.	PART NUMBER
SHIP TO	
SHIP VIA	

- OUR ORDER NO. and CUSTOMER NO. are auto dup fields. This means that you key them in the header record for the order, then set the auto dup indicator on. DFU duplicates these two fields for you when you enter each detail record from the order form.

- DFU generates the record keys.

- Figure 20 shows the three order forms you key in this example.

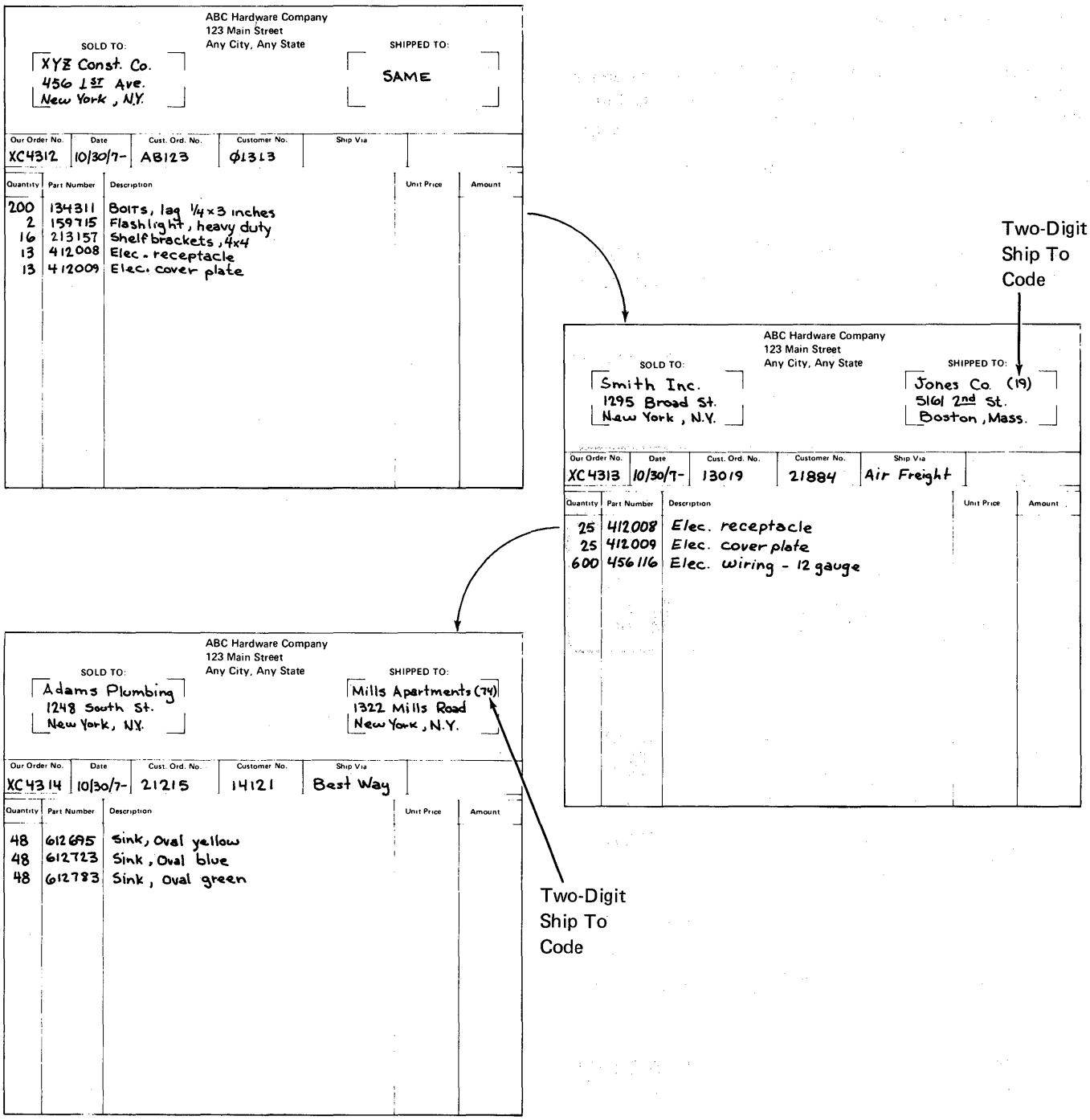


Figure 20. Sample Order Forms

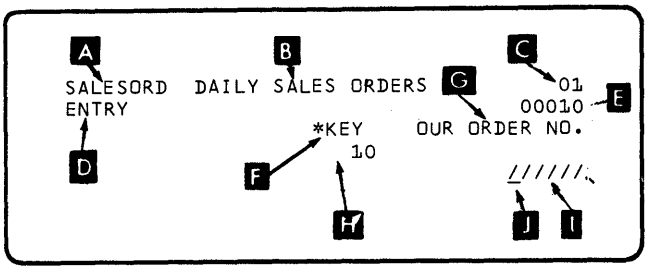
Keying the Order Forms

Begin the job by keying the following command statement (If you have just finished the setup step in example 1 in Chapter 3, the job run begins automatically and you skip this command statement.):

```
ENTER SALESORD,ORDERFMT,,50
```

Note: You can also key the word ENTER and have DFU prompt you for the necessary parameters in the command statement.

The following shows and explains the display that appears:



- The file name, SALESORD A , the job title, DAILY SALES ORDERS B , and the record type, 01 C , appear on the first line of the display.
- The word ENTRY D and the key E that DFU has generated appear on line 2.
- The key field F and order form headings G appear on line 3.
- The DFU generated key, 10 H , appears on line 4 (and line 2).
- The slashes on line 5 I indicate the field, OUR ORDER NO., that you are to key from the first order form.
- The underscore J indicates the position of the cursor.

```
SALESORD  DAILY SALES ORDERS          01
ENTRY                               00010
                  *KEY    OUR ORDER NO.
                    10
                                    //////
```

DFU has generated the first record key (10). Key the first field, XC4312, in the OUR ORDER NO. field of the header record and press ENTER.

```
SALESORD  DAILY SALES ORDERS          01
ENTRY                               00010
          *KEY   OUR ORDER NO.   DATE
            10      XC4312
                                   ////////
```

Key the date (in the form month/day/year) and press ENTER.

```
SALESORD  DAILY SALES ORDERS          01
ENTRY                               00010
  OUR ORDER NO. DATE        CUST ORD. NO.
      XC4312    10/30/7-
                                    /////
```

Key AB123 in the CUST ORD. NO. field and press ENTER.

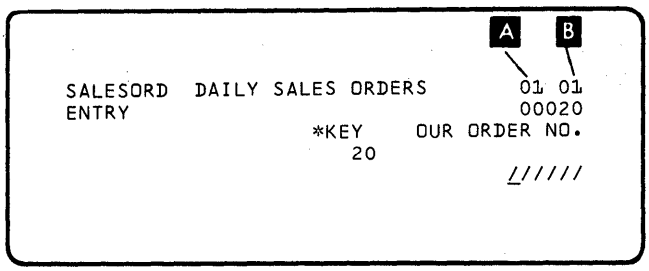
```
SALESORD  DAILY SALES ORDERS          01
ENTRY                               00010
   DATE   CUST. ORD. NO.     CUSTOMER NO.
   10/30/7-      AB123              0
                                    /////
```

Key 1313 for the CUSTOMER NO. and press ENTER.

```
SALESORD DAILY SALES ORDERS           01
ENTRY                               00010
   CUST. ORD. NO.  CUSTOMER NO.  SHIP TO
        AB123           1313
_                                   //
```

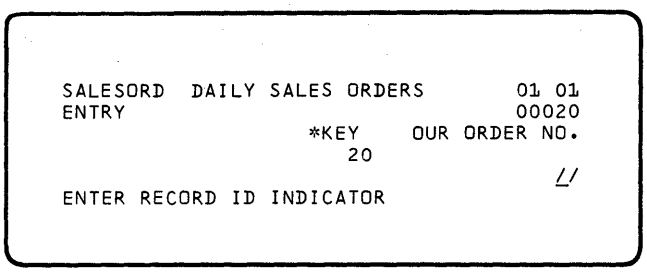
The field (SHIP TO) is ready to be keyed. Since there is no special shipping information on this order form, the customer header record is complete. Press the REC ADV key to advance to the next record. The record is printed and written to the file (SALESORD).

Note: Sample orders XC4313 and XC4314 in Figure 20 do have a 2-digit code that will be keyed into the SHIP TO field.



The display is ready to accept input for the next record. DFU has automatically incremented the record key by 10.

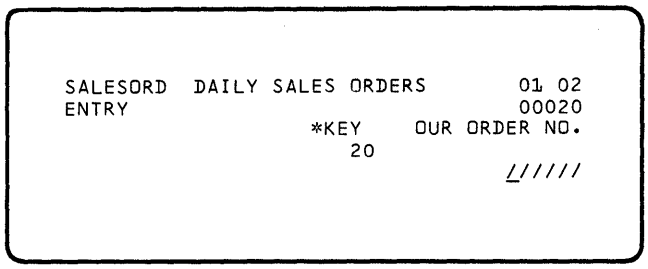
The first line indicates the previous record type was 01 A , and the current record type is 01 B .



Press the SELECT FORMAT command key to change the current record type and permit the keying of the detail record information from the order form. A prompt to

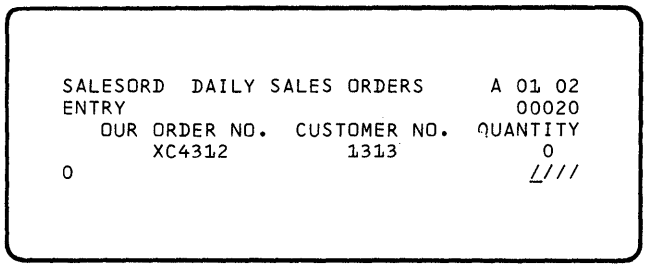
ENTER RECORD ID INDICATOR

is given. Key 2 to indicate record type 02 and press ENTER.



DFU now retrieves the format description information for record type 02. The current record type (02) appears on line 1 of the display.

Press the AUTO DUP command key to set on the auto dup indicator.



The display shows the auto dup indicator on (A appears on line 1). The fields OUR ORDER NO. and CUSTOMER NO. are automatically duplicated from the information in the record previously entered. These two fields must appear in every record in the customer order file, and the fields have the same value for each record entered from a single order form. For this reason, the two fields have been defined as auto dup fields and only need to be entered once for each order form.

Key 200 for the QUANTITY field and press ENTER. This field is a 4-position numeric field and only three numbers are keyed (200).

```
SALESORD  DAILY SALES ORDERS      A 01 02
ENTRY                               00020
NO.  CUSTOMER NO.  QUANTITY  PART NUMBER
        1313          200
                                  //////
```

Notice that QUANTITY has been right-adjusted because it is a numeric field. Key 134311, the next field (PART NUMBER). This completes the first detail record from the order form. After you press the REC ADV or ENTER key, the record is printed and written to the file (SALESORD).

```
SALESORD   DAILY SALES ORDERS     A 02 02
ENTRY                               00030
   OUR ORDER NO.  CUSTOMER NO.  QUANTITY
0      XC4312         1313           0
                                   ////
```

The display is ready to accept the next detail record. Again, DFU has incremented the record key by 10, and the current record type and previous record type are now the same. Also, two fields have been automatically duplicated from the previous record: OUR ORDER NO. and CUSTOMER NO. Thus, for each successive detail record entered from an order form, it is only necessary for you to key the two variable fields (QUANTITY and PART NUMBER).

Continue to the next page.

Refer to the first order form in Figure 20 and enter the remaining detail lines as follows:

```
SALESORD   DAILY SALES ORDERS     A 02 02
ENTRY                               00030
   OUR ORDER NO.  CUSTOMER NO.  QUANTITY
0      XC4312         1313           0
                                     2///
```

```
SALESORD   DAILY SALES ORDERS     A 02 02
ENTRY                               00030
NO. CUSTOMER NO.   QUANTITY  PART NUMBER
        1313            2
                                  159715
```

```
SALESORD   DAILY SALES ORDERS     A 02 02
ENTRY                               00040
   OUR ORDER NO.  CUSTOMER NO. QUANTITY
0      XC4312         1313           0
                                    16//
```

```
SALESORD   DAILY SALES ORDERS     A 02 02
ENTRY                               00040
NO.  CUSTOMER NO.  QUANTITY  PART NUMBER
        1313           16
                                  213157
```

```
SALESORD  DAILY SALES ORDERS      A 02 02
ENTRY                               00050
   OUR ORDER NO.  CUSTOMER NO.  QUANTITY
0      XC4312         1313           0
                                    13//
```

```
SALESORD  DAILY SALES ORDERS      A 02 02
ENTRY                               00050
NO.  CUSTOMER NO.  QUANTITY  PART NUMBER
        1313           13
                                  412008
```

```
SALESORD  DAILY SALES ORDERS      A 02 02
ENTRY                               00060
   OUR ORDER NO.  CUSTOMER NO.  QUANTITY
0      XC4312         1313           0
                                    13//
```

```
SALESORD  DAILY SALES ORDERS      A 02 02
ENTRY                               00060
NO.  CUSTOMER NO.  QUANTITY  PART NUMBER
        1313           13
                                  412009
```

This completes the entry of the first order form into the customer order file. Now press the PRINT ACCUM command key to get a batch total of the field QUANTITY for this order form. The batch total is printed after each order form is entered to simplify any error correcting that might be necessary when the totals from the order forms and the listing are compared. The batch total for each order form can aid you in quickly identifying the record(s) in error.

To enter data from the next order form, press the AUTO DUP command key to turn the auto dup indicator off. Press the SELECT FORMAT command key. Key 01 and press ENTER. Then refer to Figure 20 and enter the remaining two forms the same way you entered the first form.

After you've entered the third form, press the EOJ command key.

This display shows the prompt given when you press the EOJ command key:

```
    14 RECORDS ENTERED
     0 RECORDS UPDATED
     0 RECORDS DELETED

END OF JOB?
```

Press the YES command key; the batch and total accumulators are printed and record counts are printed if printing has previously occurred. The customer order file is saved on the disk so it can be used later by another program.

You have now entered all daily orders into the system. Figure 21 is the printed listing produced by DFU as a result of entering the daily orders.

The title is printed at the top of each page. The headings are printed above the data and are printed each time the record type changes and each time following an accumulator printout. The key field is always printed first followed by all fields defined for the record type.

```
                                DAILY SALES ORDERS                              PAGE 001

*KEY   OUR ORDER NO.  DATE       CUST. ORD. NO.  CUSTOMER NO.  SHIP TO  SHIP VIA

  10       XC4312    10/30/7-        AB123          1313

*KEY   OUR ORDER NO.  CUSTOMER NO.  QUANTITY  PART NUMBER

  20       XC4312         1313         200       134311
  30       XC4312         1313           2       159715
  40       XC4312         1313          16       213157
  50       XC4312         1313          13       412008
  60       XC4312         1313          13       412009

                                              QUANTITY

BATCH ACCUMULATORS                                244

*KEY   OUR ORDER NO.  DATE       CUST. ORD. NO.  CUSTOMER NO.  SHIP TO  SHIP VIA

  70       XC4313    10/30/7-       13019          21884         19     AIR FREIGHT
```

```
                                DAILY SALES ORDERS                              PAGE 002

*KEY   OUR ORDER NO.  CUSTOMER NO.  QUANTITY  PART NUMBER

  80       XC4313         21884          25       412008
  90       XC4313         21884          25       412009
 100       XC4313         21884         600       456116

                                              QUANTITY

BATCH ACCUMULATORS                                650

*KEY   OUR ORDER NO.  DATE       CUST. ORD. NO.  CUSTOMER NO.  SHIP TO  SHIP VIA

 110       XC4314    10/30/7-       21215          14121         74     EEST WAY

*KEY   OUR ORDER NO.  CUSTOMER NO.  QUANTITY  PART NUMBER

 120       XC4314         14121          48       612695
 130       XC4314         14121          48       612723
 140       XC4314         14121          48       612783
```

```
                                DAILY SALES ORDERS                              PAGE 003

                                              QUANTITY

BATCH ACCUMULATORS                                144

TOTAL ACCUMULATORS                               1038

    14 RECORDS ENTERED          0 RECORDS UPDATED          0 RECORDS DELETED
```

Figure 21. Listing Produced from ENTER Example

This chapter explains how to update a data file. It is assumed that the format description already exists. Refer to Appendix F for a sample form to use when maintaining a data file.

If the format description does not exist, refer to Figure 6 in Chapter 3 for assistance in creating it before proceeding.

UPDATE COMMAND STATEMENT

Note: If you have just completed the setup step, you do not need to enter the UPDATE command statement. Refer directly to *Steps for Maintaining a Data File* in this chapter.

When updating an existing file, key the word UPDATE and press the ENTER key. A prompt is issued for the name of the file:

ENTER FILENAME OF FILE TO BE MAINTAINED

Your response must be the name of an existing file.

The next prompt is:

ENTER NAME OF FORMAT DESCRIPTION
(THE DEFAULT NAME IS #DFUOBJ)

Your response specifies the name of the format description for processing this job.

The preceding prompting sequence can be skipped by initially keying the complete command statement or the name of the procedure that contains the following complete command statement:

UPDATE filename,format description name

FUNCTION AND COMMAND KEYS FOR MAINTAINING A DATA FILE

Note: Function and command keys are disabled during printing.

Figure 22 shows function and command keys that can be used when maintaining a data file. The following explains these keys:

Function Key	Use
←	The cursor moves left one position.
→	The cursor moves right one position.
REC BKSP	The current record is ignored and the cursor returns to the first field that can be updated.
REC ADV	Processing for this record is complete. All fields that have been updated are written to the file.
FIELD ADV	The field is entered as displayed. The cursor position has no effect on what is entered.

A preprinted template (GX21-7638 or equivalent) is available that shows the DFU command key assignments. Insert the template when you are using DFU.

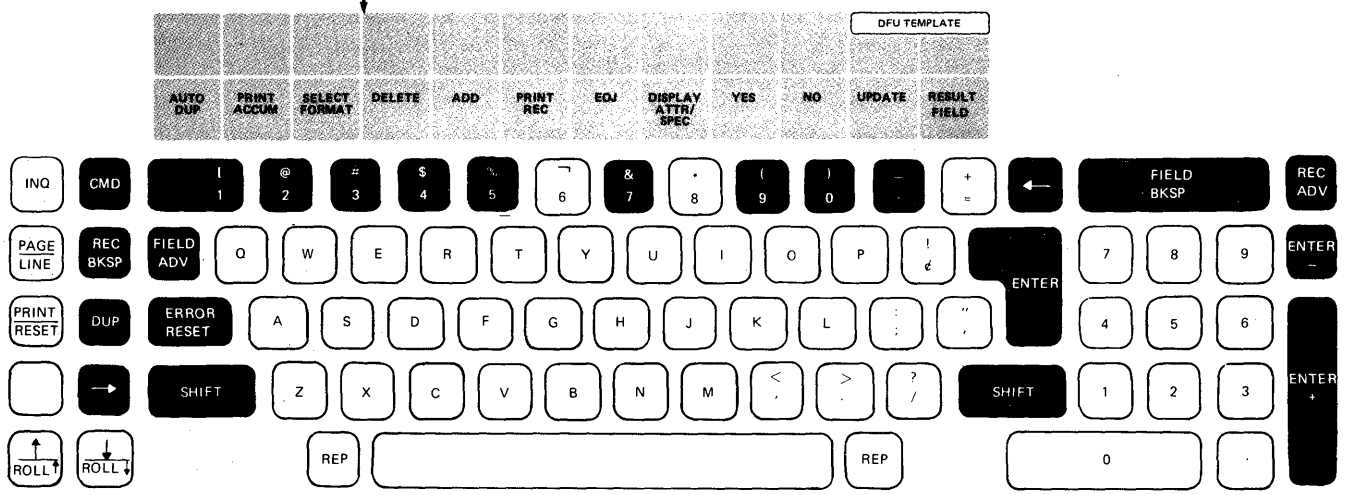
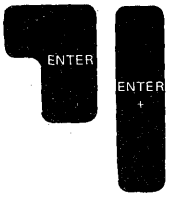


Figure 22. Function and Command Keys for Maintaining a Data File

FIELD BKSP

If the cursor is in the first position of a field, it is repositioned at the start of the preceding field. If the cursor is not in the first position of the field, it is reset to the start of the field. If the cursor is within the key field, it is repositioned at the start of the field. FIELD BKSP cannot be used to return to the key field once the key field has been entered.

Note: FIELD BKSP cannot be used to access auto dup fields unless the auto dup indicator is turned off.



Blanks the field from the cursor position to the end of the field, then enters the keyed data into the record. If the field is numeric, the data is right-adjusted and padded with leading zeros.

Performs the same tasks as ENTER and ENTER+ except the field is edited with a minus (-) sign.

(lower case—field duplication)

The character that the cursor is positioned under and all characters to its right are duplicated from the corresponding field in the record currently being processed, and the field is entered into the record.

(upper case—single character duplication)

The character that the cursor is positioned under is duplicated from the corresponding character in the record currently being processed, and the cursor moves to the next character position. If the cursor is currently in the last position of a field, it duplicates that character and performs a field enter function.

When pressed, the display stops flashing and the keyboard is freed, allowing an appropriate response to the message displayed on the screen.

Command Key	Use
AUTO DUP 	Reverses the status of the auto dup indicator. If off, it is turned on; if on, it is turned off.
PRINT ACCUM 	Prints the batch accumulators, adds them to the total accumulators, and resets them to zero.
SELECT FORMAT 	Requests a new record type.
DELETE 	Inserts a delete code into the record displayed on line 4.
ADD 	Changes processing mode from update to add. New records can be entered in add mode. If DFU is generating record keys, add mode allows the entering of record keys from the keyboard (automatic key generation is suspended). Only keys lower than the next DFU-generated key can be entered. To resume automatic key generation, respond with a null entry when prompted for the next record key.
EOJ 	Prompts for end of job. Record counts are displayed on the display screen.
YES 	Positive response to the prompt END OF JOB? Final totals are printed, record counts are printed if printing has previously occurred, and the job is terminated.
NO 	Negative response to the prompt END OF JOB? Processing continues.
UPDATE 	Changes processing mode to update. In update mode, only existing records can be altered.

USING DFU FEATURES WHILE MAINTAINING A DATA FILE

Duplicating Fields

It is possible to duplicate field data while updating a record. The data, however, is not duplicated from the previous record; it is duplicated from the record currently being processed (prior to any updating). This duplication can be performed automatically, or can be requested by pressing the DUP function key.

Automatic duplication occurs when the following two conditions are met: (1) the field was defined at setup time as an auto dup field, and (2) the auto dup indicator is currently on. Fields that meet these conditions remain unchanged in the record. You must press the AUTO DUP command key to turn off the auto dup indicator before you are allowed to update these fields.

You can also do field duplication by pressing the DUP key (lower or upper case). The lowercase DUP key performs a field duplication. The data from the cursor position to the end of the current field being processed is replaced by the corresponding data from the record (prior to any updating); the data is then written in the record. The uppercase DUP key performs a single character duplication. The character the cursor is positioned under is replaced by the corresponding character from the record (prior to any updating) and the cursor moves to the next character.

Note: If the cursor is currently at the last position in a field, the character will be duplicated and the data field will be written in the record.

Deleting Records

Unwanted records can be marked for deletion from a data file by pressing the DELETE key. DFU inserts a delete code into the record, but the record physically remains in the file. The specified delete character is put in the proper position in the record and the message RECORD DELETED is printed. For example:

```
            Delete Character
                 /
00020   X   XC4312   01313   0200   134311

RECORD DELETED
```

If the delete character is taken out of a deleted record, the record acquires a restored status. That is, it is included again as an active record in the file and can be maintained as such.

To remove records marked for deletion from the data file, use the ORGANIZE procedure as explained in Part 2, *Procedures,* of the *System/32 System Control Programming Reference Manual,* GC21-7593. File reorganization should be done periodically (for example, after each run or weekly) to reduce the amount of unnecessary data retained on the disk.

Editing Fields

When a record is being updated, line 4 contains the edited field and line 5 contains the unedited field as it currently exists. If a negative field is currently being updated, the rightmost character in the unedited field is displayed as an alphabetic character; the alphabetic character represents a negative field on the disk record.

For example, if a file contains the balance owed by each customer, a negative amount in the balance owed field indicates credit. If customer 01313 purchases an item, his balance must be updated:

```
CUSTBAL          CUSTOMER BALANCE          01
UPDATE                                  01313
        CUSTOMER NUMBER   CUSTOMER BALANCE
             1313                 12.53-
                                  00125L
```

The display shows the balance for balance for customer 1313. Note the L in the unedited display represents a minus 3. If the item purchased costs $1.51, the credit balance must be adjusted so it is only $11.02.

```
CUSTBAL          CUSTOMER BALANCE        01
UPDATE                                 01313
        CUSTOMER NUMBER   CUSTOMER BALANCE
             1313                 12.53-
                                  11025L
```

Key 1102 over the first four characters of the unedited field on line 5. Note that the cursor is positioned under character 5. Since the balance is still negative, press the ENTER- key to update the field (the last two characters are blanked and the field is right-adjusted).

```
CUSTBAL          CUSTOMER BALANCE        01
UPDATE                                 01313
        CUSTOMER NUMBER   CUSTOMER BALANCE
             1313                 11.02-
                                  00110K
```

The next time you update this customer record, it will appear on the display as indicated. Note the K in the unedited display represents a minus 2.

The representations for the negative numbers 0 through 9 are as follows:

Number	Represented By	Number	Represented By
1	J	6	O
2	K	7	P
3	L	8	Q
4	M	9	R
5	N	0	&

Printing the Record

A record is printed if:

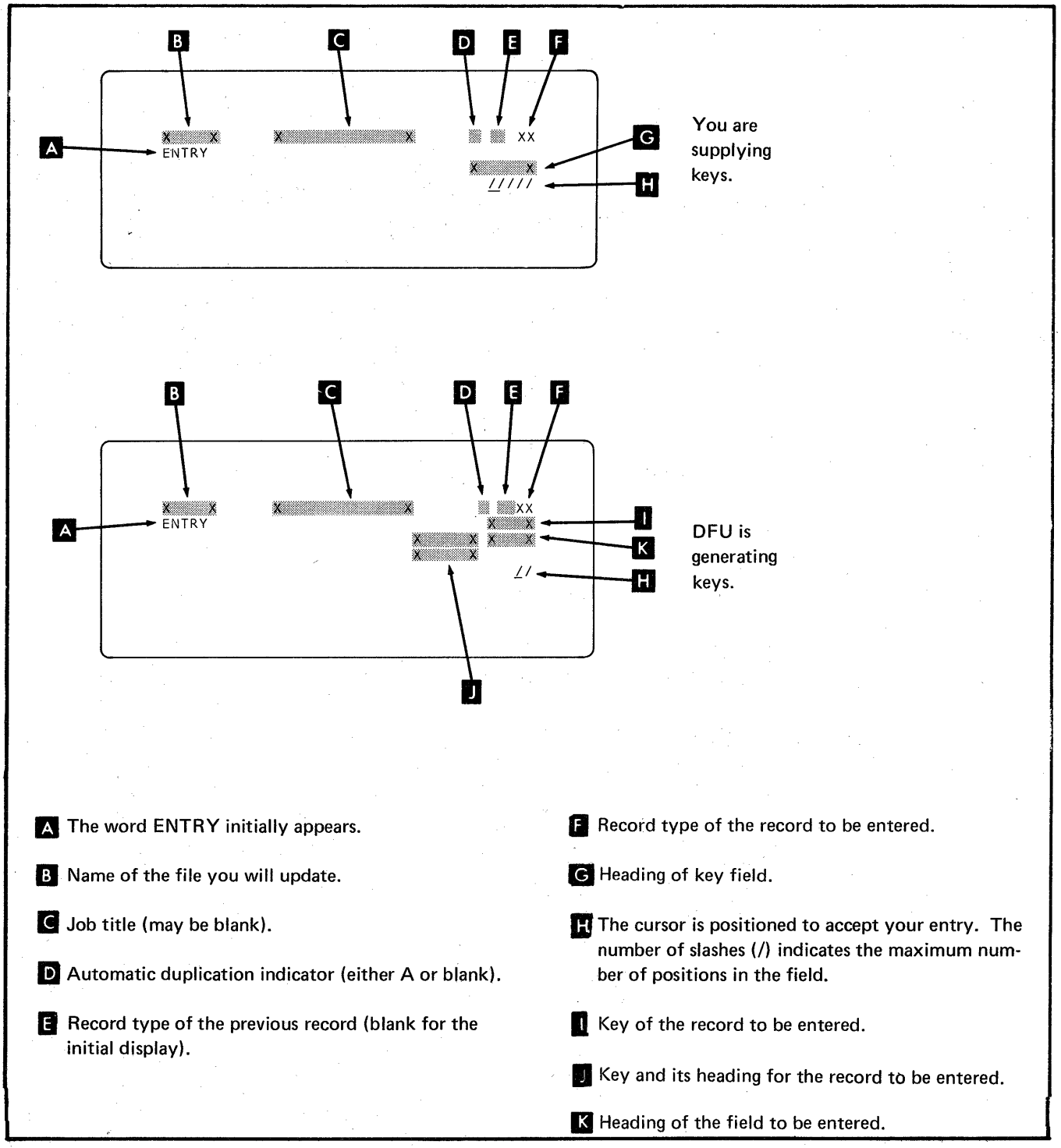
1. You changed the record.
2. You entered a new record and the print option was specified in the job setup step.
3. You pressed the DELETE command key to mark a record for deletion.
4. You removed the delete code from a record.

Two single-spaced data lines are printed for each updated record. The first line contains all defined fields in the old copy of the record. The second line is in the same format but contains only the updated fields for that record. Added records are printed on a single line. A deleted record is printed, and a message (RECORD DELETED) is printed on the next line.

Note: While one record is printing, you may key in data for the next record. If a printer hardware error occurs during the printing, it will not be detected until printing begins on the new record and the error message will be displayed at that time. A 0 or 1 option to the message causes the last printed line to be reprinted before the new updated record is printed.

STEPS FOR MAINTAINING A DATA FILE

One of two initial displays appears after you key the UPDATE command statement: one if you supply the record keys and one if DFU generates record keys for you. These are the same displays that can initially appear when you are creating a data file. Figure 23 shows the format of both displays.



A The word ENTRY initially appears.

B Name of the file you will update.

C Job title (may be blank).

D Automatic duplication indicator (either A or blank).

E Record type of the previous record (blank for the initial display).

F Record type of the record to be entered.

G Heading of key field.

H The cursor is positioned to accept your entry. The number of slashes (/) indicates the maximum number of positions in the field.

I Key of the record to be entered.

J Key and its heading for the record to be entered.

K Heading of the field to be entered.

Figure 23. Initial Displays for Maintaining a Data File

If DFU is generating keys for your job and you want to stop automatic key generation, use either the ADD or the UPDATE command keys as described below. To resume automatic key generation from entry mode, respond with a null entry when prompted for the next record key. If in update mode, return to entry mode by pressing the ADD command key and then respond with a null entry when prompted for the next record key.

The following describes how you add, delete, and update records:

- If you want to add a record to the file:

  1. Press the ADD command key.

  2. Enter a record key that does not already exist. If DFU was automatically generating keys, you must enter a key less than the next one DFU will generate.

  3. Check the state of the auto dup indicator. If necessary, press the AUTO DUP command key to reverse the state.

  4. Check that the record type on the first line of the display is the same as the desired record type. If not, use the SELECT FORMAT command key to select the proper type.

  5. Key data in the defined fields as the prompts appear and enter the new record into the file.

- If you want to delete a record from the file:

  1. Press the UPDATE command key.

  2. Enter the key of the record you want to delete. If a record is displayed that you do not want to delete, press REC BKSP, then enter the key of the record you want.

  3. Press the DELETE command key. DFU marks the record for deletion and prints a RECORD DELETED message along with the record.

- If you want to update a record in the file:

  1. Press the UPDATE command key.

  2. Enter the key of the record you want to update. If a record is displayed that you do not want to update, press REC BKSP; then enter the key of the record you want.

  3. Check that the record type on the first line of the display is the same as the desired record type. If not, use the SELECT FORMAT command key to select the proper type.

  4. To update a field in the record, press the FIELD ADV or FIELD_BKSP key until the field is displayed. Update the field and press the ENTER, ENTER+, ENTER-, or FIELD ADV key to record the change. When all updates to the record have been made, press the REC ADV key. The record is written to the disk file and printed.

- When there are no more records to add, delete, or update, press the EOJ command key.

EXAMPLE OF MAINTAINING A DATA FILE

To demonstrate how you can use DFU to update an existing data file, we will use the customer order file (SALESORD) created in Chapter 4, and the format description (ORDERFMT) created in Example 1 in Chapter 3. Assume that a customer, the XYZ Construction Company, called late in the day, after all the orders had been entered into the customer order file. This customer wanted to change the order as shown in Figure 24. Note that the customer has changed a quantity, deleted an item, and added another item. These changes were all marked on the original order form.

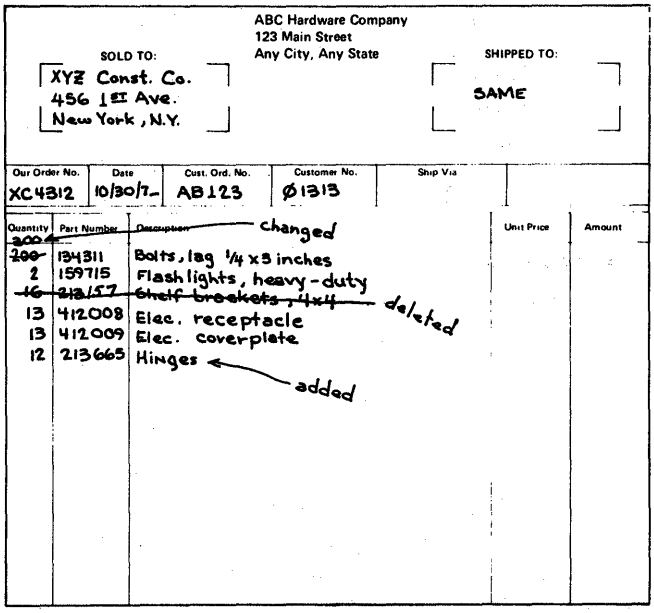


Figure 24. Sample Order Form

The following displays show you how to update the customer order file with the information shown in Figure 24. To begin the update process, key the following command statement:

```
UPDATE SALESORD,ORDERFMT
```

and press ENTER (or key the word UPDATE and respond to the prompts).

```
SALESORD  DAILY SALES ORDERS          01
ENTRY                              00150
                    *KEY    OUR ORDER NO.
                      150
                                  /////
```

This display shows that DFU has generated the key for the next record to be entered in the customer order file. Figure 21 shows the last record entered had a key of 00140, so DFU has automatically provided the next key of 00150. But in this example, records are not to be added to the end of the file. Existing records are to be updated; so press the UPDATE command key to suspend the automatic key generation by DFU.

```
SALESORD  DAILY SALES ORDERS          01
ENTRY                              00150
                                    *KEY
                                       0
                                  /////
```

Key 20, the key of the first record to update and press ENTER.

Note: You can enter the record key as either 00020 or 20. If you key 20, the system automatically right-justifies the entry, adds the leading zeros, and retrieves the record with a key of 00020. However, the leading zeros will be suppressed.

```
SALESORD  DAILY SALES ORDERS       02 02
UPDATE                             00020
                    *KEY    OUR ORDER NO.
                       20   XC4312
                                  XC4312
```

The display shows the record retrieved by DFU.

```
SALESORD  DAILY SALES ORDERS       02 02
UPDATE                             00020
   OUR ORDER NO.  CUSTOMER NO.  QUANTITY
0      XC4312         1313         200
                                    0000
```

Press the FIELD ADV key twice to skip over the fields that do not need to be updated. The field to be updated (QUANTITY) has been positioned on line 5. Key the new value of 300 and press ENTER. The field appears to have a value of 3000 but the cursor is positioned under the last digit.

```
SALESORD  DAILY SALES ORDERS       02 02
UPDATE                             00020
NO.  CUSTOMER NO.  QUANTITY  PART NUMBER
        1313          300      134311
                                  134311
```

When you press the ENTER key, the field is blanked from the cursor to the end of the field so the actual value is 300. Press the REC ADV key. The updated record is written to the customer order file and printed.

```
SALESORD  DAILY SALES ORDERS       02 02
UPDATE                             00020
                                    *KEY
                                       0
                                  /////
```

Key 40, the key of the next record to be updated and press ENTER (as shown in Figure 24, you must delete this record).

```
SALESORD  DAILY SALES ORDERS        02 02
UPDATE                              00040
                    *KEY    OUR ORDER NO.
                      40       XC4312
                                XC4312
```

Press the FIELD ADV key three times to display the field PART NUMBER. Check that you are deleting the correct record.

```
SALESORD  DAILY SALES ORDERS        02 02
UPDATE                              00040
NO.  CUSTOMER NO.  QUANTITY  PART NUMBER
        1313          16       213715
                                213157
```

Press the DELETE command key. This places a delete code in the record. The record is printed and rewritten in the customer order file with the proper delete code inserted.

```
SALESORD  DAILY SALES ORDERS        02 02
UPDATE                              00040
                                    *KEY
                                      0
                                    /////
```

Now you must add a new detail record after 00060. Press the ADD command key. Key a value of 65 and press ENTER. This record key tells DFU a new record is to be inserted into the customer order file between record 00060 and record 00070.

```
SALESORD  DAILY SALES ORDERS        02 02
ENTRY                               00065
                    *KEY    OUR ORDER NO.
                      65
                                  //////
```

The first field (OUR ORDER NO.) is prompted for. Notice the function being used now is enter, since a new record is being added to the file. The next two fields (OUR ORDER NO. and CUSTOMER NO.) have the same values as the last record processed (00040), so press the DUP key twice to duplicate these fields from the previous record.

```
SALESORD  DAILY SALES ORDERS        02 02
ENTRY                               00065
   OUR ORDER NO.  CUSTOMER NO.  QUANTITY
5      XC4312          1313          0
                                    ////
```

The two fields have been duplicated. Key 12, the new QUANTITY, and press the ENTER key.

```
SALESORD  DAILY SALES ORDERS        02 02
ENTRY                               00065
NO.  CUSTOMER NO.  QUANTITY  PART NUMBER
        1313          12
                                  //////
```

Key 213665, the new PART NUMBER and press ENTER. The record is printed and entered into the file.

The customer order file update is complete; press the EOJ command key and then the YES command key to end the job. Figure 25 is the printed listing produced by DFU as a result of updating the file.

Record 00020 is printed as it appeared before the updates were made; each field that you actually updated in record 00020 is printed immediately under its original value. This audit trail allows you to determine any changes you made to the files and also shows what values the records had before you updated them.

Record 00040 is printed, followed by a message (RECORD DELETED) that indicates this record has been marked for deletion.

The added record 00065 is printed in the same manner as shown in Figure 25.

The accumulators reflect the net value of the accumulated field (QUANTITY) as follows:

1.	Updated record:	subtract old value	-200
		add new value	+300
			100
2.	Deleted record:	subtract old value	- 16
			84
3.	Added record:	add new value	+ 12
			96

```
76/02/06                          DAILY SALES ORDERS                    PAGE 001

*KEY   OUR ORDER NO.   CUSTOMER NO.   QUANTITY   PART NUMBER

  20       XC4312          1313          200        134311
                                         300
  40       XC4312          1313           16        213157
RECORD DELETED
  65       XC4312          1313           12        213565

                                                   QUANTITY

BATCH ACCUMULATORS                                    96

TOTAL ACCUMULATORS                                    96

     1 RECORDS ENTERED                 1 RECORDS UPDATED                 1 RECORDS DELETED
```

Figure 25. Listing Produced from UPDATE Example

This chapter explains how to display records from a data file and, if necessary, print those records. It is assumed that the format description already exists. Refer to Appendix F for a sample form to use when displaying or printing records from a data file.

If the format description does not exist, refer to Figure 7 in Chapter 3 for assistance in creating it before proceeding.

INQUIRY COMMAND STATEMENT

Note: If you have just completed the setup step, you do not need to enter the INQUIRY command statement. Refer directly to *Displaying the Records* in this chapter.

To display records from a file, key the word INQUIRY and then press the ENTER key. A prompt is issued for the name of the file:

ENTER FILENAME OF INQUIRY FILE

Your response must be the name of an existing file.

The next prompt that appears is:

ENTER NAME OF FORMAT DESCRIPTION
(THE DEFAULT IS #DFUOBJ).

Your response specifies the name of the format description for processing this job.

The preceding prompting sequence can be skipped by initially keying the complete command statement or the name of the procedure that contains the following complete command statement:

INQUIRY filename,format description name

A preprinted template is available for your use (form number GX21-7638 or equivalent) that shows the DFU command key assignments. Insert the template when you are using DFU.

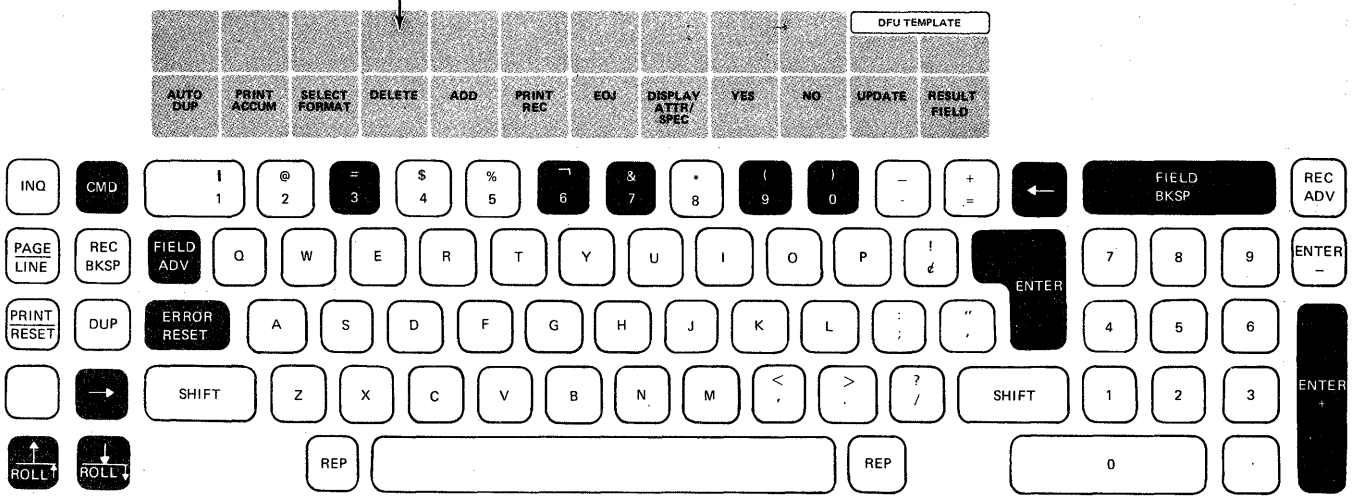


Figure 26. Function and Command Keys for Displaying Records

FUNCTION AND COMMAND KEYS FOR DISPLAYING RECORDS

Figure 26 shows the function and command keys that can be used when displaying a data file. The following explains these keys:

Function Key	Use
← (cursor left)	The cursor moves left one position.
→ (cursor right)	The cursor moves right one position.
ENTER / ENTER	Enters the key of a record that you want to display.
ROLL↑	Causes the next record in the file to be retrieved and displayed on line 4.
ROLL↓	Causes the preceding record to be retrieved and displayed on line 4.
ERROR RESET	When pressed, the display stops flashing and the keyboard is freed, allowing an appropriate response to the message displayed on the screen.
FIELD ADV	Displays the next field, shifting the record being displayed left one field.
FIELD BKSP	Displays the previous field, shifting the record being displayed right one field.

Command Key	Use
SELECT FORMAT	Requests a new record type.
= 3	
PRINT REC	Prints the record currently shown on the display screen.
¬ 6	
EOJ	Prompts for end of job.
& 7	
YES	Causes a positive response to be given to the prompt END OF JOB? The job is terminated.
(9	
NO	Causes a negative response to be given to the prompt END OF JOB? Processing continues.
) 0	

DISPLAYING THE RECORDS

After you key the INQUIRY command statement, the first record in the file is displayed on the screen. If the first record in the file does not match any of the record types defined in the format description, the following message is displayed:

UNABLE TO IDENTIFY FIRST RECORD IN FILE

Respond to this message by doing one of the following:

1. Enter the key of an existing record.
2. Press the ROLL↑ key.
3. Press the SELECT FORMAT command key to specify the format in which to display the record.

Once the first record is displayed, the succeeding records can be retrieved in two ways:

1. Pressing the ROLL↑ function key.
2. Keying a record key and pressing ENTER.

   Note: If all the keys in the file are numeric, you need enter only the rightmost digits of a record key. The field is automatically right-justified and leading zeros are inserted prior to retrieving the record.

If you press the ROLL↑ key, the next record in the file is displayed on line 4 of the display. You can scan the entire file by pressing the ROLL↑ key for each succeeding record.

Record Type Determination

The record type is determined automatically by checking the record identification code. If the record type cannot be determined, a message is displayed on the screen. You can display the record by pressing the SELECT FORMAT key, keying the record identification indicator, and pressing the ENTER key. To ignore the record, enter another record key or press ROLL↑ or ROLL↓ to retrieve the next or previous record.

To force the display of a different record type for any record, press the SELECT FORMAT key, key the record identification indicator, and press the ENTER key.

PRINTING RECORDS

To print the record being displayed, press the PRINT REC command key. All records printed are single spaced.

EXAMPLE OF DISPLAYING RECORDS

The following example demonstrates how DFU can be used for inquiry into a data file for up-to-date information on inventory status.

For this example, the inquiry function is used to inquire into the item master file in order to supply information to a waiting customer. Typically, a customer is interested in two things:

1. Is the item in stock?

2. What is the price?

The following assumptions are made for this example:

- The item master file is named ITEMAST.
- The format description already exists under the name ITMASTF1.
- Each inventory record in the item master file is identified by an item number that is the record key field.
- The fields to be displayed from the inventory record are:

  ITEM NUMBER (this is the record key)
  DESCRIPTION
  UNIT PRICE
  QTY-ON-HAND

Assume that a customer calls on the telephone and wants to place a rush order for the following electrical equipment:

1. Four-outlet electrical receptacles, item number 412000

2. Four-outlet cover plates, item number 412009

The customer wants 25 units of each item. The customer needs these items immediately and wants to confirm that the quantity is on hand before placing the order. Therefore, the inquiry function is used to see if the desired items are on hand.

Inquiry into the item master file begins by you keying the following command statement:

INQUIRY ITEMAST,ITMASTF1

(or by keying the word INQUIRY and responding to the prompts).

```
ITEMAST      ITEM MASTER FILE          01
INQUIRY                                000104
        ITEM NUMBER    DESCRIPTION
              104      COTTER PINS, 1 IN.
                                      /////
```

The display shows the first record in the item master file. The inquiry function automatically retrieves the first record in the file and displays that record. The display for inquiry is similar to that for enter or update. The word INQUIRY appears on line 2. Line 5 contains slashes (/) indicating the length of the record key (ITEM NUMBER).

Note: If DFU is unable to determine the record type of the first record in the file, an informative message appears (UNABLE TO IDENTIFY FIRST RECORD IN FILE).

```
ITEMAST      ITEM MASTER FILE          01
INQUIRY                                000104
        ITEM NUMBER    DESCRIPTION
              104      COTTER PINS, 1 IN.
                                      /////
```

Enter 412000, the key of the first record to be displayed. Press the ENTER key.

```
ITEMAST        ITEM MASTER FILE          01
INQUIRY                                412000
       ITEM NUMBER  DESCRIPTION
          412000    ELEC RECEPT,4-OUTLET
                                       //////
```

The requested item (412000) is then displayed. Verify the item by visually checking the description. Display other fields in the record by pressing the FIELD ADV key.

```
ITEMAST      ITEM MASTER FILE            01
INQUIRY                                412000
NUMBER  DESCRIPTION             UNIT PRICE
2000    ELEC RECEPT,4-OUTLET        1.15
                                    //////
                                    _
```

The record is moved left on the display to the next field defined by the format description. The display shows the next field in the record (UNIT PRICE).

```
ITEMAST      ITEM MASTER FILE            01
INQUIRY                                412000
IPTION            UNIT PRICE  QTY-ON-HAND
RECEPT,4-OUTLET         1.15         260
                                    //////
                                    _
```

Press the FIELD ADV key again to display the QTY-ON-HAND. The display shows a QTY-ON-HAND of 260 units.

The next item record is to be displayed. You can do this by entering the key in the same way as the previous record or by successively displaying the following records in the file. In this case, roll through the file and visually search for the next record. Press the ROLL↑ key until the item in question is located.

The following series of displays shows how you use the ROLL↑ key to locate the item in question.

```
ITEMAST        ITEM MASTER FILE          01
INQUIRY                                412001
       ITEM NUMBER  DESCRIPTION
          412001    ELEC SWITCH, TOGGLE
                                       //////
```

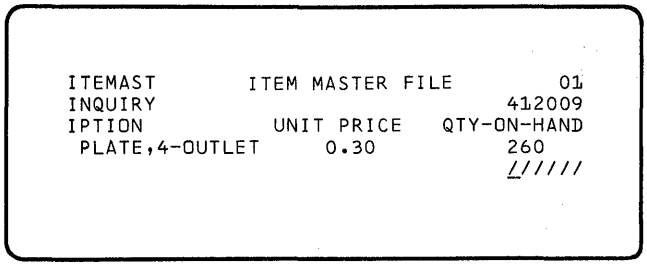
```
ITEMAST      ITEM MASTER FILE            01
INQUIRY                                412002
       ITEM NUMBER  DESCRIPTION
          412002    ELEC SWITCH, MERCURY
                                       //////
```

```
ITEMAST      ITEM MASTER FILE            01
INQUIRY                                412006
       ITEM NUMBER  DESCRIPTION
          412006    ELEC RECEPT,2-OUTLET
                                       //////
```

```
ITEMAST        ITEM MASTER FILE          01
INQUIRY                                412007
       ITEM NUMBER  DESCRIPTION
          412007    COVER PLATE,2-OUTLET
                                       //////
```

```
ITEMAST        ITEM MASTER FILE          01
INQUIRY                                412009
       ITEM NUMBER  DESCRIPTION
          412009    COVER PLATE,4-OUTLET
                                       //////
```

The display shows item 412009 has been located. Verify both the item number and description. Now press the FIELD ADV key twice to display the price and quantity.



This display shows that the quantity on hand is sufficient to fill the order. You can tell the customer that both items can be supplied immediately. Press the EOJ command key. When the

END OF JOB?

prompt appears, press the YES command key to end the job.

The preceding example shows the inquiry function being used to display the information a customer would be interested in from an item master file.

A number of different format descriptions, under different names, could be used to display other information from the item master file. For example, a format description, named ITMASTF2, might display all of the central and branch warehouse inventory information, or a different format description could exist for each warehouse to display status information for the particular warehouse location.

This chapter explains how to key the command statement to print reports from data file information. It is assumed that the format description exists. If the format description does not exist, refer to Figure 8 in Chapter 3 for assistance in creating the format description before proceeding. Refer to Appendix F for a sample form to use when printing reports from data file information.

KEYING THE LIST COMMAND STATEMENT

There are two ways to key the LIST command statement. The first is:

1. Key the word LIST. The following prompt appears:
   ENTER FILENAME OF LIST FILE

2. Key the name of the list file from which the report is printed. The following prompt appears:
   ENTER NAME OF FORMAT DESCRIPTION
   (THE DEFAULT NAME IS #DFUOBJ)

3. Key the name of the format description for the job. The following prompt appears:
   INDICATE IF THE DATA IS TO BE SORTED
   BEFORE LISTING ('NOSORT',SORT)

4. Key NOSORT (or press the ENTER key) if you do not want DFU to sort the file before printing it. Key SORT if you want DFU to sort the file before printing it.

   Note: If NOSORT was specified in the format description, DFU will not accept a response of SORT. If SORT was specified in the format description and your response is NOSORT, DFU offers you the option of ending or continuing the job. If you continue, DFU lists the file without sorting it.

   After you respond to this prompt, DFU prints the report.

The second way of keying the LIST command statement is by keying the following complete command statement or the name of the procedure that contains the complete command statement:

LIST filename,format description name,, [SORT / NOSORT]

How to Key the LIST Command Statement When You Have a Related Master File

The methods of keying the LIST command statement are modified as follows when you use a related master file:

For the first method of keying the command statement, step 1 is:

Key LIST ,,,,,,,master file name

The following prompt appears:
ENTER FILENAME OF LIST FILE

Steps 2, 3, and 4 remain the same.

For the second method, the complete command statement is:

LIST filename, format description name,,

[SORT / NOSORT] ,,,,master file name

Chapter 8. RPG II Source Member

An RPG II source member consists of:
a. A file description specification.
b. Input specifications
It is required for the DFU setup step to describe the file to be processed. DFU uses the RPG II source member along with your prompt responses to create a format description for the job. If you are setting up a job to create, maintain, or display a data file, DFU requires one RPG II source member. If you are setting up a job to list a data file, DFU requires either one or two RPG II source members:
a. One (that describes the list file) if you are not using a related master file
b. Two (one that describes the list file and one that describes the master file) if you are using a related master file

You must specify the name of the source member in the setup command statement or in the prompt for that parameter of the command statement. If you indicate a related master file in the LIST setup command statement, you must also specify the name of the RPG II source member for the master file in the prompt that appears.

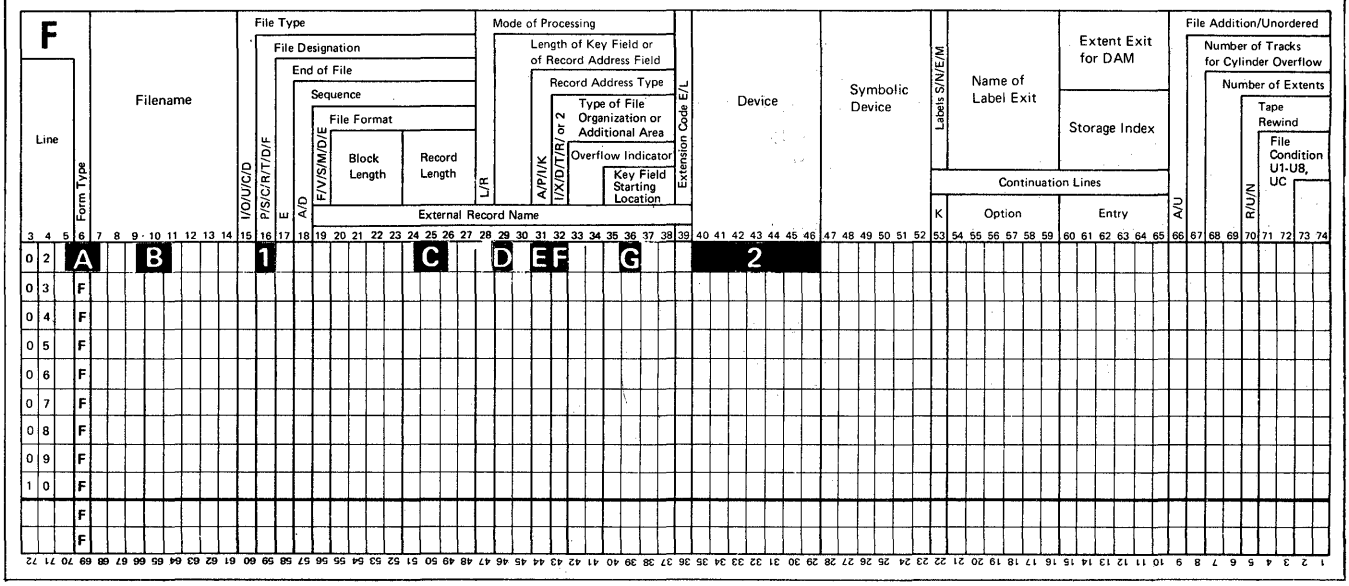
The information in the previous chapters assumed that an RPG II source member had been created, named, and saved for you on disk. To set up and run a DFU job, you only needed to know the name that had been assigned. This chapter provides detailed information about an RPG II source member—what it consists of and the format of its contents. If you have RPG II programming knowledge, you can use this information to create, name, and save an RPG II source member that describes the file you want DFU to process.

FILE DESCRIPTION SPECIFICATION

The file description specification describes the file DFU will process. The following are true of this specification:

- There can be only one file description specification in an RPG II source member.
- The file description specification must be the first non-comment specification in the member.
- You are allowed to make any valid RPG II entry on the file description specification as described in *IBM System/32 RPG II Reference Manual,* SC21-7595. DFU, however, does not require that you supply all of the entries. Figure 27 shows and explains the entries required by DFU.

File Description Specifications



A Form Type – An F must be in this position.

B Filename – Name associated with the file being described.

C Record Length – Length of the records in the file. The maximum record length is 512. This entry must end in position 27.

D Length of Key Field – Length of the record keys. The maximum length is 29. This entry is required only if the file is an indexed file.

Note: If packed keys are used, the maximum length is 8.

E Record Address Type – Format of the record keys. This entry can be either A for unpacked data or P for packed decimal. If A or blank, and the key field is greater than 15 positions in length, DFU assumes the record keys are alphameric; if the key field is 15 positions or less in length, DFU prompts determine if record keys are alphameric or unpacked decimal at job setup time.

F Type of File Organization – This entry must be an I for an indexed file. Any other entry indicates a sequential file.

G Key Field Starting Location – Beginning position in the record of the record key. This entry is required only for indexed files.

Notes:

1. If the file description specification describes a sequential file (DFU can only list such a file), entries **D** and **G** must be blank.
2. If you use SEU to enter the RPG II source member, two additional fields are required:

1 File Type – Must be an I or O.

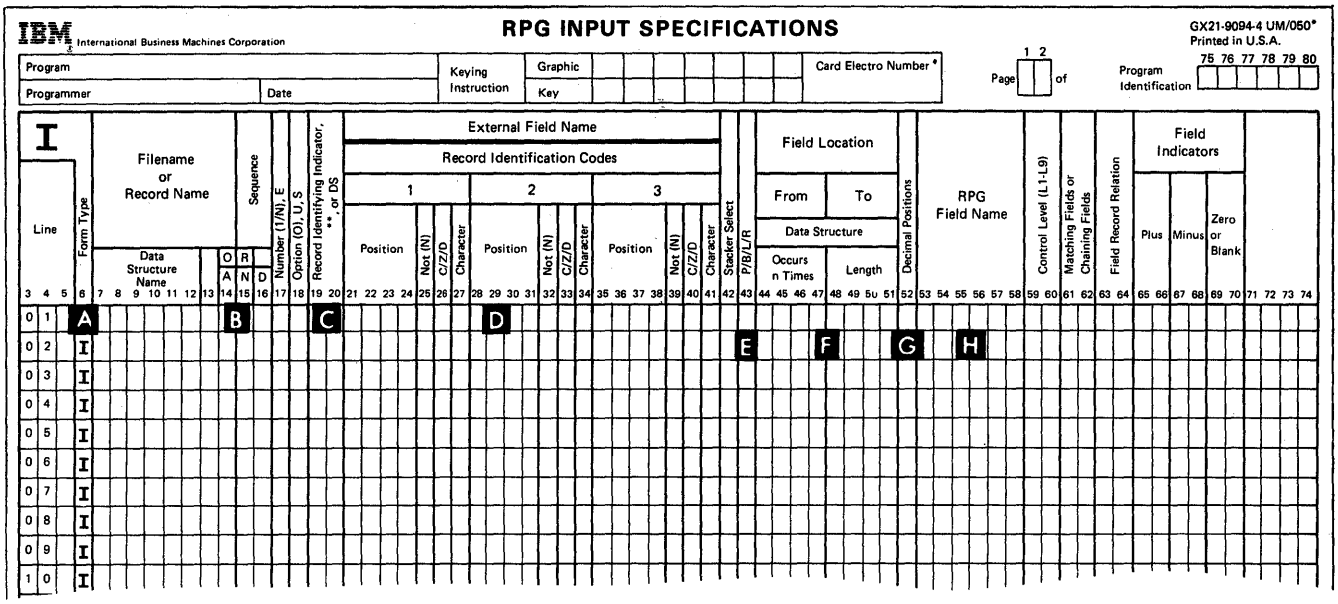
2 Device – Any valid input device. For example, DISK.

Figure 27. File Description Specification Entries Required by DFU

INPUT SPECIFICATIONS

The input specifications further define the file to be processed by DFU. It lists the record identifying indicators, record identifying codes, and field names and field locations. You can use most valid RPG II entries on the input specification as described in *IBM System/32 RPG II Reference Manual,* SC21-7595. The limitations are noted in *RPG II Source Member Considerations* later in this chapter.

DFU, however, does not require that you supply all of the entries. Figure 28 shows and explains the entries required by DFU.



A Form Type — An I must be in this position.

B OR/AND Conditioning — Entries in these positions indicate an *or* or *and* condition with the record identification indicator on the preceding line of the input specification. A maximum of eight record identification codes can be conditioned with an OR/AND conditioning.

C Record Identifying Indicator — Entry is a number from 01 to 99 that is associated with the record type defined in positions 21 to 41. This number appears in the prompt

xx—ANY FIELDS FROM THIS RECORD TYPE?

when responding to prompts in the DFU setup step.

D Record Identification Codes — Entries in these positions identify the different record types in the file. One, two, or three codes can be specified on each line. A maximum of eight codes can be specified by specifying OR/AND relationships in positions 14-16.

Three entries are allowed on each record identification code line. An entry can have a position, not (N), C/Z/D, and character.

The following (positions 43 through 64) describe the fields in each record type. The first field entry cannot be on the same line as a record type entry in positions 21 through 41.

E Packed Decimal Field — Must be a P if the field is packed decimal; otherwise, this position must be blank.

F Field Location — Beginning and ending positions of the field in the record.

G Decimal Positions — The number of decimal positions in the field. This position must be blank for an alphameric field.

H Field Name — Name of the field that occupies the location specified in positions 44-51. This name must be from one to six characters. It must begin with an alphabetic character (A through Z, @, $, or #). The remaining characters can be alphameric or numeric. Blanks cannot appear between characters in the name. In the DFU setup step, you must specify the fields DFU should process. An

ENTER FIELD NAME

prompt occurs. Your response must be one of the field names in positions 53-58 that should be processed. The prompt continues to appear until you:

1. Press the ENTER key when line 6 of the display is blank.
2. Enter all field names in the record type.
3. Enter the maximum 40 fields.

Figure 28. Input Specification Entries Required by DFU

Note: Field record relation (columns 63-64) can be used by DFU, but this field is not required and entries are not checked by DFU for validity before processing.

RPG II SOURCE MEMBER CONSIDERATIONS

DFU requires that:

- Records in a file must be 512 characters or less.
- Fields specified for processing must be 40 characters or less.
- No more than eight record identification codes can be used for each record type.
- Binary fields cannot be used.
- Numeric fields must be 15 positions or less.

RPG II ERRORS

DFU checks the RPG II file description and input specifications for errors before converting them to DFU attributes. If an error is found, the incorrect specification and an error message are printed and the job is ended. You must correct the error(s) before continuing with the job.

CREATING AN RPG II SOURCE MEMBER

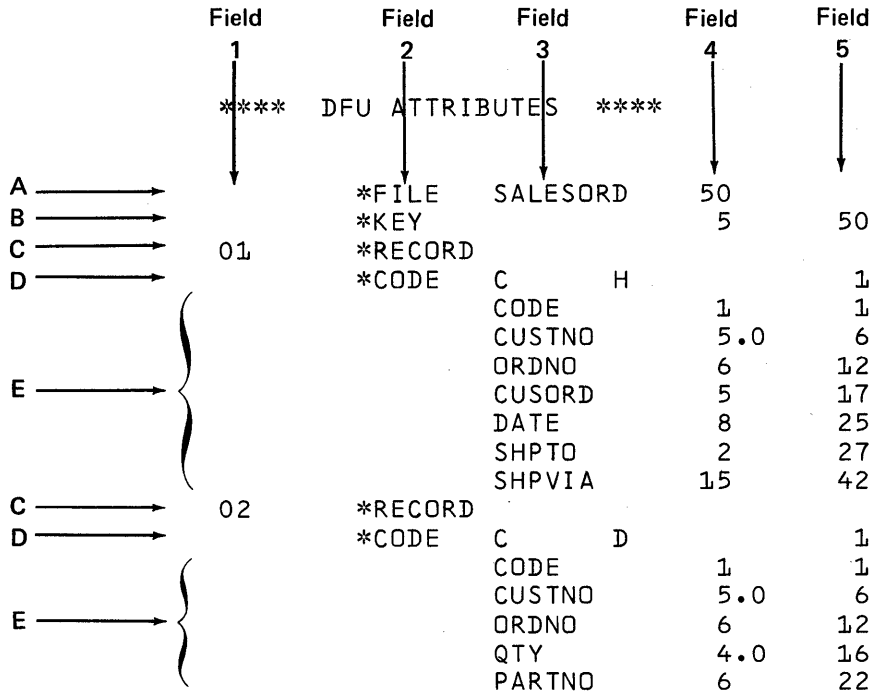
The source entry utility (SEU) program product allows you to enter the file description and input specifications (RPG II source member) that describe the data file you want DFU to process. Key a command statement of the following format:

SEU source member name, R

Then key the file description and input specifications for the RPG II source member (The *IBM System/32 Utilities Program Product Reference Manual—Source Entry Utility,* SC21-7605, explains how you key the file description and input specifications.)

DFU attributes are information about your file that DFU builds from an RPG II source member. The attributes consist of 40-character records divided into five 8-character fields. The DFU attributes appear on the display screen while you respond to prompts in the job setup step.

Figure 29 describes each line of the DFU attributes, field by field. The DFU attributes shown are those built in example 1 in Chapter 3.



A describes the file to be processed.
- Field 1: Blank
- Field 2: *FILE
- Field 3: Name of the file specified in the RPG II source member
- Field 4: Length of the records in the file
- Field 5: Blank

B describes the key field in the file to be processed. This attribute line does not appear if you are listing a sequential file.
- Field 1: Blank
- Field 2: *KEY
- Field 3: Blank
- Field 4: Length of the key field. If packed keys are used, the length is preceded by the letter P.
- Field 5: End position of the key field in the record

Figure 29 (Part 1 of 2). DFU Attributes

Each C identifies the record type described by the DFU attribute lines that follow (the D and E lines).

Field 1: Record identifying indicator for the record type
Field 2: *RECORD
Field 3: Blank
Field 4: Blank
Field 5: Blank

Each D specifies the record identifying codes for the record type described by the DFU attribute lines that follow (the E lines).

Field 1: Blank for the first or only code line; AND or OR for succeeding code lines
Field 2: *CODE
Field 3: Record identifying code
Field 4: Blank
Field 5: Position of the identifying code in the record

E describes the fields in the record type.

Field 1: Blank
Field 2: Blank
Field 3: Name of the field
Field 4: Length of the field. For a numeric field, field 4 contains the length of the field followed by a period and a digit indicating the number of decimal positions in the field. For a packed decimal field, the number in field 4 is preceded by the letter P.
Field 5: End position of the field in the record.

Figure 29 (Part 2 of 2). DFU Attributes

DFU Attributes When Using a Related Master File

If you specify a related master file in the LIST setup command statement, you must also specify the RPG II source member that describes the master file and the name of the field in the list file that DFU uses to retrieve master file records. DFU builds attributes for the list file (using the RPG II source member that describes the list file), then builds attributes for the master file (using the RPG II source member that describes the master file). The master file attributes directly follow the list file attributes and are the same as those described in Figure 29 with the following exceptions:

- In B, field 3 names the field in the list file that corresponds to the key field in the master file.
- C and D do not appear in the DFU attributes. DFU does not determine the record type when retrieving a master record.

Figure 15 in Chapter 3 shows an example of DFU attributes built for a list job that uses a related master file.

Displaying DFU Attributes

DFU attributes are shown on lines 1 through 4 of the display screen while you respond to prompts in the DFU setup step. Initially, as shown in the example below, the first four attributes are displayed.

```
        *FILE   SALESORD    50
        *KEY                 5       50
01      *RECORD
        *CODE   C      H              1
SHOULD RECORDS BE PRINTED?
```

The following keys can be used to display the DFU attributes:

- The ROLL↑ key moves the attributes up one line at a time.
- The ROLL↓ key moves the attributes down one line at a time.
- The DISPLAY ATTR/SPEC command key causes DFU specifications to be displayed on lines 1 through 4 of the display screen (if DFU attributes are currently displayed), or it causes DFU attributes to be displayed on lines 1 through 4 of the display screen (if DFU specifications are currently displayed).

The ROLL↑ and ROLL↓ keys are useful when responding to the

ENTER FIELD NAME

prompt that reappears when setting up a job. They can be used to position the desired DFU attribute field name on line 4 of the display screen. When the DUP key is pressed, the field name is copied from the attribute line onto line 6. This helps speed keying field names and also minimizes keying errors when setting up a job.

The DISPLAY ATTR/SPEC command key is useful to check the DFU specifications as they are being built from your prompt responses. For example, you can check that all field names that you want to define have been defined before keying a null response to the

ENTER FIELD NAME

prompt.

DFU specifications are created by DFU from your responses to prompts in the setup step. They describe, in easily readable form, the exact file processing desired. DFU specifications can be saved as a *source member* in the library and used as input for other DFU jobs, allowing you to skip the prompting sequence in the setup step.

This chapter explains the contents of the DFU specifications, how they can be modified, and how they can be saved and used in subsequent setup steps.

DFU SPECIFICATION LINES

DFU specifications consist of 40-character records (lines). There are five different types of lines:

1. The *header* line is the first line in the DFU specifications. It specifies the DFU job type and special processing to be done when the job is run.

2. The *key* line is the second line. It specifies special processing to be done when the job is run and the heading for the record key field.

3. The *title* line is the third line. It specifies the column spacing value and title for the job.

4. The *record* line(s) specifies the record identifying indicators for the record types to be processed. There is one record line for each record type to be processed. The record lines must be in the same relative order as they are in the RPG II source member.

5. The *operation code* lines describe the processing to be done on the individual fields within the record types. There is one operation code line for each field in the record type to be processed.

   For a record list or summary list of DFU specifications, two or more operation code lines are needed to specify calculated result fields and record selection criteria.

Each specification line consists of five fields. These fields are explained in detail in the following explanation of *Types of DFU Specifications.*

TYPES OF DFU SPECIFICATIONS

The DFU specifications built by DFU are basically the same; however, they differ depending on the type of job being set up. Figure 30 shows the DFU specifications built for the various DFU jobs.

Creating or Maintaining a Data File

Field 1	Field 2	Field 3	Field 4	Field 5
	*ENT/UPD	*LIST or blank	code, position	
	*KEY	*GENKEY	heading	
		*NUMERIC		
		or		
		blank		
col sp	*TITLE	actual title		
R.I.D.	*RECORD			
	*opcode	field	heading	
	*opcode	field	heading	

Displaying a Data File

Field 1	Field 2	Field 3	Field 4	Field 5
	*INQUIRY			
	*KEY	*NUMERIC	heading	
		or blank		
col sp	*TITLE	actual title		
R.I.D.	*RECORD			
	*opcode	field	heading	
	*opcode	field	heading	

Printing a Data File (Record List)

Field 1	Field 2	Field 3	Field 4	Field 5
	*LIST	*RECORD		
	*KEY	*PRINT	heading	
		*NUMERIC		
		or blank		
col sp	*TITLE	actual title		
R.I.D.	*RECORD			
	*opcode	field	heading	
	*opcode	field	heading	

Printing a Data File (Summary List)

Field 1	Field 2	Field 3	Field 4	Field 5
	*LIST	*SUMMARY	*DETAIL or blank	
	*KEY	*PRINT	heading	
		*NUMERIC		
col sp	*TITLE	actual title		
R.I.D.1	*RECORD			
R.I.D.2	*RECORD			
	*opcode	field	heading	
	*opcode	field	heading	

Note: col sp is the column spacing value.
R.I.D. is the record identifying indicator.

Figure 30. Types of DFU Specifications

Header Line Fields

Only fields 2, 3, and 4 are used in the header line. Field 2 identifies the type of DFU job:

*ENT/UPD — enter or update

*INQUIRY — inquiry

*LIST — list

Field 3 has a special meaning, depending on the DFU job type:

Field 2	Field 3	
*ENT/UPD	*LIST	— indicates records will be printed as they are keyed or updated
	Blank	— indicates records will not be printed as they are keyed. However, updated records will be printed
*INQUIRY	Not used	
*LIST	*RECORD	— indicates a record list
	*SUMMARY	— indicates a summary list

Field 4 is used only if field 2 is *ENT/UPD or if fields 2 and 3 are *LIST and *SUMMARY:

Field 2	Field 3	Field 4	
*ENT/UPD	Blank or *LIST	Contains the delete character and position that will be used when records are deleted	
*LIST	*SUMMARY	Blank	— indicates that only control fields will be printed (detail records will not be printed)
		*DETAIL	— indicates that detail records will be printed

Key Line Fields

Fields 2, 3, 4, and 5 are used in the key line.

Field 2 identifies the line as *KEY.

Field 3 has a special meaning, depending on the DFU job type:

Header Line Field 2	Key Line Field 3	
*ENT/UPD	*GENKEY	— indicates DFU generates a 5-digit numeric key
	*NUMERIC	— indicates you supply a numeric key
	Blank	— indicates you supply an alphameric key
*INQUIRY	*NUMERIC	— indicates you supply a numeric key
	Blank	— indicates you supply an alphameric key
*LIST	*PRINT	— indicates DFU prints a key field first for each record
	Blank	— indicates DFU does not print the key field first for each record
	*NUMERIC	— indicates DFU prints a numeric key field first for each record

Fields 4 and 5 contain the heading for the key field, unless field 2 in the header line is *LIST and field 3 in the key line is blank; in this case, fields 4 and 5 are blank.

Title Line Fields

All five fields are used in the title line:

Field 1 indicates the column spacing value used for the job.

Field 2 identifies the line as *TITLE.

Field 3, 4, and 5 contain the title used for the job.

Record and Operation Code Lines Fields

For an enter, update, inquiry, or record list DFU specifications, each record line is immediately followed by all of the operation code lines pertaining to the record line. For summary list DFU specifications, all of the record lines appear consecutively, followed by all of the operation code lines.

Note: In a summary list in which all of the record types are to be included in the list, the record lines will be omitted from the DFU specifications.

Record Line Fields

A record line uses fields 1 and 2. Field 1 contains the record identifying indicator for the record type. Field 2 identifies the line as *RECORD.

Operation Code Line Fields

An operation code line uses fields 2, 3, 4, and 5.

Field 2 contains the operation code describing the special operations for the field named in field 3. Figure 31 shows the operations that can be done. Field 3 contains the field name to be processed. Fields 4 and 5 contain the heading associated with the field.

For a record list or summary list DFU specifications, multiple operation code lines are required for calculated result fields. Multiple operation code lines are possible for record selection criteria.

For a calculated result field, the first operation code line has the following fields:

Field 1: Length of the field, followed by a period and a digit indicating the number of decimal positions in the field.

Field 2: Either * or *ADD. * indicates that no special operation is performed on the calculated result field. *ADD indicates that the result field is to be accumulated.

Field 3: Either *RESULT or +name. *RESULT indicates that the result is not saved. +name specifies the name under which the result will be saved for future calculations.

Fields 4 and 5: Heading associated with the result field.

The succeeding operation code lines have the following fields:

Field 1: ADD, SUB, MULT, or DIV. This indicates the operation performed to calculate the result field. The first operation is always ADD.

Field 2: Blank.

Fields 3, 4, and 5: Name of the field used to calculate the result field, or the constant used to calculate the result field. A constant can be a maximum of 15 digits.

A maximum of six result fields can be specified, and a maximum of four operations can be performed to calculate the result field; therefore, at most, 24 operation code lines can appear in the DFU specifications for the calculated result fields.

Figure 16 in Chapter 3 shows an example of how calculated result fields appear in DFU specifications.

For record selection, a maximum of 10 selection criteria can be specified for the list. There is one operation code line for each comparison of one field to another, and there are two operation code lines for each comparison of a field to a constant. Therefore, at most, 20 operation code lines can appear in the DFU specifications for the record selection criteria. These lines are always the last in the DFU specifications. For a field to field comparison, the operation code line has the following fields:

Field 1: Blank for the first *SELECT line; OR if starting a new set of selection criteria; AND if adding to the previous selection criteria

Field 2: *SELECT

Field 3: Select field name (factor 1)

Field 4: Condition to be satisfied (EQ, NE, GT, LT, GE, or LE)

Field 5: Factor 2 field name

For a field to constant comparison, the first operation code line has the following fields:

Field 1: Blank for the first *SELECT line: OR if starting a new set of selection criteria; AND if adding to the previous selection criteria

Field 2: *SELECT

Field 3: Select field name (factor 1)

Field 4: Condition to be satisfied (EQ, NE, GT, LT, GE, or LE)

Field 5: Blank

The next operation code line has the following fields:

Field 1: Blank

Field 2: B lank

Fields 3 through 5: Constant value, consisting of a maximum 20 characters (factor 2)

Op Code	Operation	Used By
*	No special operation desired.	ENTER, UPDATE, INQUIRY, LIST
*C	Modulus 10 self-check field.	ENTER, UPDATE
*K	Modulus 11 self-check field.	ENTER, UPDATE
*D	Auto dup field.	ENTER, UPDATE
*ADD	Accumulate this field.	ENTER, UPDATE, LIST
*SORTA	Sort on this field in ascending sequence.	LIST
*SORTD	Sort on this field in descending sequence.	LIST
*TOTAL	Use this field as a control field.	LIST
*SELECT	Use this field for record selection.	LIST
*CD	Modulus 10 self-check field and auto dup field.	ENTER, UPDATE
*KD	Modulus 11 self-check field and auto dup field.	ENTER, UPDATE
*ADDC	Modulus 10 self-check field and an accumulator field.	ENTER, UPDATE
*ADDK	Modulus 11 self-check field and an accumulator field.	ENTER, UPDATE
*ADDD	Auto dup field and an accumulator field.	ENTER, UPDATE
*ADDCD	Modulus 10 self-check field and an auto dup field and an accumulator field.	ENTER, UPDATE
*ADDKD	Modulus 11 self-check field and an auto dup field and an accumulator field.	ENTER, UPDATE

Note: The *SORTA, *SORTD, *TOTAL, and *SELECT lines must follow the operation code lines for the last record type.

Figure 31. Operation Codes

UPDATING DFU SPECIFICATIONS

Before the DFU specifications are checked for errors and the format description is built, you are allowed to update the specifications. You would want to update a specification if you had made a mistake when responding to a prompt or if you intend to use a previously saved specification and modify it for the current job being set up.

Initially, the first four DFU specifications are shown on lines 1 through 4 of the display screen and a message appears on lines 5 and 6 (as shown in Figure 32). The ROLL↑ and ROLL↓ keys allow you to display the other lines of the DFU specifications.

```
      *ENT/UPD *LIST      X,1
      *KEY     *GENKEY    *KEY
2     *TITLE   DAILY SALES ORDERS
01    *RECORD
HIT EOJ CMD KEY TO CONTINUE PROCESSING,
OR YOU MAY NOW UPDATE DFU SPECS.
```

Figure 32. Example Initial DFU Specifications Display

Function and Command Keys for Updating DFU Specifications

The following lists and explains the function and command keys you can use when updating the DFU specifications:

Function Key	Use
ENTER	Causes the data to the left of the cursor to be processed by DFU. The remainder of the line, from the cursor position on, is set to blanks. If you press ENTER when inserting records and the cursor is in the first position of a blank line, it indicates you have finished inserting a record or a group of records.
ERROR RESET	Stops the display screen when it is flashing a message. You can then read the message and make the appropriate keyboard response.
FIELD ADV	Serves as a tab key, allowing you to space over DFU specifications field by field. If the cursor is in the last field of the line, FIELD ADV causes the entire line to be processed by DFU.
FIELD BKSP	If the cursor is in the first position of a field, moves it to the first position of the previous field. If the cursor is beyond the first position of a field, returns it to the first position of the field.
REC ADV	When changing a line, causes the entire line to be processed by DFU (regardless of the position of the cursor). When adding a line or group of lines, causes the entire line to be processed by DFU (regardless of the position of the cursor within the line). A blank line appears as the fourth line of display screen. If you press REC ADV when inserting records and the cursor is in the first position of a blank line, it indicates you have finished inserting a record or a group of records.
ROLL↑	Causes the DFU attributes or DFU specifications on lines 1 through 4 of the display screen to be moved up one line to show the next DFU attribute or specification line.
ROLL↓	Causes the DFU attributes or DFU specifications on lines 1 through 4 of the display screen to be moved down one line. The preceding attribute or specifications line is shown on line 1. The fourth line is moved down off the display screen.

Command Key	Use
DELETE $ 4	Causes the specification line on line 4 of the display screen to be deleted.
ADD % 5	Allows a DFU specification line (or a group of lines) to be inserted after the one displayed on line 4. The specifications are moved up one line. Line 4 is blank, allowing you to key the new line(s).
PRINT REC ¬ 6	Causes the DFU attributes and specifications to be listed on the printer.
EOJ & 7	If pressed when the following prompt is displayed, causes the DFU specifications to be checked for errors; and if none are found, converts the specifications to a format description: HIT EOJ CMD KEY TO CONTINUE PROCESSING, OR YOU MAY NOW UPDATE THE DFU SPECS If pressed while correcting a DFU-diagnosed error in the DFU specifications, causes the job to be canceled.
DISPLAY ATTR/ SPEC * 8	Causes the display screen lines 1 through 4 to be changed. If DFU attributes are displayed, the last four DFU specifications that were displayed are shown. If DFU specifications are displayed, the last four DFU attributes that were displayed are shown.

Changing DFU Specifications

Use the ROLL↑ or ROLL↓ key to move the specifications line to be changed to line 4 of the display screen. Press the FIELD ADV key to move the cursor to the first position of the field to be changed (unless you are changing the first field in the line). Change the field(s), pressing FIELD ADV after each one is updated. Press REC ADV instead of FIELD ADV after keying the last change in the line.

For example, assume the DFU specifications in Figure 33 have been created from a prompting sequence.

```
****   DFU SPECIFICATIONS    ****

       *ENT/UPD *LIST      X,1
       *KEY     *GENKEY    *KEY
2      *TITLE   DAILY SALES ORDERS
01     *RECORD
       *D       ORDNO      OUR ORDER NO.
       *        DATE       DATE
       *        CUSORD     CUST ORD. NO.
       *D       CUSTNO     CUSTOMER NO.
       *        SHPTO      SHIP TO
       *        SHPVIA     SHIP VIA
02     *RECORD
       *D       ORDNO      OUR ORDER NO.
       *D       CUSTNO     CUSTOMER NO.
       *ADD     QTY        QUANTITY
       *        PARTNO     PART NUMBER
```

Figure 33. Sample DFU Specifications

```
2     *TITLE   DAILY SALES ORDERS
01    *RECORD
      *D       ORDNO     OUR ORDER NO.
      *        DATE      DATE
HIT EOJ CMD KEY TO CONTINUE PROCESSING,
OR YOU MAY NOW UPDATE DFU SPECS
```

The heading is to be changed from DATE to ORDER DATE for the field named DATE. Move the cursor to the beginning of the heading DATE by pressing the FIELD ADV key three times.

```
2     *TITLE   DAILY SALES ORDERS
01    *RECORD
      *D       ORDNO     OUR ORDER NO.
      *        DATE      DATE
HIT EOJ CMD KEY TO CONTINUE PROCESSING,
OR YOU MAY NOW UPDATE DFU SPECS
```

Key ORDER DATE and press REC ADV to enter the change and the line.

Inserting DFU Specifications

If a line or a group of lines are to be inserted in the DFU specifications, press the ROLL↑ or ROLL↓ key to display the line that will immediately precede the new line(s). Press the ADD command key and enter the specifications a field at a time. After keying the last field in the line, press ENTER or REC ADV to add the line.

When you are finished adding the line(s), press the ENTER or REC ADV key when the cursor is in the first position of a blank line.

For example, assume that the line to be added to the DFU specifications in Figure 33 is:

```
*       CUSNME   CUSTOMER NAME
```

This is to be added after the line:

```
*       CUSORD   CUST ORD. NO.
```

```
      *ENT/UPD *LIST     X,1
      *KEY     *GENKEY   *KEY
2     *TITLE   DAILY SALES ORDERS
01    *RECORD
HIT EOJ CMD KEY TO CONTINUE PROCESSING,
OR YOU MAY NOW UPDATE DFU SPECS
```

Use the ROLL↑ key to move the

```
*  CUSORD   CUST ORD. NO.
```

line to line 4 of the display screen.

```
01    *RECORD
      *D       ORDNO     OUR ORDER NO.
      *        DATE      ORDER DATE
      *        CUSORD    CUST ORD. NO.
HIT EOJ CMD KEY TO CONTINUE PROCESSING,
OR YOU MAY NOW UPDATE DFU SPECS
```

Press the ADD command key. The DFU specifications are moved up one line and line 4 is left blank for you to key the new line.

```
      *        DATE      ORDER DATE
      *        CUSORD    CUST ORD. NO.
      *        CUSNME    CUSTOMER NAME

HIT EOJ CMD KEY TO CONTINUE PROCESSING,
OR YOU MAY NOW UPDATE DFU SPECS
```

Key the line, field by field. Press REC ADV or ENTER after keying CUSTOMER NAME.

Press ENTER or REC ADV to indicate that all additions have been made. (You could make additions after other lines in the DFU specifications.)

Deleting DFU Specifications

A DFU specification line can be deleted by moving it to line 4 of the display screen (using the ROLL↑ or ROLL↓ key) and pressing the DELETE command key. For example, assume the line you want to delete from the DFU specifications in Figure 33 is:

```
*D        CUSTNO   CUSTOMER NO.
```

Use the ROLL↑ or ROLL↓ key to move the

```
*D  CUSTNO   CUSTOMER NO.
```

line to line 4 of the display screen.

```
        *D      ORDNO   OUR ORDER NO.
        *       DATE    ORDER DATE
        *       CUSORD  CUST ORD. NO.
        *D      CUSTNO  CUSTOMER NO.
HIT EOJ CMD KEY TO CONTINUE PROCESSING,
OR YOU MAY NOW UPDATE DFU SPECS
```

Press the DELETE command key. The line is deleted and the next line appears on the display screen.

Checking the DFU Specifications for Errors

When you have finished updating the DFU specifications, press the EOJ command key. This indicates the end of the setup step and causes the DFU specifications to be checked for errors and converted to a format description. After pressing the EOJ command key, you cannot do further updating of the DFU specifications in this setup step. Changes are allowed only if an error is detected.

If an error is detected, the line in error is shown on line 4 of the display screen and an error message is shown on line 6. A 4-character message identification code (MIC) associated with the error is shown in positions 5 through 8 of line 5. Use the MIC to find an explanation of the error in the *IBM System/32 Displayed Messages Guide*, GC21-7704.

The following shows an example of an error that has been detected. The field name on line 4 has been misspelled; it should be ORDNO instead of ORDON.

```
        *       SHPVIA   SHIP VIA
02      *RECORD
        *D      CODE     CODE
        *D      ORDON    OUR ORDER NO.
DFU 0135
UNDEFINED FIELD NAME IN DFU SPEC
```

Press FIELD ADV twice, key the correct field name, and press REC ADV.

Note: If an error is shown that you do not understand, press the PRINT REC command key to get a listing of the DFU specifications and attributes before canceling the job.

SAVING DFU SPECIFICATIONS AND USING DFU SPECIFICATIONS THAT HAVE BEEN SAVED

It is possible to save and name the DFU specifications built during the setup step. They can be used as input for the setup step of a similar job, modified if necessary, then used in building the format description for that job. When using saved DFU specifications, the prompting sequence is skipped in the setup step. You can modify the saved DFU specifications to indicate the processing desired.

How to Save DFU Specifications

The setup command statement indicates whether or not the DFU specifications should be saved. Figure 34 shows the command statements keyed to save the specifications as a *source member* in the library and assign a name to them.

Type of DFU Setup	Command Statement	Parameter Explanation
Creating a data file	ENTER filename,[format description name], RPG II source member name, number of records,,NY,library name	*NY* – indicates that the DFU specifications built during the setup step will be saved in the library as a *source member* and given the name specified by the library name parameter. *library name* – indicates the name assigned to the *source member* that will contain the DFU specifications.
Maintaining a data file	UPDATE filename,[format description name], RPG II source member name,,,NY, library name	*Note:* The filename, format description name, RPG II source member name, number of records, SORT and NOSORT, and master file name parameters are explained in Chapter 3, *DFU Setup Step* and in Appendix A, *Setup and Run Command Statements.*
Displaying data from a data file	INQUIRY filename,[format description name], RPG II source member name,,,NY, library name	
Preparing and printing reports	LIST filename,[format description name], RPG II source member name, [SORT / NOSORT],,NY,library name,master file name	

Figure 34. Setup Command Statements for Saving DFU Specifications

How to Use a Previously Saved DFU Specifications

The setup command statement indicates whether or not a previously saved DFU specifications will be used in the set-up step. If it is, the prompting sequence is skipped. If specified in the command statement, you will be allowed to modify the specifications before a format description is built from them.

Only one parameter needs to be changed in the command statements shown in Figure 34 to indicate that saved DFU specifications will be used.

If you want to use previously saved specifications that need no modification for this setup step, the NY parameter should be GO. The library name parameter specifies the DFU specifications to use. For example:

```
ENTER SALESORD,ORDERFMT,SALESRPG,
450,,GO,DFUSAVED
```

If you want to use previously saved specifications, modify them, and then have them replace the original, the NY parameter in Figure 34 should be YY. The library name parameter specifies the DFU specifications to use, then becomes the name of the modified DFU specifications. For example:

```
UPDATE SALESORD,ORDERFMT,SALESRPG,,,
YY,DFUSAVED
```

If you want to use previously saved specifications, modify it, but then leave the original specifications in their unchanged form, the NY parameter in Figure 34 should be YN. The library name parameter specifies the DFU specifications to use. For example:

```
INQUIRY ITEMAST,ITMASTF2,ITEMRPG,,,
YN,DFUKEPT
```

Note: The format description you name in the command statement must *not* exist if you specify the use of previously saved DFU specifications.

CONVERTING DFU SPECIFICATIONS TO A FORMAT DESCRIPTION

When the DFU specifications have been diagnosed and found to be error-free, DFU converts them to a format description. The format description consists of records that are saved as a load member in the library. The format description is used to control the job run. Once it is created and saved, the format description can be used over and over again. This means that the setup step need only be done once to create the format description.

The format description can be built for one DFU job (for example, creating a data file), then used for another DFU job (for example, listing the same data file). Once the format description is created and saved, it is interchangeable between the various DFU job types.

Note: If you want to remove a format description from the library, use the REMOVE command statement. If you want to copy a format description to diskette, use the FROMLIBR command statement. These command statements are explained in *IBM System/32 System Control Programming Reference Manual,* GC21-7593.

DFU may also be used to process source and procedure members in addition to data file processing. A procedure member is a collection of related OCL (operational control language) or OCL and data statements stored in the library. A source member is also stored in the library; however, it is a collection of records used as input for a program, as opposed to input data contained in a data file.

The records in a source or procedure member have no record keys associated with them. When a source or procedure member is created, DFU allocates a temporary disk file. As the records are keyed, DFU assigns a 5-digit statement number to each statement. The statement number is similar to the record key assigned in a data file. The first statement number starts with 00010 and is incremented by 10 for each following statement. After all records have been keyed, the temporary file is converted to a library member. At this time, the statement numbers are removed and the statements are stored in the library as a sequential file.

When source or procedure members are updated, DFU again allocates a temporary file and assigns numbers to each statement in increments of 10. The first statement is numbered 00010, the second is 00020, etc. This allows up to nine new statements to be added between two previously keyed statements. The size of a source or procedure member being updated is therefore restricted to 9999 statements. When the member has been updated, the updated version in the temporary file replaces the member in the library. The member's statements will be in statement number sequence; however, the statement numbers will be dropped.

Note: Statements deleted during the update will not be written into the library member.

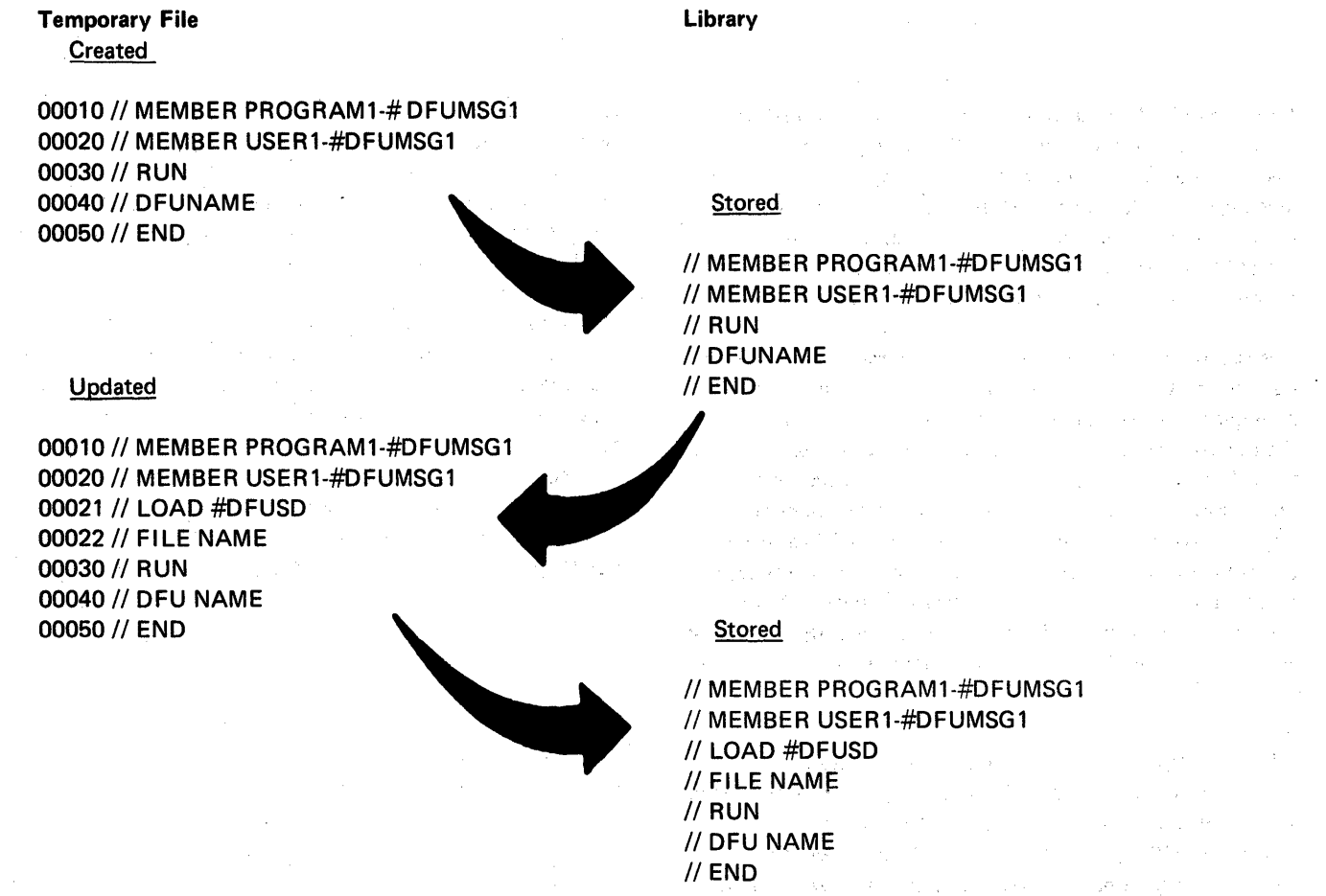
Displaying a source or procedure member is similar to maintaining one, except the specified member is not rewritten to the library when processing is completed.

Using the list function of DFU, only a sequential listing of a source or procedure member can be obtained; no sorting is allowed. You can request in the setup step that DFU print a key for each statement listed by responding yes to the prompt

SHOULD RECORD KEYS BE PRINTED?

In this case, the first field listed for each statement will be a key; the list function will generate this key (starting with 00010 and incrementing by 10 for each successive statement).

The following illustrates how DFU handles source or procedure members:



Note: #DFURPG is an RPG II source member provided with DFU that you can use to set up a DFU job to create a source or procedure member. #DFURPG is described in Appendix C.

COMMAND STATEMENTS

The command statements used to set up and process source or procedure members are similar to those used with data files. The primary difference between the command statements is that for source or procedure members, the file type parameter is required (otherwise, a data file is assumed).

Figure 35 lists the command statements for processing source and procedure members.

Refer to Appendix A, *Setup and Run Command Statements,* if you need an explanation of the command statement parameters.

DFU Setup Command Statements

```
{ENTER  }
{UPDATE } filename,format description name,RPG II source member name,number of records,{S},[GO  ] ,library name
{INQUIRY}                                                                               {P} [YY  ]
                                                                                            [YN  ]
                                                                                            [NY  ]
                                                                                            [NN  ]

LIST    filename,format description name,RPG II source member name,,{S},[GO  ] ,library name
                                                                    {P} [YY  ]
                                                                        [YN  ]
                                                                        [NY  ]
                                                                        [NN  ]
```

DFU Run Command Statements

```
{ENTER  }
{UPDATE } filename,format description name,,number of records, {S}
{INQUIRY}                                                      {P}

LIST    filename,format description name,,, {S}
                                            {P}
```

Figure 35. DFU Setup and Run Command Statements for Processing Source and Procedure Members

Figure 36 shows the formats of the ENTER, UPDATE, INQUIRY, and LIST command statements for setting up and running DFU jobs. These command statements can be keyed in one of the following ways:

- Key the command statement and supply all of the required parameters according to the coding rules in Appendix B. For example, you can key:

  ENTER SALESORD,ORDERFMT,SALESRPG,
  450,D,NY,SALESDFU

  or

  ENTER SALESORD,ORDERFMT,SALESRPG,
  450,,NY,SALESDFU (since the default value for the fifth parameter is D if you omit it)

  The shaded parameters in Figure 36 are prompted for if you omit them. These parameters are:

  Filename
  Format description name
  RPG II source member name
  Number of records
  [SORT / NOSORT]

  Therefore, you can key the previous command statement example as:

  ENTER ,,,,D,NY,SALESDFU

  Then have DFU prompt you for the omitted parameters.

- Key the initial word of the command statement (ENTER, UPDATE, INQUIRY, or LIST), then have DFU prompt you for the remaining parameters.

  DFU does not prompt for the following parameters:

  [S / P / D]

  [GO / YY / YN / NY / NN]

  [library name]

  [master file name]

Figure 37 lists and explains all command statement parameters.

Use	Command Statement Format	Examples
Build a format description to create a data file, source member, or procedure member.	ENTER filename,[format description name], RPG II source member name,number of records, [S / P / D], [GO / YY / YN / NY / NN] [,library name]	ENTER SALESORD, ORDERFMT,SALESRPG,450, D,NY,SALESDFU
Create a data file, source member, or procedure member using an existing format description.	ENTER filename,format description name,, number of records	ENTER SALESORD, ORDERFMT,,50
Build a format description to update a data file, source member, or procedure member.	UPDATE filename,[format description name], RPG II source member name,, [S / P / D], [GO / YY / YN / NY / NN] [,library name]	UPDATE SALESORD, ORDRFMT1,SALESRPG,, D,GO,SALESDFU
Update a data file, source member, or procedure member using an existing format description.	UPDATE filename,format description name	UPDATE SALESORD, ORDRFMT1

Figure 36 (Part 1 of 2). DFU Command Statements

Use	Command Statement Format	Examples
Build a format description to display a data file, source member, or procedure member.	INQUIRY filename,[format description name], RPG II source member name,, [S P D], [GO YY YN NY NN] [,library name]	INQUIRY ITEMAST, ITMASTF2, ITEMRPG,,D
Display a data file, source member, or procedure member using an existing format description.	INQUIRY filename,format description name	INQUIRY ITEMAST, ITMASTF2
Build a format description to print a report.	LIST filename,[format description name], RPG II source member name, [SORT NOSORT], [S P D], [GO YY YN NY NN] ,[library name] [,master file name]	LIST SALESORD,SALESLST, SALESRPG,SORT,D,NY, SALSLST1
Print a report using an existing format description.	LIST filename,format description name,, [SORT NOSORT] ,,,[,master file name]	LIST SALESORD,SALESLST,, SORT

Figure 36 (Part 2 of 2). DFU Command Statements

Parameter	Explanation
filename	Specifies the name of the file being created, updated, displayed, or listed. The name must not be more than eight characters in length and must start with an alphabetic character (A through Z, #, $, or @). For ENTER, it must not be an existing name. However, for UPDATE, INQUIRY, or LIST, it must be an existing name. The filename parameter is required. If omitted, a prompt is issued for it.
format description name	Specifies either (1) the name of the format description that DFU will build and save as a *load member* in the system library or (2) the name of an existing format description. The name must be eight characters or less and must begin with an alphabetic character. Since the format description is saved and given the name you specify, you need only remember this name in order to skip the setup step the next time the job is run. *Notes:* if you do not want to save the format description for a later run, omit this parameter. DFU will then build the format description in a system load member called #DFUOBJ. DFU will remove the format description from the system library after you run the job. *The format description must not already exist if you intend to save or use existing DFU specifications (use the GO, YY, YN, or NY and library name parameters).*

Figure 37 (Part 1 of 4). DFU Command Statement Parameter Explanations

Parameter	Explanation
RPG II source member name	Specifies the name of the RPG II source member in the system library. The name must be eight characters or less and must begin with an alphabetic character. *Note:* For the LIST command statement, this is the name of the RPG II source member that describes the list file.
number of records	Specifies the maximum number of records you will enter in the file. This parameter is not required for the UPDATE or INQUIRY command statements, unless a source or procedure member with more than 500 statements will be processed.
[SORT NOSORT]	Indicates whether or not the data file should be sorted before it is listed. SORT indicates that a sorted report should be created; NOSORT indicates that a sorted report is not desired. This parameter is valid only for data files.
[S P D]	Indicates the type of file being created, maintained, displayed, or listed, where: S = Source member in the library P = Procedure member in the library D = Data file This parameter is optional. If it is omitted, DFU assumes that a data file will be processed.

Figure 37 (Part 2 of 4). DFU Command Statement Parameter Explanations

Parameter	Explanation
[GO YY YN NY NN]	This parameter allows you to: 1. Save DFU specifications in a *source member* for future use before DFU creates a format description from it. 2. Either modify or use without modification a DFU specification that has been previously saved in a *source member*. This enables you to skip the prompting sequence in the job setup step. In order to use this parameter, you must supply a nonexisting format description name in the command statement. If you omit this parameter or if you have DFU prompt you for the parameters, DFU assumes the NN parameter. The parameters have the following meanings: *GO* — The DFU specifications (as specified by the library name parameter) that were saved as a *source member* from a previous run are used as input for this run. The prompting sequence of the DFU setup step will be skipped. You cannot update these DFU specifications before the format description is built. *YY* — The DFU specifications (as specified by the library name parameter) that were saved as a *source member* from a previous run are used as input for this run. The prompting sequence of the DFU setup step will be skipped. You can update these DFU specifications and the resulting DFU specifications are saved (under library name) before the format description is built.

Figure 37 (Part 3 of 4). DFU Command Statement Parameter Explanations

Parameter	Explanation
	YN — The DFU specifications (as specified by the library name parameter) that were saved as a *source member* from a previous run are used as input for this run. The prompting sequence of the DFU setup step will be skipped. You can update these DFU specifications before the format description is built. However, the resulting DFU specifications are not saved. The DFU specifications used as input for this run remain unchanged. *NY* — The DFU specifications (as specified by the library name parameter) do not currently exist in the library. The DFU specifications are created from your responses to a series of prompts and are saved as a *source member* and given the library name before the format description is built. *NN* — The DFU specifications are created from your responses to a series of prompts and are not saved.
library name	Specifies the name of the *source member* in which to save DFU specifications or the name of an existing *source member* in which DFU specifications have been previously saved. The name must not be more than eight characters in length and must begin with an alphabetic character (A through Z, #, $, or @). This parameter is required if the DFU specifications processing parameter is GO, YY, YN, or NY. *Note:* In order to use this parameter, you must supply a nonexisting format description name in the command statement.
master file name	Specifies the name of the indexed file that contains master file information for the list file. If you omit this parameter, DFU assumes that there is no master file to process.

Figure 37 (Part 4 of 4). DFU Command Statement Parameter Explanations

Command	Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5	Parameter 6	Parameter 7	Parameter 8
ENTER	Filename	DFU format name	RPG II source name	Number of records	Filetype	DFU source processing	Library name	
UPDATE	Filename	DFU format name	RPG II source name	Number of records	Filetype	DFU source processing	Libary name	
INQUIRY	Filename	DFU format name	RPG II source name	Number of records	Filetype	DFU source processing	Library name	
LIST	Filename	DFU format name	RPG II source name	SORT/ NOSORT	Filetype	DFU source processing	Libary name	Master file name
SORT	Input filename	Source member name	Output filename	Number of records				

Figure 38. Command Statement Parameters

SYMBOLS

The symbols [ ] and { } are used in this publication to define command statement parameters. They are only used to indicate how a command statement may be written, and are not keyed as a part of the statement.

Brackets [ ] are used to identify optional parameters, while braces { } identify mandatory parameters. If a parameter within the brackets or braces is underlined, DFU assumes that parameter if none of the options are chosen.

Example:

$$\left\{\begin{matrix}\text{ENTER}\\\text{UPDATE}\end{matrix}\right\} \text{filename,[format description],[rpgname],,} \left[\begin{matrix}\text{S}\\\text{P}\\\underline{\text{D}}\end{matrix}\right]$$

In this example, the $\left\{\begin{matrix}\text{ENTER}\\\text{UPDATE}\end{matrix}\right\}$ parameter is required; one of the two identified must be entered as indicated by the braces. The [format description] and [rpgname] parameters need not be entered, as they are optional parameters. Filetype $\left[\begin{matrix}\text{S}\\\text{P}\\\underline{\text{D}}\end{matrix}\right]$ presents another situation. The parameter is optional as indicated by the brackets; however, if the parameter is used, one of the three specified letters must be entered. The underlined D indicates that if none of the options is chosen, DFU assumes D.

GENERAL CODING RULES

The following general coding rules must be followed:

- Leave one or more blanks between the end of the function name and the first parameter.
- If you need more than one parameter, use a comma to separate them. No blanks are allowed within or between parameters.
- Write the parameters in the order they are shown in this manual.
- Commas are used in place of parameters not entered:

  UPDATE filename,,rpgname

In this case, the format description name was not specified. However, commas are not used to replace omitted parameters at the end of a command statement:

UPDATE filename,format description

#DFURPG is a source member supplied with DFU that can be used to help you create the following:

- RPG II source members
- Sort sequence specifications to be used as input to the sort function (described in Appendix D)
- Procedure members

#DFURPG contains RPG II file description and input specifications that describe the records in each of the above three items. The following example shows the file and input specifications of #DFURPG.

```
0102 F#DFURPG IP   120 120
     I *RECORD TYPE 01 IS USED TO ENTER SOURCE MEMBERS IN THE LIBRARY
0201 I#DFURPG    01   6NC
0202 I       OR       7 C*
0203 I                                        1   5 XXXX5
0204 I                                        6  10 XXX10
0205 I                                       11  15 XXX15
0206 I                                       16  20 XXX20
0207 I                                       21  25 XXX25
0208 I                                       26  30 XXX30
0209 I                                       31  35 XXX35
0210 I                                       36  40 XXX40
0211 I                                       41  45 XXX45
0212 I                                       46  50 XXX50
0213 I                                       51  55 XXX55
0214 I                                       56  60 XXX60
0215 I                                       61  65 XXX65
0216 I                                       66  70 XXX70
0217 I                                       71  75 XXX75
0218 I                                       76  80 XXX80
0219 I                                       81 100 XX100
0220 I                                      101 120 XX120
     I* RECORD TYPE 02 IS USED TO ENTER PROCEDURE MEMBERS IN THE LIBRARY
0301 I           02   I C/  2 C/
0302 I       OR       I C*
0303 I                                        1  40 FRST40
0304 I                                       41  80 SECD40
0305 I                                       81 120 THRD40
     I* RECORD TYPE 03 IS USED TO ENTER SORT SPECIFICATIONS IN THE LIBRARY
     I* THIS RECORD TYPE CAN ALSO BE USED TO ENTER SOURCE MEMBERS
0101 I           03   6NC
0102 I       OR       7 C*
0103 I                                        1  16 FRST16
0104 I                                       17  32 SECD16
0105 I                                       33  40 THRD08
0106 I                                       41  80 COMMNT
```

DFU allows you to use the sort function (described in *IBM System/32 Utilities Program Product Reference Manual—Sort,* SC21-7633) by keying a SORT command statement.

The SORT command statement is:

SORT input filename,source member name,
output filename, number of records

input filename is the name of an existing data file to be sorted. This filename must not be more than eight characters in length, and must start with an alphabetic character.

source member name is the name of the source member containing the sequence specifications. (See *IBM System/32 Utilities Program Product Reference Manual—Sort,* SC21-7633, for a description of the sequence specifications.) The source member name must not be more than eight characters in length and must start with an alphabetic character. If the source member does not exist, the DFU ENTER procedure is called to allow you to create the sequence specifications.

output filename is the name of the file that will contain the sorted data. This must not be an existing filename. The filename must not be more than eight characters in length, and must start with an alphabetic character.

Note: By using the SORT command statement, you cannot replace an existing file with the sort output file; however, you can do this by executing sort through your own OCL statements.

number of records is the number of records the new output file will contain.

The command statement can be keyed in its entirety, or the word SORT can be keyed, causing the following prompts:

ENTER FILENAME OF FILE TO BE SORTED

ENTER SORT SPECS SOURCE MEMBER NAME

ENTER FILENAME FOR SORT OUTPUT FILE

ENTER NUMBER OF RECORDS TO BE SORTED

After the SORT command statement has been entered, all the parameters are checked to ensure their accuracy. Missing parameters will be prompted for, and a message will be displayed when errors are encountered. If the source member does not exist, the DFU ENTER procedure is called to allow you to create a source member of sort sequence specifications. (#DFUSORT is a special format description supplied with DFU that can be used to create sequence specifications for sort.) The sequence specifications are stored under the source member name in the library. (*IBM System/32 Utilities Program Product Reference Manual—Sort,* SC21-7633, describes the meaning of each entry in the sequence specifications.)

Example

Suppose you have a data file named CASH containing all your cash transactions for the past month. Each record in CASH is 50 characters long and contains the following information:

Position	Field Description
1	Record code: R—sales transactions P—credit transactions
2-6	Customer number
7-14	Date of transaction
15-20	Invoice number
21-26	Amount of transaction
27-45	Reason for credit (only applicable on a credit transaction)

For a special report at the end of the month you require a data file that consists of only the sales transactions for the month. Within this data file the records must be in ascending order according to the customer number. In addition, the transaction records for each different customer number must be in ascending order by the date of transaction.

Your sequence specifications are stored in the source member CASHSPEC and your output file is named SALES1. You wish to sort approximately 200 records. You would key the following command statement:

```
SORT CASH,CASHSPEC,SALES1,200
```

Appendix E. Calculating the Modulus 10 and Modulus 11 Self-Check Digits

MODULUS 10 SELF-CHECK DIGIT

To compute the modulus 10 self-check digit, do the following:

1. Multiply the units position and every alternate position of the base number by 2.
2. Add the digits in the products to the digits in the base number that were not multiplied.
3. Subtract the sum from the next higher number ending in zero (if the sum ends in zero, the self check digit will be zero).

The difference is the self-check digit.

For example:

Base number	6 1 2 4 8
Units position and every alternate position	6 2 8
Multiply by 2	12 4 16
Digits not multiplied	1 4
Add	1+2+1+4+4+1+6 = 19
Next higher number ending in 0	20
Subtract	-19
Self-check digit	1

MODULUS 11 SELF-CHECK DIGIT

To compute the modulus 11 self-check digit, do the following:

1. Assign a weighting factor to each digit position of the base number. These factors are: 2, 3, 4, 5, 6, 7, 2, 3, 4, 5, 6, 7, 2, 3, ... starting with the units position of the number and progressing toward the high-order digit. For example, the base number 991246351 would be assigned the weighting factors as follows:

base number	9	9	1	2	4	6	3	5	1
weighting factor	4	3	2	7	6	5	4	3	2

<

2. Multiply each digit by its weighting factor.
3. Add the products .
4. Divide this sum by 11.
5. Subtract the remainder from 11.

The difference is the self-check digit.

For example:

Base number	1 3 7 3 9
Weighting factors	6 5 4 3 2
Multiply	6 15 28 9 18
Add	6 + 15 + 28 + 9 + 18 = 76
Divide	76/11 = 6 plus a remainder of 10
Subtract	11-10 = 1
Self-check digit	1

Note: If the remainder from step 4 is 0, the self-check digit is 0. If the remainder is 1, the base number has no self-check digit; you must ensure that these base numbers are not used in the fields you define as self-check fields.

Appendix F. DFU Setup Sheets

The following setup sheets and their examples are intended to aid you in performing the DFU setup step. The setup step can apply to creating a data file, maintaining a data file, displaying records from a data file, and printing reports from a data file. A sample of each sheet is provided in this appendix, along with blank copies that you can use to make additional copies.

DFU ENTER/UPDATE SETUP SHEET

Programmer John Smith

(ENTER) or UPDATE

filename	format name	RPG name	# of rcds	type of file (S P D)	specs (GO YY YN NY NN)	DFU specs library name
MSTINV	MSTNTRY	MSTFDI	500			MSTNTRY

RECORDS PRINTED? (YES) NO DELETE CODE, POSITION D, 85

GENERATE KEYS? YES (NO) RECORD KEYS ALL NUMERIC? (YES) NO

COLUMN HEADING FOR KEYS ITEM# COL SPACING 2

TITLE MSTINV FILE LOAD

01 ANY FIELDS FROM THIS RECORD TYPE? (YES) NO
(Use a separate sheet for each record type.)

Field Name	RCD ADV or ENTER	Column Heading	RCD ADV or ENTER	AUTO DUP	Accumulate This Field?	Self Check	#
RCDCD	RA (E)	CD	RA (E)	(Y) N RA	Y (N) RA	Y (N)	10 11
ITDSC	RA (E)	DESCRIPTION	(RA) E	Y N RA	Y N RA	Y N	10 11
VNDNR	(RA) E		RA E	Y N RA	Y N RA	Y N	10 11
LSTPC	RA (E)		RA (E)	Y (N) RA	(Y) N RA	Y (N)	10 11
CSTPC	RA (E)		RA (E)	Y (N) RA	(Y) N RA	Y (N)	10 11
QTYOH	(RA) E		RA E	Y N RA	Y N RA	Y N	10 11
QTYOO	(RA) E		RA E	Y N RA	Y N RA	Y N	10 11
QTYRS	(RA) E		RA E	Y N RA	Y N RA	Y N	10 11
SAQMO	(RA) E		RA E	Y N RA	Y N RA	Y N	10 11
SAQYR	(RA) E		RA E	Y N RA	Y N RA	Y N	10 11
SALMO	RA (E)		RA (E)	Y (N) RA	(Y) N RA	Y (N)	10 11
SALYR	RA (E)	YR-TO-DATE	RA (E)	Y (N) RA	(Y) N RA	Y (N)	10 11
DELET	(RA) E		RA E	Y N RA	Y N RA	Y N	10 11
	RA E		RA E	Y N RA	Y N RA	Y N	10 11
	RA E		RA E	Y N RA	Y N RA	Y N	10 11

Notes or special instructions:

DFU INQUIRY SETUP SHEET

Programmer John Smith

INQUIRY MSTINV (filename), INQI (format name), MSTFDI (RPG name), _____ (# of rcds), _____ (type of file: D/S/P), _____ (specs: NN/GO/NY/YN/YY), _____ (DFU specs library name)

RECORD KEYS ALL NUMERIC? (YES) NO

COLUMN HEADING FOR KEYS ITEM # COL SPACING 3

TITLE INQUIRY

01 ANY FIELDS FROM THIS RECORD TYPE? (YES) NO
(Use a separate sheet for each record type.)

Field Name	RCD ADV or ENTER	Column Heading
ITDSC	RA (E)	DESCRIPTION
QTYOH	(RA) E	
CSTPC	(RA) E	
LSTPC	(RA) E	
	RA (E)	
	RA E	
	RA E	
	RA E	
	RA E	
	RA E	
	RA E	
	RA E	
	RA E	
	RA E	
	RA E	

PRINT REC

Notes or special instructions: When INQI is executed, all inquiries should be printed by pressing [CMD] [PRINT REC].

DFU LIST SETUP SHEET

Programmer John Smith

LIST MSTINV (filename), INVLI (format name), MSTFDI (RPG name), SORT / [NOSORT] (sort), S / P / D (type of file), GO / YY / YN / NY / NN (specs), INVLIB (DFU specs library name), _____ (master file name)

SUMMARY LIST? (YES) NO DETAIL RECORDS LISTED? (YES) NO

RECORD KEYS PRINTED? (YES) NO RECORD KEYS ALL NUMERIC? (YES) NO

COLUMN HEADING FOR KEYS ITEM# COL SPACING 2

TITLE SPECIAL

01 ANY FIELDS FROM THIS RECORD TYPE? (YES) NO ALL

Field Name, *Result, Lgth. Dec (6.2)	RCD ADV or ENTER	Column Heading	RCD ADV or ENTER	Accumulate This Field?
ITDSC	RA (E)→	DESCR	(RA) E→	Y N
LSTPC	RA (E)	LIST	(RA) E	Y N
QTYOH	(RA) E		RA E	Y N
*❶ 7.2	RA E	VALUE	RA (E)	(Y) N
QTYOO	RA E		RA E	Y N
SAQYR	RA E	QTY-YR	(RA) E	Y N
SALYR	RA E	$SALES-YR	(RA) E	Y N
*❷ 7.2	RA E	AVG SALE	(RA) E	Y N
	RA E		RA E	Y N
	RA E		RA E	Y N

Use These Lines to Code Factors and Operations for *Result Fields;
For Example, QTYOH, QTYOO, ADD, LSTPC, MULT, SALYR, ADD.

* Press the RESULT FIELD command key.
❶ LSTPC QTYOH MULT
❷ SALYR SAQYR DIV

CONTROL FIELDS? Y (N)
NAMES _____, _____, _____, _____, _____

SORT FILE? Y (N)
NAMES _____, _____, _____, _____, _____

ASCENDING? Y N Y N Y N Y N Y N

SELECT RECORDS? (Y) N (If yes, see coding sheet) [PRINT REC] [EOJ] [NOSORT] SORT

DFU LIST SETUP SHEET (continued)

Programmer: John Smith

IF SELECT RECORDS YES:

	Factor 1			Factor 2 constant?		Factor 2
	ITNBR	EQ NE GT (GE) LT LE		(Y)	N	20900
(AND) OR	ITNBR	EQ NE GT GE LT (LE)		(Y)	N	40000
(AND) OR	SAQYR	EQ NE GT GE (LT) LE		(Y)	N	10
(AND) OR	LSTPC	EQ NE GT GE LT (LE)		(Y)	N	30000
AND (OR)	SAQYR	EQ NE (GT) GE LT LE		Y	(N)	QTYOH

Notes or special instructions: _____

DFU LIST SETUP SHEET (continued)

Programmer *John Smith*

IF SELECT RECORDS YES:

	Factor 1				Factor 2
	ITNBR	EQ NE GT **(GE)** LT LE	Factor 2 constant? **(Y)** N		*12000*
(AND) OR	*ITNBR*	EQ NE GT GE LT **(LE)**	Factor 2 constant? **(Y)** N		*13000*
(AND) OR	*SAQYR*	EQ NE GT GE **(LT)** LE	Factor 2 constant? **(Y)** N		*10*
(AND) OR	*LSTPC*	EQ NE GT GE **(LT)** LE	Factor 2 constant? **(Y)** N		*35000*
(AND) OR	*QTHOH*	EQ NE GT GE **(LT)** LE	Factor 2 constant? **(Y)** N		*1*

Notes or special instructions: _____

_____

_____

DFU ENTER/UPDATE SETUP SHEET

Programmer ____________________ Page ___ of ___

					S P D	GO YY YN NY NN	
ENTER or UPDATE	______,	______,	______,	______,	______,	______,	______
	filename	format name	RPG name	# of rcds	type of file	specs	DFU specs library name

RECORDS PRINTED? YES NO DELETE CODE, POSITION ______

GENERATE KEYS? YES NO RECORD KEYS ALL NUMERIC? YES NO

COLUMN HEADING FOR KEYS ____________________ COL SPACING ____

TITLE ______________________________

___ANY FIELDS FROM THIS RECORD TYPE? YES NO
(Use a separate sheet for each record type.)

Field Name	RCD ADV or ENTER	Column Heading	RCD ADV or ENTER	AUTO DUP		Accumulate This Field?		Self Check	#
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11
_______	RA E	____________________	RA E	Y N	RA	Y N	RA	Y N	10 11

Notes or special instructions: ____

____

____

DFU INQUIRY SETUP SHEET

Programmer _____ Page___ of___

INQUIRY _____, _____, _____, _____, _____, _____, _____

filename	format name	RPG name	# of rcds	type of file (D/S/P)	specs (NN/GO, NY, YN, YY)	DFU specs library name

RECORD KEYS ALL NUMERIC? YES NO

COLUMN HEADING FOR KEYS _____ COL SPACING _____

TITLE _____

___ANY FIELDS FROM THIS RECORD TYPE? YES NO
(Use a separate sheet for each record type.)

Field Name	RCD ADV or ENTER	Column Heading
_____	RA E	_____
_____	RA E	_____
_____	RA E	_____
_____	RA E	_____
_____	RA E	_____
_____	RA E	_____
_____	RA E	_____
_____	RA E	_____
_____	RA E	_____
_____	RA E	_____
_____	RA E	_____
_____	RA E	_____
_____	RA E	_____
_____	RA E	_____
_____	RA E	_____

PRINT REC

Notes or special instructions: _____

DFU LIST SETUP SHEET

Programmer _____ Page __ of __

LIST _____, _____, _____, SORT / NOSORT, S / P / D, GO / YY / YN / NY / NN, _____, _____

filename, format name, RPG name, sort, type of file, specs, DFU specs library name, master file name

SUMMARY LIST? YES NO DETAIL RECORDS LISTED? YES NO

RECORD KEYS PRINTED? YES NO RECORD KEYS ALL NUMERIC? YES NO

COLUMN HEADING FOR KEYS _____ COL SPACING _____

TITLE _____

___ANY FIELDS FROM THIS RECORD TYPE? YES NO ALL

Field Name, *Result, Lgth. Dec (6.2)	RCD ADV or ENTER	Column Heading	RCD ADV or ENTER	Accumulate This Field?
_____	RA E	_____	RA E	Y N
_____	RA E	_____	RA E	Y N
_____	RA E	_____	RA E	Y N
_____	RA E	_____	RA E	Y N
_____	RA E	_____	RA E	Y N
_____	RA E	_____	RA E	Y N
_____	RA E	_____	RA E	Y N
_____	RA E	_____	RA E	Y N
_____	RA E	_____	RA E	Y N
_____	RA E	_____	RA E	Y N

Use These Lines to Code Factors and Operations for *Result Fields;
For Example, QTYOH, QTYOO, ADD, LSTPC, MULT, SALYR, ADD.

_____

_____

_____

_____

CONTROL FIELDS? Y N
NAMES _____, _____, _____, _____, _____

SORT FILE? Y N
NAMES _____, _____, _____, _____, _____

ASCENDING? Y N Y N Y N Y N Y N

SELECT RECORDS? Y N (If yes, see coding sheet) PRINT REC EOJ NOSORT SORT

DFU LIST SETUP SHEET (continued)

Programmer _____ Page ___ of ___

IF SELECT RECORDS YES:

	_____ Factor 1	EQ NE GT GE LT LE	Factor 2 constant?	Y	N	_____ Factor 2
AND OR	_____ Factor 1	EQ NE GT GE LT LE	Factor 2 constant?	Y	N	_____ Factor 2
AND OR	_____ Factor 1	EQ NE GT GE LT LE	Factor 2 constant?	Y	N	_____ Factor 2
AND OR	_____ Factor 1	EQ NE GT GE LT LE	Factor 2 constant?	Y	N	_____ Factor 2
AND OR	_____ Factor 1	EQ NE GT GE LT LE	Factor 2 constant?	Y	N	_____ Factor 2

Notes or special instructions: _____

_____

_____

This appendix describes printed DFU programmer messages. These messages and their associated message identification codes are printed when errors are detected in RPG II specifications. For a description of DFU operator error messages, see the *IBM System/32 Displayed Messages Guide*, GC21-7704.

DFU-0100 RECORD LENGTH NOT NUMERIC

An error exists in the RPG II file description specification. The record length specified in the RPG II file description specification (columns 24 through 27) must be numeric.

DFU-0101 KEY START OR LENGTH NOT VALID NUMERIC

An error exists in the RPG II file description specification. The record key length (columns 29 and 30) or record key start (columns 35 through 38) specified in the RPG II file description specification must be numeric and greater than zero when processing a data file with the enter, update, or inquiry functions. This is also true for the list function if record key length and record key start are specified.

DFU-0123 INVALID FIELD NAME

An invalid field name was specified.

If the message is issued while you are entering DFU specifications from the keyboard, a field name is invalid. Correct the field name.

If the message is issued while DFU specifications are being diagnosed, a field name is either invalid or undefined. The specification in error is displayed on line 4 of the display screen. Do one of the following:

- Key the correct field name.
- Delete the specification by pressing the DELETE command key.
- Cancel the job by pressing the EOJ (end of job) command key.

If RPG II specifications are being diagnosed, this message indicates an input specification contains an invalid field name. No operator action is required.

Note: The first character of a field name must be alphabetic (A through Z, @, #, or $), and the field name cannot contain more than six characters. All characters after the first must be alphameric.

DFU-0134 DUPLICATE RECORD ID INDICATOR

The RPG II specifications contain duplicate record ID indicators.

DFU-0148 RECORD LENGTH TOO LONG – COLS 24-27

An error exists in the RPG II file description specification. The specified record length must be less than 121 for source and procedure members and less than 513 for data files.

DFU-0150 FILE ORG (COL 32) MUST BE 'I' FOR KEYS

An error exists in the RPG II file description specification. Position 32 of the RPG II file description specification must contain an I for indexed data files.

DFU-0151 INVALID RECORD ID INDICATOR – COLS 19-20

An error exists in an RPG II input specification. A record identification indicator in positions 19 and 20 of the RPG II input specifications must be present for each record type and must be a valid two-digit number.

DFU-0153 INVALID RPG II SPECIFICATION DETECTED

An error exists in an RPG II specification. One of the following errors was detected:

- The first RPG II specification does not contain an F in column 6.
- The RPG II specification does not contain an I in column 6.
- DFU cannot determine whether an input specification is a record identifier or a field specification.

DFU-0154 REC ID POS GREATER THAN RECORD LENGTH

An error exists in an RPG II specification. One of the record identification code positions (columns 21 through 24, columns 28 through 31, or columns 35 through 38 of the RPG II input specifications) contains a value that is greater than the record length.

DFU-0155 INVALID 'C/Z/D' ENTRY

An error exists in an RPG II specification. Positions 26, 33, and 40 of the RPG II input specifications must be blank or contain C, Z, or D.

DFU-0156 MORE THAN 8 RECORD ID CODES

An error exists in an RPG II input specification. DFU allows only eight record identification codes to identify a particular record type. These eight codes may be ANDed or ORed together in any combination.

DFU-0158 INVALID PACKED FIELD ENTRY – COL 43

An error exists in an RPG II input specification. Column 43 of the RPG II input specifications must be blank for an alphameric field and must be blank or P for a numeric field.

DFU-0159 FROM/TO NOT ALL NUMERIC – COLS 44-51

An error exists in an RPG II input specification. The From/To entry in the RPG II input specifications is invalid because the positions specified are not both numeric.

DFU-0160 INVALID DECIMAL ENTRY – COL 52

An error exists in an RPG II input specification. Decimal positions specified in column 52 of the RPG II input specification must be numeric and equal to or less than the field length.

DFU-0165 KEY END POS GREATER THAN RECORD LENGTH

An error exists in the RPG II file description specification. The end position of the specified key field is greater than the specified record length.

DFU-0166 KEY FIELD TOO LONG – COLS 29-30

An error exists in the RPG II file description specification. The key length (columns 29 and 30 in the RPG II file description specification) must be less than 30 positions for an unpacked key, and less than 9 positions for a packed key.

DFU-0171 INVALID NOT (N) CONDITION

An error exists in an RPG II input specification. Columns 25, 32, and 39 must contain an N or be blank.

DFU-0173 RECORD ID POSITIONS NOT NUMERIC

An error exists in an RPG II input specification. Columns 21 through 24, 28 through 31, and 35 through 38 in the RPG II record type specification must be numeric.

DFU-0175 FROM/TO POSITION GREATER THAN REC LENGTH

An error exists in an RPG II input specification. The field location in columns 44 through 51 of the RPG II file type specification is greater than the defined record length.

DFU-0176 FROM LOCATION GREATER THAN TO LOCATION

An error exists in an RPG II input specification. The From field location in columns 44 through 47 is not less than or equal to the To field in columns 48 through 51 of the RPG II field type specification.

DFU-0177 NUMERIC FIELD LENGTH OVER 15 POSITIONS

An error exists in an RPG II input specification. A numeric field length is greater than 15 positions. The difference between the From and To field locations must not be greater than 15 for all numeric fields.

DFU-0193 MISSING INPUT SPECIFICATIONS

An error exists in the RPG II source member. The RPG II source member must contain input specifications.

DFU-0257 FIELD ATTRIBUTES DO NOT MATCH MASTER KEY

You were requested to enter the name of the field from your list file to be used as the record key field to retrieve corresponding master file records (ENTER FIELD NAME FOR MASTER FILE KEY). The name you specified, however, did not have the same attributes as the key field for the master file (the lengths were different). The job is canceled.

When a DFU command is keyed, DFU generates, via procedures, the required OCL to process the job. Following is a list of the procedures used by DFU for the ENTER, UPDATE, INQUIRY, LIST, and SORT commands.

ENTER Command

```
*  5725-UT1 COPYRIGHT IBM CORP 1974
*  ENTER FILENAME,FORMAT,RPGNAME,#RECORDS,TYPE,DFUSPEC,LIBNAME
// MEMBER PROGRAM1-#DFUMSG1
// MEMBER USER1-#DFUMSG1
// IFF ?5'D'?/D IFF ?5?/S IFF ?5?/P #DFUTRM 0201
// IF ?5?/D IF DATAF1-?1R'0202'? #DFUTRM 0203
// IF  ?5?/S IF SOURCE-?1R'0202'? #DFUTRM 0203
// IF  ?5?/P IF PROC-?1R'0202'? #DFUTRM 0203
// IF ?2?/ * 0261
// IFF ?2R'0262'?/ IFF ?2?/#DFUOBJ IFF LOAD-?2? #DFUMP E,?2?,?3?,?5?,?6?,?7?
// ELSE IF ?2'#DFUOBJ'?/#DFUOBJ #DFUMP E,?2?,?3?,?5?,?6?,?7?
// IFF ?5?/D IF DATAF1-DFUDATAF DELETE DFUDATAF,F1
// #DFUEU N,?1?,?2?,?4?,?5?
// IFF ?5?/D #DFUDS ?1?,?5?
// IFF ?2?/#DFUOBJ RETURN
// LOAD $MAINT
// RUN
// DELETE NAME-#DFUOBJ,LIBRARY-O
// END
```

UPDATE Command

```
*  UPDATE FILENAME,FORMAT,RPGNAME,#RECORDS,TYPE,DFUSPEC,LIBNAME
// MEMBER PROGRAM1-#DFUMSG1
// MEMBER USER1-#DFUMSG1
// IFF ?5'D'?/D IFF ?5?/S IFF ?5?/P #DFUTRM 0201
// IF ?5?/D IFF DATAF1-?1R'0208'? #DFUTRM 0209
// IF  ?5?/S IFF SOURCE-?1R'0208'?  #DFUTRM 0209
// IF  ?5?/P IFF PROC-?1R'0208'?  #DFUTRM 0209
// IF ?2?/ * 0261
// IFF ?2R'0262'?/ IFF ?2?/#DFUOBJ IFF LOAD-?2? #DFUMP E,?2?,?3?,?5?,?6?,?7?
// ELSE IF ?2'#DFUOBJ'?/#DFUOBJ #DFUMP E,?2?,?3?,?5?,?6?,?7?
// IFF ?5?/D #DFUSD ?1?,?4?,?5?
// #DFUEU O,?1?,?2?,?4?,?5?
// IFF ?5?/D #DFUDS ?1?,?5?
// IFF ?2?/#DFUOBJ RETURN
// LOAD $MAINT
// RUN
// DELETE NAME-#DFUOBJ,LIBRARY-O
// END
```

INQUIRY Command

```
* INQUIRY FILENAME,FORMAT,RPGNAME,#RECORDS,TYPE,DFUSPEC,LIBNAME
// MEMBER PROGRAM1-#DFUMSG1
// MEMBER USER1-#DFUMSG1
// IFF ?5'D'?/D IFF ?5?/S IFF ?5?/P #DFUTRM 0201
// IF ?5?/D IFF DATAF1-?1R'0210'? #DFUTRM 0209
// IF ?5?/S IFF SOURCE-?1R'0210'? #DFUTRM 0209
// IF ?5?/P IFF PROC-?1R'0210'? #DFUTRM 0209
// IF ?2?/ * 0261
// IFF ?2R'0262'?/ IFF ?2?/#DFUOBJ IFF LOAD-?2? #DFUMP I,?2?,?3?,?5?,?6?,?7?
// ELSE IF ?2'#DFUOBJ'?/#DFUOBJ #DFUMP I,?2?,?3?,?5?,?6?,?7?
// IFF ?5?/D #DFUSD ?1?,?4?,?5?
// LOAD #DFUIN
// IF ?5?/D FILE NAME-DFUDATAF,LABEL-?1?,RETAIN-P
// IFF ?5?/D FILE NAME-DFUDATAF,RETAIN-S
// RUN
// DFU NAME-?1?,TYPE-D,FILE-?5?,OBJNAME-?2?
// END
// IFF ?2?/#DFUOBJ RETURN
// LOAD $MAINT
// RUN
// DELETE NAME-#DFUOBJ,LIBRARY-O
// END
```

LIST Command

```
*  LIST FILENAME,FORMAT,RPGNAME,SORT,TYPE,DFUSPEC,LIBNAME,MASTER NAME
// MEMBER PROGRAM1-#DFUMSG1
// MEMBER USER1-#DFUMSG1
// IFF ?5'D'?/D IFF ?5?/S IFF ?5?/P #DFUTRM 0201
// IF ?5?/D IFF DATAF1-?1R'0211'? #DFUTRM 0209
// IF ?5?/S IFF SOURCE-?1R'0211'? #DFUTRM 0209
// IF ?5?/P IFF PROC-?1R'0211'? #DFUTRM 0209
// IFF ?4?/ IFF ?4?/NOSORT IFF ?4?/SORT #DFUTRM 0212
// IFF ?5?/D IF ?4?/SORT #DFUTRM 0212
// IFF ?8?/ IFF DATAF1-?8? #DFUTRM 0290
// IF ?2?/ * 0261
// IFF ?2R'0262'?/ IFF ?2?/#DFUOBJ IFF LOAD-?2? #DFUMP L,?2?,?3?,?5?,?6?,?7?,?8?
// ELSE IF ?2'#DFUOBJ'?/#DFUOBJ #DFUMP L,?2?,?3?,?5?,?6?,?7?,?8?
// IF ?5?/D IF ?4?/ * 0273
// IF ?5?/D IFF ?4R'0274'?/ IFF ?4?/NOSORT IFF ?4?/SORT #DFUTRM 0212
// IF ?5?/D IF DATAF1-DFUTAG DELETE DFUTAG,F1
// IF  ?5?/D IF ?4'NOSORT'?/SORT #DFUSRT ?1?,?2?
// LOAD #DFULS
// IF ?5?/D FILE NAME-DFUDATAF,LABEL-?1?,RETAIN-P
// IF DATAF1-DFUTAG FILE NAME-DFUTAG,RETAIN-S
// IFF ?8?/ FILE NAME-DFUMAST1,LABEL-?8?,RETAIN-P
// RUN
// DFU NAME-?1?,TYPE-D,FILE-?5?,OBJNAME-?2?,MASTER1-?8?
// END
// IFF ?2?/#DFUOBJ RETURN
// LOAD $MAINT
// RUN
// DELETE NAME-#DFUOBJ,LIBRARY-O
// END
```

SORT Command

```
* SORT INPUT-FILENAME,SOURCE-FILENAME,OUTPUT-FILENAME,# RECORDS
// MEMBER PROGRAM1-#DFUMSG1
// MEMBER USER1-#DFUMSG1
// IFF DATAF1-?1R'0250'? #DFUTRM 0209
// IFF SOURCE-?2R'0251'? ENTER ??2?,#DFUSORT,,,S
// * 0219
// LOAD #GSORT
// FILE NAME-INPUT,LABEL-?1?
// IF DATAF1-?3R'0252'? #DFUTRM 0254
// FILE NAME-OUTPUT,LABEL-?3?,RECORDS-?4R'0253'?
// RUN
// SOURCE ??2?
```

ADDITIONAL PROCEDURES USED BY THE COMMANDS

#DFUMP

```
*  #DFUMP UTILITY,FORMAT,RPGNAME,TYPE,DFUSPEC,LIBNAME,MASTER NAME
// IFF ?5'NN'?/NN IFF ?5?/YY IFF ?5?/YN IFF ?5?/NY IFF ?5?/GO #DFUTRM 0205
// IFF ?5?/NN IF ?6?/   #DFUTRM 0214
// IF  ?5?/YN IFF SOURCE-?6? #DFUTRM 0206
// IF  ?5?/YY IFF SOURCE-?6? #DFUTRM 0206
// IF  ?5?/GO IFF SOURCE-?6? #DFUTRM 0206
// IF  ?5?/NY IF  SOURCE-?6? #DFUTRM 0214
// IFF SOURCE-?3R'0215'? #DFUTRM 0216
// LOAD #DFUMP
// RUN
// DFU  UTIL-?1?,FILE-?4?,CATL-?5?,LIBNAME-?6?,RPGNAME-?3?,OBJNAME-?2?,MASTER1-?7?
// END
```

#DFUDS

```
*  #DFUDS FILENAME,TYPE
// * 0218
// LOAD #DFUDS
// FILE NAME-DFUDATAF,RETAIN-S
// RUN
// DFU NAME-?1?,TYPE-0,FILE-??2?
// END
```

#DFUSD

```
*  #DFUSD FILENAME,#RECORDS,TYPE
// * 0217
// IF DATAF1-DFUDATAF DELETE DFUDATAF,F1
// LOAD #DFUSD
// FILE NAME-DFUDATAF,RETAIN-T,RECORDS-?2'500'?
// RUN
// DFU NAME-?1?,TYPE-N,FILE-?3?
// END
```

#DFUEU

```
*  #DFUEU STATUS,FILENAME,FORMAT,#RECORDS,TYPE
// LOAD #DFUEU
// FILE NAME-DFUDATAF,
// IFF ?5?/D IF ?1?/D RETAIN-T
// IFF ?5?/D IF ?1?/N RETAIN-T,RECORDS-?4'500'?
// IF ?5?/D IF ?1?/D LABEL-?2?,RETAIN-P
// IF ?5?/D IF ?1?/N LABEL-?2?,RETAIN-P,RECORDS-?4R'0207'?
// RUN
// DFU NAME-?2?,TYPE-?1?,FILE-?5?,OBJNAME-?3?
// END
```

#DFUSRT

```
*  #DFUSRT FILENAME,FORMAT
// * 0219
// LOAD #DFUSB
// RUN
// DFU OBJNAME-?2?
// END
// LOAD #GSORT
// FILE NAME-INPUT,LABEL-?1?
// FILE NAME-OUTPUT,LABEL-DFUTAG,RETAIN-T,BLOCKS-10
// RUN
// SOURCE #DFUSRT
// LOAD $MAINT
// RUN
// DELETE NAME-#DFUSRT,LIBRARY-S
// END
```

#DFUTRM

```
*  #DFUTRM MIC
// * 'DFU ?1?'
// * ?1?
// IFF ?1?/  CANCEL
```

alphabetic character: Letters A through Z and special symbols #, $, and @.

auto dup indicator: The presence or absence of an A in position 34 of line 1 on the display screen identifying the status of the auto dup feature.

automatic field duplication: A DFU feature that allows one or more fields to be copied from one record to another.

automatic key generation: A DFU feature of assigning 5-digit keys to the records of a file.

batch accumulator: An area on disk where subtotals for a field are kept.

command key: One of the top rows of data keys on the keyboard (numeric, -, =) when used in conjunction with the CMD function key. These keys allow functions that are not provided by the function keys defined for the system. The command key functions are identified by a template inserted directly above the keys.

command statement: A statement that is used to request the performance of a particular function. It always contains the command name and may include parameters. Specifically, a command statement is a special form of the // INCLUDE OCL statement. A command statement evokes a procedure and can pass information to the procedure via parameters included in the statement. The procedure named by the command name is evoked by the command statement.

control field: A field within a record that identifies the record's relationship to other records (such as a part number in an inventory record). Control fields are compared from record to record to determine when certain operations are to be performed.

constant: A data item that does not change during the execution of a program.

cursor: A dash of light on the display screen that indicates where the next character will be entered.

data: A collection of facts, numbers, letters, and symbols that can be processed by a computer.

delete code: Character that identifies an inactive record in a file.

DFU attributes: A group of 40 character records, each record having 5 eight position fields, that is created from the RPG II specifications when the DFU job is being set up.

DFU specifications: A group of 40 character records, each record having 5 eight-position fields, that is created from the responses to prompts when the DFU job is being set up.

display screen: The screen on which the system displays data, messages, and other information for the operator.

edit: To punctuate a field by suppressing zeros and inserting commas, decimal points, dollar signs, or other constant information.

factor: A field name or constant used in a calculation operation.

field: One or more adjacent record positions that contain related information.

field accumulation: A DFU feature that enables one or more fields to be totaled during processing.

file: An organized collection of related records.

filename: The name associated with a file.

format description: The end result of the setup step. It describes your file and exactly how you want DFU to process it. DFU creates it by combining information from the DFU attributes and DFU specifications. It is stored in the library as a load member with the name you specify.

function keys: Special keys on the keyboard used to request specific system functions.

indexed file: A file in which the position of the records is recorded in a separate portion of the file called an index. The index contains a record key and disk address for each record in the file.

library: An area on the disk that contains procedure members, source members, load members, and subroutine members as well as areas required by the system control program.

library member: A named collection of statements or records in the library that can contain source statements, format descriptions, OCL statements, utility control statements, or executable instructions.

list file: A data file from which information is extracted to print a report.

load member: A collection of instructions, stored in the library, that the system can execute to perform a particular function whether the function is requested by the operator or specified in an OCL statement.

master file key name: The name of the field in a list file which is used to retrieve records from the master file.

modulus 10 and 11: Formulas that you use to calculate a self-check digit for a self-check field and DFU uses to verify the self-check digit.

null response: A response made by pressing the ENTER key without keying any information.

operation code: A description of how a field is to be used in a DFU job (as an accumulator, self-check field, or an auto dup field). The operation is specified in the prompting sequence when defining the DFU job.

parameter: The values specified in a command statement.

procedure: A named collection of related OCL statements, and possibly, utility control statements.

procedure member: A procedure stored in the library.

prompt: A statement appearing on the display screen that aids the user by requesting a particular kind of response in order to continue processing.

record: A collection of related data, treated as a unit. For example, one line of an invoice may form a record. A complete set of records may form a file.

record identification indicator: A code placed in a record to identify a record type.

record key: (1) One or more characters within an item of data that are used to identify the data. (2) When running DFU, the record key is entered to obtain the record to be processed. When entering/updating records, either you or the DFU program can create the record key.

record types: Records from one file that have different fields and/or format.

related master file: A data file from which information is extracted, and combined with the information available in the list file, to print a report.

result field: The name of a field in which the outcome of arithmetic calculations are stored.

right-justify: The placement of data in a field with the last significant digit in the rightmost position.

RPG II source member: RPG II file description and input specifications that describe the file to be processed by DFU

select field: A field which is tested for a condition to determine whether a record contains information that should be printed in a report.

selection criteria: Conditions that must be met when determining whether a record contains information that should be printed in a report.

self-check field: Fields that consist of a base number and a check digit. DFU verifies a self-check field as you key it by calculating the check digit from the base number and comparing the result with the check digit you keyed.

source member: A collection of records (such as RPG II specifications or sort sequence specifications) that are used as input for a program. Source members are stored in the library.

total accumulator: An area on disk where final totals for a field are kept.

International Business Machines Corporation

General Systems Division
4111 Northside Parkway N.W.
P.O. Box 2150
Atlanta, Georgia 30301
(U.S.A. only)

General Business Group/International
44 South Broadway
White Plains, New York 10601
U.S.A.
(International)

IBM System/32 Utilities Program Product Reference Manual Data File Utility (File No. S32-32) Printed in U.S.A. SC21-7600-3