File Manager for z/OS V6R1

# Addenda to V6R1 User's Guides and Customization Guide

File Manager for z/OS V6R1

# Addenda to V6R1 User's Guides and Customization Guide

# Contents

# About this document

This document provides details of all the APAR service fixes that impact upon documentation, for IBM File Manager for z/OS Version 6.1, since the most recent edition of the product manuals in May 2006. These editions are:

- User's Guide and Reference (SC18-9655-01) - Second Edition
- User's Guide and Reference for DB2 Data (SC18-9656-01) - Second Edition
- User's Guide and Reference for IMS Data (SC18-9657-01) - Second Edition
- Customization Guide (SC18-9650-01) - Second Edition

The Addendum document is divided into two parts:

- **Part One: PTF/APAR Documentation Changes**

  This section lists the changes to the File Manager for z/OS Version 6 Release 1 documentation that are required to reflect new behavior resulting from the application of APAR fixes.

  The fixes are listed by PTF number, in reverse date order, so that the most recently released fix appears at the beginning of the document. Each description shows:

  - The set of PTF numbers in the release
  - The date of the PTF release
  - The APARs included in the released fix
  - Details of those APAR changes that affect documentation
  - Page references for the manuals affected by the change

  **Notes:**

  1. This document does NOT describe those APAR fixes that do not have an impact upon documentation.
  2. The enhancements and corrections described in this section are only available after applying the listed PTFs for the APAR.

- **General Documentation Changes**

  This section describes enhancements, corrections and updates in the documentation for File Manager for z/OS Version 6 Release 1. These changes are not associated with PTF numbers, as they do not require the application of any code updates.

  The changes are grouped by manual and listed within each section in reverse date order. That is, the most recent documentation change appears at the beginning of each manual section.

# Part 1. PTF/APAR Documentation Changes

# UK37309, UK20628, UK20629, UK20631, UK20634, UK20636

Release Date: **26 June 2008**

This set of PTFs contains the following APAR fixes:

| APAR # | APAR Abstract | Doc Impact |
|---|---|---|
| PK67188 | File Manager FCH PTF UK35982 does not fix the reported problem. | User's Guide and Reference (SC18-9655-01) |
| PK67923 | AE PK67188 FIX COMPLETION. | None. |

## PK67188

**Initial problem description**

1. When using the Find/Change utility to change JCL and a trailing space is specified in the CHANGE command, no match will be found because trailing spaces are not included in the logical JCL statement built by File Manager.

2. The Find/Change utility does not recognize the VCONTEXT command when the JCL option has been selected.

**Outline of solution**

1. Trailing spaces will be include in the logical JCL statement against which the Find parameter is compared.

2. The VCONTEXT command parameters will be taken into consideration when producing the Find/Change utility report and the LIST=LONG and JCL options have been specified.

**Documentation impact**

This APAR requires changes to be made to:
• User's Guide and Reference (SC18-9655-01)

### Changes to the User's Guide and Reference

#### Chapter 7, ″**Managing data sets**″

In the section, ″Working with files or members containing JCL″, change the definition for *Parameter part* to:

*Parameter part*

Contains the contents of the first character after the *name part*, including any blanks after the last character of the *parameter part*.

**PK67188**

# UK21447, UK21448, UK21450, UK21452

Release Date: **29 January 2007**

This set of PTFs contains the following APAR fixes:

| APAR # | APAR Abstract | Doc Impact |
|---|---|---|
| PK28860 | FMN1OPT and/or the FM settings do not allow more than 3 RESLIB definitions. | Customization Guide (SC18-9650-01) User's Guide and Reference for IMS Data (SC18-9657-01) |
| PK32055 | ABEND0C1 using sample COBOL security exit FMN1SXT for File Manager. | None. |
| PK35829 | File Manager getting 'ICE024A 1 RECORD LENGTH VALUE ERROR' message. | None. |
| PK35854 | FM V6.1 abend 0C4-4 during find loadmodule. | Customization Guide (SC18-9650-01) |
| PK36554 | File Manager IMS did not get the correct MACLIB. | None. |
| PK37368 | ABEND0C4 in batch compare with new dsn empty. | None. |
| PK37634 | FM/IMS TEMPHLQ value ignored for temporary dataset used by extract. | None. |
| PK37840 | Printing an audit report using SMF input from a tape is not working. | None. |

## PK28860

**Initial problem description**
> FMN1OPT and/or the FM settings do not allow more than 3 RESLIB definitions.

**Outline of solution**
> FM/IMS has been modified to support up to 6 IMS RESLIBs.

**Documentation impact**
> This APAR requires changes to be made to:
> - Customization Guide (SC18-9650-01)
> - User's Guide and Reference for IMS Data (SC18-9657-01)

## Changes to the Customization Guide

### Chapter 18, Customizing FM/IMS

In the section: "Customizing to use File Manager/IMS in DLI mode", change the first bullet point, to read:
- The IMS RESLIB load library names, that is, the fully qualified data set names as specified on the DFSRESLB DD for a DLI batch job. Supply these names in the RESLIB1 option (and also the RESLIB2. RESLIB3, RESLIB4, RESLIB5, and RESLIB6 options, if required), in FMN1POPD.

In the section: "Customizing to use File Manager/IMS in BMP mode", change the second bullet point, to read:
- The IMS RESLIB load library names used by the dependent region. Supply these names in the RESLIB1 option (and also the RESLIB2, RESLIB3, RESLIB4, RESLIB5, and RESLIB6 options, if required), in FMN1POPI.

In section: "Examples of FMN1POPI and FMN1AGNT macros", change example IF22 to read:

```
FMN1POPI SSID=IF22,                                          +    IF22
        DESC='IMS Version 7.1 FM/IMS TEST System',           +
        RESLIB1='IMSV710.IF22.SDFSRESL',                     +
        RESLIB2='IMS.V7R1M0.SDFSRESL',                       +
        RESLIB3='IMSV710.IF22.USERLIB',                      +
        RESLIB4='IMS.V7R1M0.RESLIB',                         +
        RESLIB5='IMSV810.IF22.SDFSRESL',                     +
        RESLIB6='IMSV810.IF22.USERLIB',                      +
        DYNACB='IMSV710.IF22.DOPTLIB',                       +
        DYNPRFX=FMN0,                                        +
        DYNPRFN=199,                                         +
        MACLIB='IMSV710.IF12.MACLIB'
```

Change note 2 under IF22 to read:
- The six RESLIB keywords specify six IMS RESLIBs to be used by File Manager/IMS in BMP mode.

In the section: "Changing the default options", figure 11, parts 2 and 3, (FMN1POPD and FMN1POPI), add three new options after RESLIB3 in each case:

```
        RESLIB4=,                                            X
        RESLIB5=,                                            X
        RESLIB6=,                                            X
```

### Chapter 22, Verifying the customization of File Manager/IMS

In Step 5, Set DLI data set options, change figure 14 to read:

```
  Process   Options   Help
 ────────────────────────────────────────────────────────────────
 FM/IMS                    DLI Mode Data Sets
 DFSVSAMP:
    Data set name . . 'IMSVS.WA.V710.PROCLIB'
    Member  . . . . . DFSVSMDB
 RESLIB:
    Data set name #1  'IMSVS.WA.V710.SDFSRESL'
    Data set name #2  'IMSVS.WA.V710.USERLIB'
    Data set name #3
    Data set name #4
    Data set name #5
    Data set name #6
 IMS Macros:
    Data set name . . 'IMSV710.IF22.SDFSMAC'
 RECON:
    Primary Data set  'IMSVS.WA.V710.RECON1'
    Secondary . . . . 'IMSVS.WA.V710.RECON2'
    Spare . . . . . . 'IMSVS.WA.V710.RECON3'


 Command ===> _____
```

*Figure 1. DLI Mode Data Sets*

## Appendix C, File Manager/IMS options

Add the following new File Manager/IMS option to appendix C, FMN1POPD,
where RESLIBn is RESLIB4, RESLIB5 and RESLIB6:
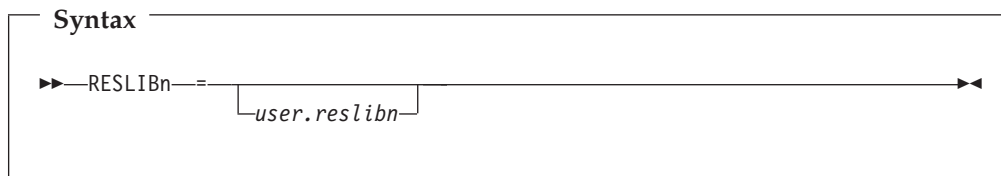
**RESLIBn:**

> **Syntax**
>
> ►►──RESLIBn──=──────────────────────────────────►◄
>         └─*user.reslibn*─┘

**RESLIBn**      specifies an additional user-defined library, if required, for File
Manager/IMS in DLI mode: for example, a library containing
dynamic allocation or user exit routines. If an additional library is
not required, leave this option blank.

Add the following new File Manager/IMS option to appendix C, FMN1POPI,
where RESLIBn is RESLIB4, RESLIB5 and RESLIB6:
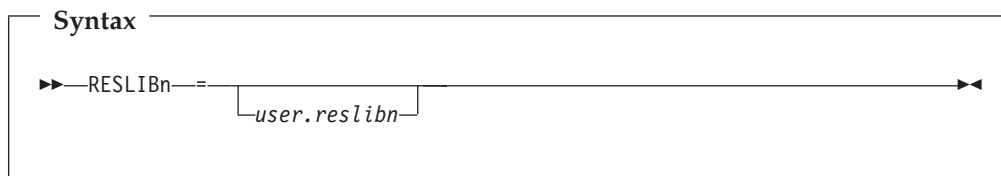
**RESLIBn:**

> **Syntax**
>
> ►►──RESLIBn──=──────────────────────────────────►◄
>         └─*user.reslibn*─┘

**RESLIBn**      specifies an additional user-defined library, if required, for File
Manager/IMS in BMP mode: for example, a library containing

> dynamic allocation or user exit routines. If an additional library is not required, leave this option blank.

## Changes to the User's Guide and Reference for IMS Data

### Chapter 9, Panels and Fields
In section, "DLI Mode Data Sets panel", update the DLI Mode Data Sets panel:

Under the RESLIB header, add entries for the new fields: Data set name #4, Data set name #5, and Data set name #6. Text should be the same as that for Data set name #3.

### Chapter 11, Functions
In sections "IEB (IMS Edit Batch)", "ILB (IMS Load)", "IPR (IMS Print)" and "IXB (IMS Extract)", add optional parameters:
>     RESLIB4=reslib4
>     RESLIB5=reslib5
>     RESLIB6=reslib6

to the DLI parameter section of the syntax diagrams.

In section "IEB (IMS Edit Batch)", change the entry header:

RESLIB2=reslib2/RESLIB3=reslib3

to:

RESLIB2=reslib2/RESLIB3=reslib3/RESLIB4=reslib4/RESLIB53=reslib5/ RESLIB6=reslib6

---

# PK35854

**Initial problem description**
> 1. ABENDS0C4 abend S0C4 in FMNEDJPN may be reported when performing a FIND command in a member list (for example LOAD MOD MEMBER SELECTION 3.10)
>
> 2. Users in DBCS-enabled environment are unable to enter lower-case File Manager commands.

**Outline of solution**
> 1.Program logic has changed to make sure that the "pair table" is always allocated before creating FIND argument pairs for DBCS processing.
>
> 2. File Manager is being enhanced to allow for a user-customizable command verbs and command keywords translation table and allowing lower case command verb and keyword input in DBCS environments.

**Documentation impact**
> This APAR requires changes to be made to:
> • Customization Guide (SC18-9650-01)

## Changes to the Customization Guide

### Chapter 3, Customizing File Manager
In the section: "Changing the print and display tables":
• Remove the fourth paragraph.
• Add the following new translate table to the list, before TRUP.

TRUPC        used to translate lower case characters to uppercase characters for File Manager command verbs and command keywords only. This table generally does not need to be customized in an SBCS environment.

## Chapter 6, Customizing File Manager for national languages

In the section: ″Changing the print and display translation tables for languages other than English″, add the following paragraph to step 4.b:

File Manager provides support under ISPF for command verbs and command keywords entered in lower or uppercase as long as they are a valid SBCS-only strings and the a - z codepoints in the host codepage used correspond to the standard EBCDIC a-z codepoints. If this is not the case, and you want to be able to enter lower or mixed case command verbs and command keywords, then the TRUPC translation table in FMNTRTBS needs to be modified to specify the uppercase A - Z EBCDIC codepoints at locations corresponding to the a - z codepoints in the codepage used.

Add the same paragraph to step 2, in the section: ″Customizing for the Japanese national language - Modifying the Japanese translation tables″.

**PK35854**

# UK20626, UK20628, UK20629, UK20631, UK20634, UK20636

Release Date: **20 December 2006**

This set of PTFs contains the following APAR fixes:

| APAR # | APAR Abstract | Doc Impact |
|---|---|---|
| PK31454 | File Manager IMS - setting 0.10 volume nonsms not saved. | None. |
| PK31588 | Device type ignored for output data set allocation: field "unit type" in panel 0.10 allocate with UNIT=SYSALLDA hardcoded. | Customization Guide (SC18-9650-01) |
| PK34144 | System abend code 0C4 IN DB2/FM. | None. |
| PK34379 | FM/IMS cannot handle database records with more than 32766 segments (sysroute of PK20161). | None. |
| PK34692 | File Manager job hangs if no ISPF stats. | None. |
| PK34895 | FM/DB2 edit delete row with varchar greater than 4000 bytes gets sqlcode -313 (with DB2 V8). | None. |
| PK35200 | Problems browsing a segment in SNGL mode (sysroute of PK28214). | None. |
| PK35356 | FM/DB2 gen stmt: SQL statements are generated for all the indexes in the database instead of only one. | None. |
| PK35498 | MSGFMN0357I inconsistent block length, using DSP and reading a multi-volume VB tape file. | None. |
| PK35507 | FCH delete JCL records with JCL=YES option (sysroute of PK33455). | User's Guide and Reference (SC18-9655-01) |
| PK35529 | ABEND0C4 in selection criteria screen (sysroute of PK34146). | None. |
| PK35530 | FM/IMS delims with CKEY does not work as expected (sysroute of PK32940). | None. |
| PK35536 | ABEND0C4 in FMNEDIT2 doing repeat find (sysroute of PK35291). | None. |

| APAR # | APAR Abstract | Doc Impact |
|---|---|---|
| PK35582 | File Manager reformat output incorrect. | None. |

# PK31588

**Initial problem description**
MSGIKJ56893I, MSGIGD01008I, and MSGIGD01001I produced for allocation errors with the SYSLOUT data set due to unit SYSALLDA being used.

**Outline of solution**
File Manager has been changed to allow you to customize the unit, data, storage and management classes for allocation of temporary and various permanent data sets. The changes apply at the system and user level.

**Documentation impact**
This APAR requires changes to be made to:
- Customization Guide (SC18-9650-01)

## Changes to the Customization Guide

### Chapter 3, Customizing File Manager
In chapter 3, Customizing File Manager, page 15, in figure 2, after

```
          USEIOX=DISABLE,           or (ENABLE,name) if enabled    X
```

add the following 8 keywords:

```
          TUNIT=SYSALLDA,           or other temp unit name        X
          TDATAC=,                  or sms data class              X
          TSTORC=,                  or sms storage class           X
          TMGMTC=,                  or sms management class        X
          PUNIT=SYSALLDA,           or other permanent unit        X
          PDATAC=,                  or sms data class              X
          PSTORC=,                  or sms storage class           X
          PMGMTC=,                  or sms management class        X
```

### Appendix A, File Manager options
Add the following new File Manager options to appendix A:

**TUNIT:**

```
  ┌─ Syntax ──────────────────────────────────────────────────────────┐
  │                                                                    │
  │                        ┌─SYSALLDA─┐                                │
  │  ►►──TUNIT──=──────────┼──────────┼──────────────────────────►◄    │
  │                        └─xxxxxxxx─┘                                │
  │                                                                    │
  └────────────────────────────────────────────────────────────────────┘
```
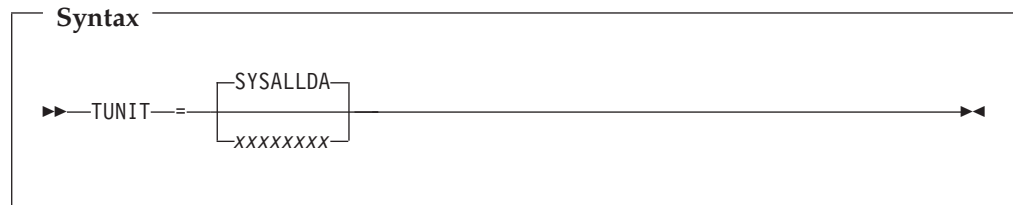
**TUNIT**        specifies the default temporary unit that File Manager will use when allocating work data sets for internal use.

The default is SYSALLDA. The maximum length you can specify for TUNIT is eight (8) bytes.

**TDATAC:**
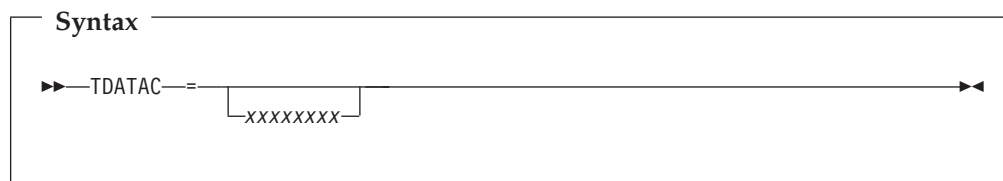
```
┌─ Syntax ─────────────────────────────────────────────────────────────┐
│                                                                       │
│  ►►──TDATAC──=─┬──────────────┬────────────────────────────────►◄     │
│               └─xxxxxxxx─┘                                             │
│                                                                       │
└───────────────────────────────────────────────────────────────────────┘
```

**TDATAC**

> specifies the default temporary SMS data class that File Manager will use when allocating work data sets for internal use.

TDATAC is optional. The maximum length you can specify for TDATAC is eight (8) bytes.

**TSTORC:**

```
┌─ Syntax ─────────────────────────────────────────────────────────────┐
│                                                                       │
│  ►►──TSTORC──=─┬──────────────┬────────────────────────────────►◄     │
│               └─xxxxxxxx─┘                                             │
│                                                                       │
└───────────────────────────────────────────────────────────────────────┘
```
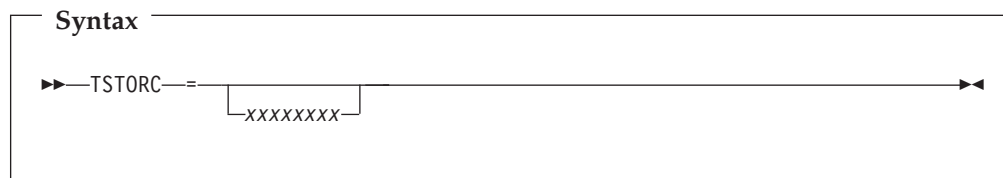
**TSTORC**

> specifies the default temporary SMS storage class that File Manager will use when allocating work data sets for internal use.

TSTORC is optional. The maximum length you can specify for TSTORC is eight (8) bytes.

**TMGMTC:**

```
┌─ Syntax ─────────────────────────────────────────────────────────────┐
│                                                                       │
│  ►►──TMGMTC──=─┬──────────────┬────────────────────────────────►◄     │
│               └─xxxxxxxx─┘                                             │
│                                                                       │
└───────────────────────────────────────────────────────────────────────┘
```

**TMGMTC**

> specifies the default temporary SMS management class that File Manager will use when allocating work data sets for internal use.

TMGMTC is optional. The maximum length you can specify for TMGMTC is eight (8) bytes.

**PUNIT:**

```
┌─ Syntax ────────────────────────────────────────────────────────────┐
│                      ┌─SYSALLDA─┐                                     │
│  ▶▶──PUNIT──=─────────┼──────────┼──────────────────────────────▶◀   │
│                      └─xxxxxxxx─┘                                     │
│                                                                      │
└──────────────────────────────────────────────────────────────────────┘
```

**PUNIT**        specifies the default permanent unit that File Manager will use when allocating permanent data sets. (eg. print file).

The default is SYSALLDA. The maximum length you can specify for PUNIT is eight (8) bytes.

**PDATAC:**

```
┌─ Syntax ────────────────────────────────────────────────────────────┐
│                                                                      │
│  ▶▶──PDATAC──=───┬──────────┬─────────────────────────────────▶◀     │
│                  └─xxxxxxxx─┘                                         │
│                                                                      │
└──────────────────────────────────────────────────────────────────────┘
```

**PDATAC**
        specifies the default temporary SMS data class that File Manager will use when allocating work data sets for internal use.
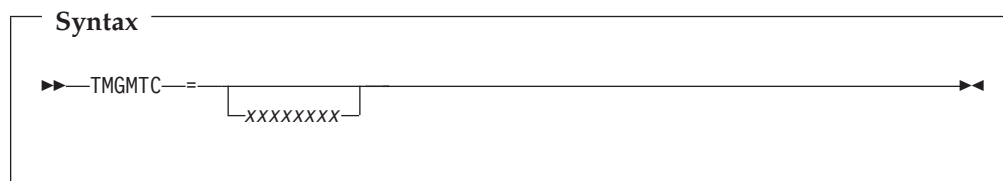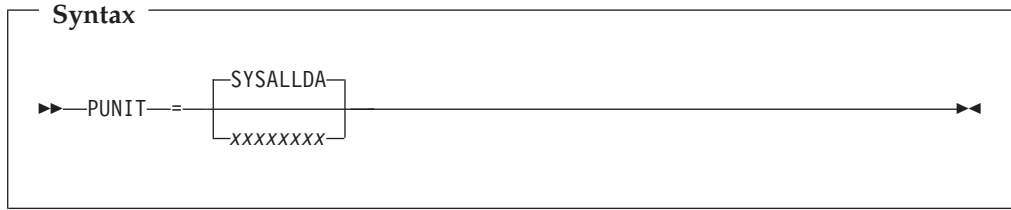
PDATAC is optional. The maximum length you can specify for PDATAC is eight (8) bytes.

**PSTORC:**

```
┌─ Syntax ────────────────────────────────────────────────────────────┐
│                                                                      │
│  ▶▶──PSTORC──=───┬──────────┬─────────────────────────────────▶◀     │
│                  └─xxxxxxxx─┘                                         │
│                                                                      │
└──────────────────────────────────────────────────────────────────────┘
```

**PSTORC**
        specifies the default permanent SMS storage class that File Manager will use when allocating permanent data sets. (eg. print file).

PSTORC is optional. The maximum length you can specify for PSTORC is eight (8) bytes.

**PMGMTC:**

```
┌─ Syntax ────────────────────────────────────────────────────────────┐
│                                                                      │
│  ▶▶──PMGMTC──=───┬──────────┬─────────────────────────────────▶◀     │
│                  └─xxxxxxxx─┘                                         │
│                                                                      │
└──────────────────────────────────────────────────────────────────────┘
```

**PMGMTC**
>
> specifies the default permanent SMS management class that File Manager will use when allocating permanent data sets. (eg. print file).

PMGMTC is optional. The maximum length you can specify for PMGMTC is eight (8) bytes.
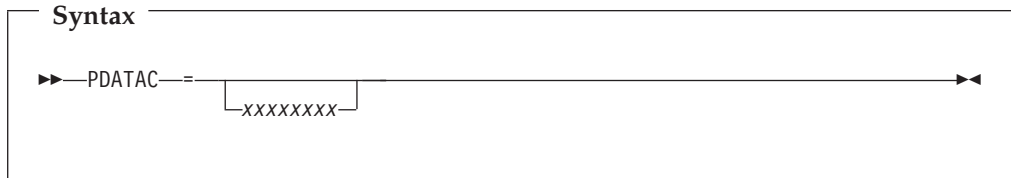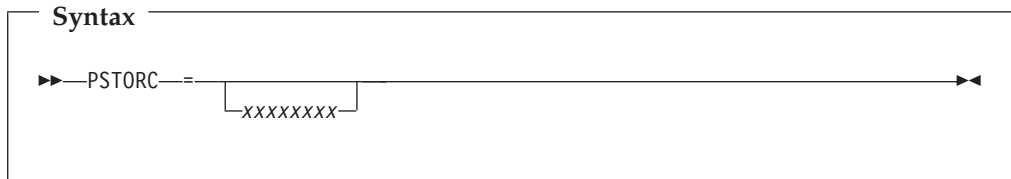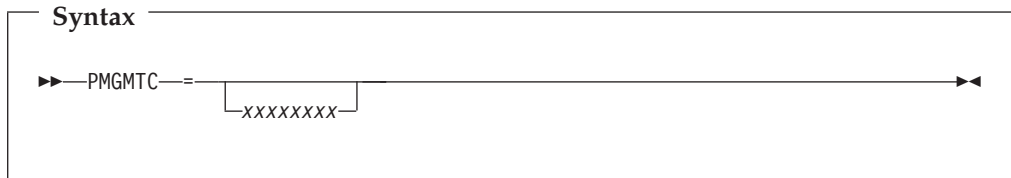
# PK35507

**Initial problem description**
>
> 1.Error "FMN0355I Record size (80) invalid for FIXED,90 output" may be reported and changed member corrupted.
>
> 2. When creating a template from PLI copybooks where the template is created using multiple copybooks and 01 field names are supplied for each of the copybooks, some supplied field names are ignored and an asterisk (*) is substituted (PK33914).
>
> 3. FM job may abend with ABENDS978, abend S978 (PK35058).
>
> 4. ABENDS0C4 in IRXINIT, ABENDS978 in ISPF, ABENDS0C4, ABEND S0C4, or unpredictable results after specifying template criteria using the by field panels and RG operators (PK33920).
>
> 5. Help panel FMNR317I missing error.
>
> 6. "Invalid value" error may be reported when updating BIT or AX fields in TABL view with HEX ON (PK34944).

**Outline of solution**
>
> 1.Program logic has been corrected to allow for JCL records to be upto 160 bytes, regardless of PAD setting. Only data in columns 1 - 72 is considered to be JCL statements, data past column 72 will be treated as non-changeable sequence numbers/comments. Note that when change causes a JCL record to spill over and cause a continuation record to be created then any data in columns 72-end-of-record will be copied from the original record to the newly inserted record.
>
> For all FM components, the SET function has been enhanced to allow for a change of the TEMPHLQ value dynamically in batch processing (for the duration of the batch job).
>
> 2. When creating a template from PLI copybooks where the template is created using multiple copybooks and 01 field names are supplied for each of the copybooks, some supplied field names will be used.
>
> 3 to 7. Program logic has been corrected.
>
> 4. The storage overlay has been corrected.
>
> 6. Panel FMNR317I has been reshipped with the product.

**Documentation impact**
>
> This APAR requires changes to be made to:
> • User's Guide and Reference (SC18-9655-01)

## Changes to the User's Guide and Reference

### Chapter 17, "Functions"

In the section, "FCH (Find/Change)", in the description for the JCL=YES option, replace the following text:

"The data in columns 73-80 for new physical records is copied from the last related original physical record."

with

"The data in columns 73-record length for new physical records is copied from the last related original physical record. Data past column 72 will be treated as non-changeable sequence numbers/comments."

## Chapter 17, "**Functions**"

In the section, "SET (Set Processing Options)", add the following to the syntax diagram:

```
►►─┬───────────────────────┬──────────────────────────►◄
   └─TEMPHLQ=temp_hlq─┘
```
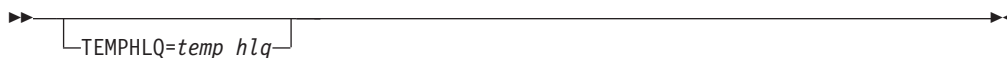
In the descriptions following the syntax diagram, add the following:

**TEMPHLQ**

TEMPHLQ specifies an installation-specific high-level qualifier for the temporary data sets created during a File Manager session. You can specify any valid qualifier, including multilevel if required, up to a total length of 24 characters. If you specify &USER. for any part of the qualifier, this is replaced with the user's userid. A multilevel qualifier with symbols follows the same rules as in the specification of a data set name with symbolic parameters in JCL procedures. Thus, a single period used after a symbol combines the symbol with the text following it (See Example 1 below); a period separating parts of a multilevel qualifier must be doubled (see Example 2 below).

**Example 1**

If the userid is XXXX, then specifying &USER.1.TEMP results in a high-level-qualifier of XXX1.TEMP

**Example 2**

If the userid is ZZZZ then specifying FMN.&USER..ABCD results in a high-level-qualifier of FMN.ZZZZ.ABCD

**Notes:**

1. The TEMPHLQ parameter can be up to 24 characters long. However, temporary data sets can have different lengths, and the system part of the data set name may be longer than 20 characters. In this case, minor levels of TEMPHLQ are ignored, to the extent that the final data set name is less than or equal to 44 characters.

2. Concatenation of &USER.string could result in a part of the high-level qualifier being longer than 8 characters. In this case, the string is shortened to the extent that the final part of the qualifier is 8 characters long.

# UK19267, UK19268, UK19269, UK19270, UK19271

Release Date: **6 November 2006**

This set of PTFs contains the following APAR fixes:

| APAR # | APAR Abstract | Doc Impact |
|---|---|---|
| PK25408 | File Manager should display message instead of ABENDU0999 when problem allocating temporary dataset. | None. |
| PK29961 | File Manager may go into a loop following a DB2 error. | None. |
| PK30501 | Problem with the print utility. | None. |
| PK31568 | ABEND0C4 doing a compare of file with COBOL array. | None. |
| PK31570 | ABEND0C4 in FMNCADAT when executing 3.3 COPY using copybooks. | None. |
| PK32120 | FCH batch update of a PDSE member with JCL=YES and DIRINTEGR=YES gets "member in use" message and/or records are not changed. | None. |
| PK32493 | Using 'EDIT DATA PANEL OPTION' on a dataset with LRECL greater than 32752 results in an LRECL change to 32756. | None. |
| PK32622 | A compare with File Manager V6.1 using COBOL copybooks gets an ABEND0C4. | None. |
| PK33041 | File Manager DSC batch records truncated but job gets RC=0 | Customization Guide (SC18-9650-01) |

## PK33041

**Initial problem description**

DSC may copy records truncating them (because of record length difference between input and output). No specific return code value is associated with such condition and value of 0 is being returned.

**Outline of solution**

File Manager has been enhanced to provide a customizable return code for the conditions of dsc_rectrc (record truncation occurred) and dsc_fldtrc (field truncation occurred). These return codes default to 0 (to match

existing behavior) and can be user-customized using the procedures
described in the File Manager Customization Guide.

**Documentation impact**

This APAR requires changes to be made to:

- Customization Guide (SC18-9650-01)

# Changes to the Customization Guide

## Chapter 3, Customizing File Manager

In chapter 3, Customizing File Manager, add the following new customizable
return codes to figure 3 on page 19, for the DSC function:

```
RECTRC=0,        Records truncated              X
FLDTRC=0,        Fields  truncated              X
```

# UK18127, UK18130, UK18131, UK18141, UK18159, UK18162

Release Date: **28 September 2006**

This set of PTFs contains the following APAR fixes:

| APAR # | APAR Abstract | Doc Impact |
|---|---|---|
| PK24350 | System abend code 0C4 in File Manager DB2. | Customization Guide (SC18-9650-01) |
| PK26558 | Paging and scrolling problems in File Manager IMS. | None. |
| PK26868 | In the primary menu panel FMN2ST00, the SQL-ID can be specified and should be saved. | None. |
| PK27732 | File Manager DB2 toleration support for DB2 V9. | None. |
| PK28780 | When the SQL command is used for programs compiled after switching to new function mode, the output is not readable (sysroute of PK21520). | None. |
| PK29853 | File Manager V6.1 "Batch Compare" shows old record 068031 as deleted but it exists on new as 068063. | User's Guide and Reference (SC18-9655-01). |

## PK24350

**Initial problem description**
AbendS0C4 may be received when running the FMN2GEN utility.
Additional symptom: IBM0534S ONCODE=8094 Protection exception from compile unit ADB2GEN at entry point GEN_INDEX.

**Outline of solution**
File Manager, DB2 component has been updated to correct the problem.

**Documentation impact**
This APAR requires changes to be made to:
- Customization Guide (SC18-9650-01)

## Changes to the Customization Guide

### Part 2. Customizing File Manager/DB2 Component
With the application of PTF UK18131, the following additions should be made to the Customization Guide, in Part 2. "Customizing File Manager DB2 Component".

In Table 11, "Summary of steps for customizing File Manager/DB2 and the operating environment", add the following after step 9:

- Create a segmented table space in a TEMP database - **required**.

**19**

In Chapter 12, Customizing File Manager/DB2, add the following new DBRMs to the list in the section "Binding DB2 (**required**)":

**DBRM**       **ALIAS**
**FMN2ARM**    ADB2REM
**FMN2ARP**    ADB2REP
**FMN2ARR**    ADB2RER
**FMN2ARS**    ADB2RES

In Chapter 12, Customizing File Manager/DB2, add the following section after the section "Binding DB2 (**required**)":

**Creating a segmented table space (required):**   File Manager/DB2 uses DECLARE GLOBAL TEMPORARY TABLE statements to implement some features of reverse engineering. Each DB2 subsystem that runs File Manager/DB2 must have a database that is defined as TEMP that contains a segmented table space to support the use of declared temporary tables. The page size of the segmented table space must be at least 8 KB.

After File Manager/DB2 is installed, you must create a database that is defined as TEMP and a segmented table space in this database on each DB2 subsystem that runs File Manager/DB2, and that is not already set up to support the use of declared temporary tables.

A sample job, FMN2CGEN, is provided in FMN.SFMNSAM1, to help you do this. Sample statements are supplied to create the objects necessary to support declared temporary tables. You must run a version of this job for each DB2 subsystem you want File Manager/DB2 to use. Refer to the comments in the job for information about changes you may need to make to the job.

# PK29853

**Initial problem description**

When comparing data sets using read-ahead synchronization with a synchronization length greater than one records may be reported out of sequence or may not be reported at all.

When generating a BATCH compare job the SHOWATTR processing option is not being created correctly (PFM02068).

When generating a BATCH compare job there is not parameter to specify that numeric fields are to be left-justified if a Formatted report has been requested with the WIDE option set to NO (PFM01997).

FIND/RFIND command does not work correctly on last position of the screen or off screen (PFM01585, PFM01814, PFM01913).

Incorrect fields marked as changed with HOLD in DSM Compare templates (PFM01719).

New IAM dataset allocation with a model dataset may get allocated with incorrect attributes (PFM01730, PFM01731).

Incorrect record deleted during auxiliary edit of a large QSAM dataset after a series of EXCLUDE commands (PFM02034).

When executing the DSP batch function twice in the same job and the first function execution include the PROC parameter the second execution of the function expects the PROC parameter to be supplied (PK28839).

When producing an IMS Audit report File manager may ABENDS0C4 (PK29446).

When changing JCL and the replace string is longer than the search string the changed JCL statement may be corrupt (PK29597).

File Manager may ABENDSOC4 when generating the DB2 Audit report (PK29754, PFM01526).

File Manager may ABENDS0C4 when generating the Audit Report if the record and the template do not match and the record is shorter than the record definition in the template (PK30204).

ABEND0C4 S0C4 may occur in FMNFCH when the user specifies a user proc with no REXX/FASTREXX statements in FCH batch (PK30703).

Datasets/DDNAME not found condition may be reported when multiple functions reference the same VSAM dataset DDNAME (PFM02084).

User IO exit never called for term and other IO exit processing problems (PFM01917).

When comparing data sets using a template containing arrays, array elements may be incorrectly reported as changed when they are not (PMR 06888,101,616).

**Outline of solution**

When comparing data sets using read-ahead synchronization with synchronization length greater than one all records will be reported correctly.

Skeleton and code have been corrected to allow for proper SHOWATTR generation.

The JUST option has been added to the skeleton and the roadmap.

When executing the DSP or DSC batch functions twice in the same job and the first function execution includes the PROC parameter the second execution of the function will execute successfully if the PROC parameter is not supplied.

When producing an IMS Audit report File Manager will confirm that the current template is for the same DBD as the log record being report. If this is not the case a message will be issued and an unformatted report generated.

The changed JCL statement will not be corrupted when the replace string is longer than the search string.

If an invalid template is found when generating the DB2 Audit report a message will be generated and an unformatted report will be produced. This APAR also resolves internally reported problem PFM01526.

File Manager will not examine fields whose boundaries exceed the record boundary.

**Documentation impact**

This APAR requires changes to be made to the User's Guide and Reference (SC18-9655-01).

## Changes to the User's Guide and Reference

### Chapter 17, Functions

Add the following parameter to the **Comparison options** in the DSM Syntax diagram Part 3 of 3:

```
          ┌─JUST=NO──┐
►►────────┤          ├──────────────────────────────────────►◄
          └─JUST=YES─┘
```
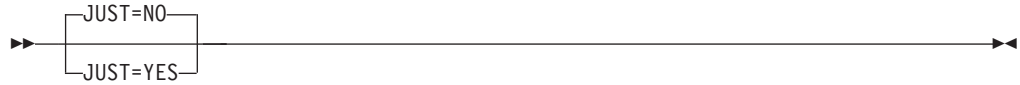
In the list of descriptions that follow the syntax diagrams, add the following:

**Yes**    Left-justify numeric field if formatted report selected and WIDE listing option not selected.

**No**    Do not left-justify numeric field if formatted report selected and WIDE listing option not selected.

# UK17161, UK17165, UK17197, UK17204, UK17223, UK17224

Release Date: **22 August 2006**

This set of PTFs contains the following APAR fixes:

| APAR # | APAR Abstract | Doc Impact |
|---|---|---|
| PK24798 | A blank was missing before character ″9″ in column 68. A blank was assumed. | User's Guide and Reference (SC18-9655-01) User's Guide and Reference for DB2 Data (SC18-9656-01) User's Guide and Reference for IMS Data (SC18-9657-01) |
| PK27656 | Using FM FCH function with a CHANGE command and the JCL option can produce invalid results. | User's Guide and Reference (SC18-9655-01). |
| PK27731 | Loop copying uncataloged data set. | None. |
| PK28454 | SYSUT3 allocation error in FM due to tso session pre-allocating it. | None. |
| PK29594 | File Manager edit data corruption may occur when editing a VSAM KSDS data set. | None. |
| PK29663 | FM SVC99 deallocate failed - on exit from Browse that uses a copybook from a subsystem library. | None. |

## PK24798

**Initial problem description**

When generating a template using a copybook which contains a value clause where the decimal point in the numeric value is a comma a Compilation Error will occur. This is because the copybook needs the COBOL SPECIAL-NAMES paragraph ″Decimal-point is comma″.

**Outline of solution**

Support for a COBOL SPECIAL-NAMES paragraph ″Decimal_point as comma″ and the COBOL compile option Arith(extend) have been added.

**Documentation impact**

This APAR requires changes to be made to:
- User's Guide and Reference (SC18-9655-01)
- User's Guide and Reference for DB2 Data (SC18-9656-01)
- User's Guide and Reference for IMS Data (SC18-9657-01)

# Changes to the User's Guide and Reference

### Chapter 4, ″**Creating and editing templates**″

In the section, ″Setting your COBOL processing options″:

- Add two new steps after step 5:

  6. To use the COBOL SPECIAL-NAMES paragraph, ″Decimal-point is comma″, when compiling COBOL copybooks, select the option, **Decimal-point is comma**.

  When you select this option, the COBOL compiler exchanges the functions of the period and the comma in PICTURE character strings and in numeric literals.

  7. To use the Arith(extend) COBOL compile option when compiling COBOL copybooks, select the option, **Arith(extend)**.

- Renumber Step 6 to Step 8.

### Chapter 15, ″**Panels and fields**″

In the section, ″Set COBOL Processing Options panel (option 0.6)″:

- Replace the panel shown with the following:

```
 Process    Options    Help
 _____

 File Manager              Set COBOL Processing Options

 Additional SYSLIB data sets:
 1.  _____
 2.  _____
 3.  _____
 4.  _____
 5.  _____
 6.  _____
 7.  _____
 8.  _____
 9.  _____
 10  _____

 COBOL Replacing Options:
    From string                    To string
 1. _____ by_____
 2. _____ by_____
 3. _____ by_____
 4. _____ by_____
 5. _____ by_____

 COBOL Compiler Options:
    Enter "/" to select option
    _ DBCS
    _ Decimal-point is comma
    _ Arith(extend)

    Maximum Return Code to be accepted from compiler  04  (0-99)
 Command ===> _____
  F1=Help      F2=Split     F3=Exit      F4=CRetriev  F7=Backward  F8=Forward
  F9=Swap      F10=Actions  F12=Cancel
```

*Figure 2. Set COBOL Processing Options*

- In the list of descriptions that follows the panel, after the description for **DBCS** (under ″COBOL Compiler Options″), add:

  **Decimal-point is comma**
  > Uses the ″Decimal-point is comma″ SPECIAL-NAMES paragraph when compiling COBOL copybooks.

**Arith(extend)**
Uses the Arith(extend) COBOL compile option when compiling COBOL copybooks.

## Chapter 17, ″**Functions**″

In the functions DSB, DSC, DSE, DSEB, DSG, DSM, DSP, DSU and PBK:

- Add two new COBOL options, CDPC and CAE, to the syntax diagram (under ″COBOL options″):

```
          ┌─CDPC=NO──┐   ┌─CAE=NO──┐
►►────────┤          ├───┤         ├───────────────────────────────────────►◄
          └─CDPC=YES─┘   └─CAE=YES─┘
```

- In the list of descriptions that follows the syntax diagram, add the following under ″COBOL options″:

**CDPC=NO**
Do not use the COBOL SPECIAL-NAMES paragraph ″Decimal-point is comma″.

**CDPC = YES**
Use the COBOL SPECIAL-NAMES paragraph ″Decimal-point is comma″.

**CAE=NO**
Do not use the COBOL compile option ARITH(EXTEND).

**CAE = YES**
Use the COBOL compile option ARITH(EXTEND).

# Changes to the User's Guide and Reference for DB2 Data

## Chapter 16, ″**FM/DB2 panels and fields**″

In the section, ″Set COBOL Processing Options panel″:

- Replace the panel shown with the following:

```
  _____
 /                                                                \
|   Process   Options   Utilities   Help                          |
|  _____ |
|   FM/DB2 (DFA2)           Set COBOL Processing Options           |
|                                                                  |
|   Additional SYSLIB data sets:                                   |
|   1.  _____                       |
|   2.  _____                       |
|   3.  _____                       |
|   4.  _____                       |
|   5.  _____                       |
|   6.  _____                       |
|   7.  _____                       |
|   8.  _____                       |
|   9.  _____                       |
|   10. _____                       |
|                                                                  |
|   COBOL REPLACE Options:                                         |
|      From string                      To string                 |
|   1. _____ by  _____ |
|   2. _____ by  _____ |
|   3. _____ by  _____ |
|   4. _____ by  _____ |
|   5. _____ by  _____ |
|                                                                  |
|   COBOL Compiler Options:                                        |
|      Enter "/" to select option                                 |
|      _ DBCS                                                      |
|      _ Decimal-point is comma                                   |
|      _ Arith(extend)                                            |
|                                                                  |
|      Maximum Return Code to be accepted from compiler  04 (0-99) |
|   Command ===> _____ |
|    F1=Help      F2=Split     F3=Exit      F7=Backward F8=Forward  F9=Swap |
|   F12=Cancel                                                     |
 \                                                                /
  ----------------------------------------------------------------
```

- In the list of descriptions that follows the panel, replace the description for **COBOL Compiler Options** with:

**COBOL Compiler Options**
> Compiler options to be used when compiling a COBOL copybook into a template:
>
> **DBCS**  Instructs File Manager to use the DBCS compiler option.
>
> **Decimal-point is comma**
> > Uses the "Decimal-point is comma" SPECIAL-NAMES paragraph when compiling COBOL copybooks.
>
> **Arith(extend)**
> > Uses the Arith(extend) COBOL compile option when compiling COBOL copybooks.

## Changes to the User's Guide and Reference for IMS Data

### Chapter 9, "Panels and fields"
In the section, "Set COBOL Processing Options panel":

- Replace the panel shown with the following:

```
  _____
 /                                                                    \
|  Process    Options    Help                                          |
|  _____ |
|                                                                      |
|  FM/IMS                    Set COBOL Processing Options              |
|                                                                      |
|   COBOL Replacing Options:                                           |
|      From string                      To string                     |
|   1. _____ by _____  |
|   2. _____ by _____  |
|   3. _____ by _____  |
|   4. _____ by _____  |
|   5. _____ by _____  |
|                                                                      |
|                                                                      |
|   COBOL Compiler Options:                                            |
|      Enter "/" to select option                                      |
|      _ DBCS                                                          |
|      _ Decimal-point is comma                                        |
|      _ Arith(extend)                                                 |
|                                                                      |
|      Maximum Return Code to be accepted from compiler  04  (0-99)    |
|                                                                      |
|                                                                      |
|  Command ===> _____  |
|   F1=Help      F2=Split     F3=Exit      F4=CRetriev  F7=Backward  F8=Forward  |
|   F9=Swap      F10=Actions  F12=Cancel                               |
 \                                                                    /
  --------------------------------------------------------------------
```

*Figure 3. Set COBOL Processing Options panel*

- In the list of descriptions that follows the panel, after the description for **DBCS** (under ″COBOL Compiler Options″), add:

  **Decimal-point is comma**
  > Uses the ″Decimal-point is comma″ SPECIAL-NAMES paragraph when compiling COBOL copybooks.

  **Arith(extend)**
  > Uses the Arith(extend) COBOL compile option when compiling COBOL copybooks.

# PK27656

**Initial problem description**
> When one uses FCH and specifies a CHANGE command with the JCL option, and *string2* is shorter (different) than *string1*, then data (sequence numbers) get shifted in the non-JCL records.

**Outline of solution**
> When using the **JCL source format** option, the columns searched for FIND or CHANGE command processing are set to 3 through 71, unless the statement is not a JCL statement. A statement is considered to be a JCL statement if it begins with any of the strings ′/*′ or ′//′. If the statement does not begin with any of these strings, it is not considered to be a JCL statement in which case any column range specified on the CHANGE or FIND command (or preset using BOUNDS command) is honored. This method can be used to stop data being moved in non-JCL records.

**Documentation impact**
> This APAR requires changes to be made to the User's Guide and Reference (SC18-9655-01).

# Changes to the User's Guide and Reference

### Chapter 15, Panels and fields

Add the following comment to the description of the **JCL source format** option on the Find/Change Utility panel:

"When using the **JCL source format** option, the columns searched are set to 3 through 71, unless the statement is not a JCL statement. A statement is considered to be a JCL statement if it begins with the strings "/*" or "//". If the statement does not begin with either of these strings, it is not considered to be a JCL statement in which case any column range specified on the FIND (or CHANGE, respectively) command or preset using the BOUNDS command is honored. If no column range has been specified, the full record is searched."

### Chapter 16, Primary commands

Add the following comment to the documentation for the BOUNDS Primary command:

"If this command is used in the Find/Change utility and the **JCL source format** option has been selected, the columns searched are set to 3 through 71, unless the statement is not a JCL statement. A statement is considered to be a JCL statement if it begins with the strings "/*" or "//". If the statement does not begin with either of these strings, it is not considered to be a JCL statement in which case any column range specified is honored."

Replace the existing **Note** in the description of the **col1** parameter of the CHANGE primary command with:

"When you use the CHANGE primary command in the Find/Change Utility, if the **JCL Source format** option is selected, the columns searched are set to 3 through 71, unless the statement is not a JCL statement. A statement is considered to be a JCL statement if it begins with the strings "/*" or "//". If the statement does not begin with either of these strings, it is not considered to be a JCL statement in which case any column range specified on the FIND (or CHANGE, respectively) command or preset using the BOUNDS command is honored. If no column range has been specified, the full record is searched."

Replace the existing **Note** in the description of the **col1** parameter of the FIND primary command with:

"When you use the FIND primary command in the Find/Change Utility, if the **JCL Source format** option is selected, the columns searched are set to 3 through 71, unless the statement is not a JCL statement. A statement is considered to be a JCL statement if it begins with the strings "/*" or "//". If the statement does not begin with either of these strings, it is not considered to be a JCL statement in which case any column range specified on the FIND (or CHANGE, respectively) command or preset using the BOUNDS command is honored. If no column range has been specified, the full record is searched."

### Chapter 17, Functions

Add the following comment to the description of the **JCL=YES** parameter of the FCH (Find/Change) function:

"The columns searched are set to 3 through 71, unless the statement is not a JCL statement. A statement is considered to be a JCL statement if it begins with the strings "/*" or "//". If the statement does not begin with either of these strings, it

is not considered to be a JCL statement in which case any column range specified on the FIND (or CHANGE, respectively) command or preset using the BOUNDS command is honored. If no column range has been specified, the full record is searched."

**PK27656**

# UK15298, UK15307, UK15308, UK15317

Release Date: **20 June 2006**

This set of PTFs contains the following APAR fixes:

| APAR # | APAR Abstract | Doc Impact |
|---|---|---|
| PK25463 | In FM/IMS, the REPLIMS function is updating the wrong segment. | User's Guide and Reference for IMS Data (SC18-9657-01). |

## PK25463

**Initial problem description**

All components:

1. If the cursor is in the data when a FIND, CHANGE, RFIND or RCHANGE command is issued and the search reaches the end of the search range without finding an occurrence of the search string, the cursor moves back to the command line. It should remain where it was before the command was issued.

FM/IMS component:

2. When editing or browsing a Fast Path database with a view that has selection criteria on the root segment, FM/IMS does not display all the segments that satisfy the selection criteria. The problem occurs when FM/IMS converts the selection criteria into a qualified SSA that specifies a range of root key values.

3. FM/IMS Edit and Browse functions go into a loop repeatedly issuing the message: 'FMN1POSN+00000708 fail get first sc=GE' after pressing the Enter key on the View Specification panel. This error occurs when the database becomes empty after the function has accessed the database.

4. The Edit and Browse functions do not always correctly handle an IMS abend.

5. When a view with a selection criteria on the root segment is used to edit or browse an HDAM database, FM/IMS does not always display all the segments that satisfy the selection criteria.

6. Segment Update/Delete Warning popup displayed when another user hasn't updated or deleted the segment you were attempting to update or delete. The problem occurs when:
- Editing with a view and there is a selection criteria either on the segment you were attempting to update or on a parent of the segment you were attempting to update, and
- The selection criteria is on a segment with a nonunique key or no key and FM/IMS converts the selection criteria to a qualified SSA that does not specify key fields.

7. When multiple line commands are entered on the Edit or Browse data panel, FM/IMS does not validate them all before processing them.

**31**

8. If the FIND...ALL command is issued when the scope is DBREC and the top of window line is displayed, the editor scrolls down one line.

9. A CHANGE...ALL command issued when the scope is DBREC does not change an occurrence of the search string in the current database record and after the command is executed the editor displays a different database record.

10. A FIND...ALL command issued when the first occurrence of the search string is on the current page scrolls the data. The editor should only scroll when the first occurrence of the search string is not on the current page.

11. A FIND...ALL command issued when the first occurrence of the search string is before the start of the current page does not always display the target string at the top of the page.

12. An RFIND command issued after a FIND...REC command searches for strings outside of the search range when you place the cursor outside of the search range.

13. An RFIND command issued after a FIND...ALL command doesn't wrap to the top of the search range after the bottom of the search range has been reached.

14. A FIND...NOREC...FIRST command issued when the scope is DBREC does not find the first occurrence of the search string in the database.

15. The FIND...REC command and the FIND...CHILD command highlight occurrences of strings that are outside the search range.

16. The FIND...SEGMENT(segment-name) command highlights strings in all segments.

17. An RFIND command issued after a FIND...CHILD command issues message 'Segment has no dependents' when you place the cursor outside of the search range.

18. An RFIND command issued after a FIND...CHILD command wraps to the start of the database (rather than to the start of the search range) when the bottom of the search range has been reached.

19. If you:
- Issue a FIND...CHILD command, then
- Scroll the occurrence of the search string that is found to the top of the page, then
- Issue RFIND commands

the RFIND command does not wrap when it reaches the bottom of the range. It issues the message 'No chars found' instead.

20. Message "DFS0929I BLDL FAILED" and IMS U929 abend PFM1634 in the Edit or Browse dialog when you return to the Database Data Set Specification panel and then go forward in the dialog again.

21. Key Specification panel displays garbage data in key value field when a segment has no key. When a segment has no key, the panel should not display a line for the segment.

22. Garbage data displayed and some Cmd and Key Value fields missing after issuing the HEX ON or HEX OFF command on the Database Positioning panel.

23. On the IMS data panel, fields of segments that IMS does not allow you to update should:

- Be highlighted,
- Be protected when in SNGL or TABL display format, and
- Not be translated to uppercase when the user has CAPS ON.

But FM/IMS does not highlight or protect the key field of the destination parent in a concatenated segment and when the user has CAPS ON, it translates the characters in this field to uppercase, which causes the REPL call to fail with a status code of DA.

FM/IMS IEB (IMS Edit Batch):

24. A DELIMS, REPLIMS or ISRTIMS call without a WHERE clause issues message 'IMS error occurred for the function GET HOLD Status code=GE' and receives a return code of 8 when there is a PRINT call between the preceding GETIMS call and the failing update call. The error occurs when the concatenated key of the segment that the PRINT call prints has unprintable characters.

25. A GETIMS call with the NEXT parameter and a WHERE clause should retrieve the next segment in the database that satisfies the WHERE clause. But it retrieves the first segment in the database that satisfies the WHERE clause.

26. A REPLIMS/DELIMS call with a WHERE clause replaces/deletes the first segment in the database that satisfies the WHERE clause. But the call doesn't allow you to replace/delete the next segment in the database that satisfies the WHERE clause.

27. A REPLIMS/DELIMS call without a WHERE clause should replace/delete the segment that the preceding GETIMS call retrieved. But when this segment or one of its parents has a nonunique key or no key, the REPLIMS/DELIMS call may replace/delete the wrong segment.

28. If you position on the parent of the segment that you want to insert and then issue an ISRTIMS call without a WHERE clause, the segment should be inserted under this parent. But when the parent or one of its parents has a nonunique key or no key, the ISRTIMS call may insert the segment under the wrong parent.

29. A GETIMS call with the ROOT parameter should retrieve the root segment that the current segment is under. But when the root segment has no key, the GETIMS call retrieves the first root segment in the database. When the root segment has a nonunique key, the GETIMS call retrieves the first root segment with the same root key as the current segment.

30. A GETIMS call with the TOP and the USING_VIEW parameters may retrieve the first root segment in the database even when the root segment is not included in the view. This call should retrieve the first segment in the database that is in the view.

31. An ISRTIMS call without a SET clause should obtain the data for the new segment from the OUTREC variable. But when the ISRTIMS call has a WHERE clause, it may obtain the data from the first segment in the database that satisfies the WHERE clause.

32. A REPLIMS call without a SET clause should replace the segment in the database with the data in the OUTREC variable. But when the REPLIMS call has a WHERE clause, it may not modify the I/O area before it executes the DL/I REPL call.

**Outline of solution**

File Manager has been modified to correct all the listed defects.

2. FM/IMS has been modified to generate a different qualified SSA that will retrieve all the root segments that satisfy the selection criteria. Please note, however, that for this qualified SSA to retrieve the required segments, you also need to apply APAR PK22138 (UK14330) to IMS V9 or APAR PK24421(UKxxxxx) to IMS V8. Without these IMS fixes, the qualified SSA still won't retrieve all the root segments that satisfy the selection criteria.

3. FM/IMS Edit and Browse functions have been modified to check whether the database is empty immediately prior to the display of the Database Positioning panel. The empty database check has been integrated with the unavailable partition/area check and some other Database Positioning calls to make it more reliable and to reduce the number of IMS calls. When in Edit, FM/IMS no longer checks whether the database is empty when the database is first accessed. It now checks immediately prior to the display of the Database Positioning panel. If it is empty, FM/IMS bypasses the display of the Database Positioning panel.

When in Browse, FM/IMS now only performs the initial empty database check when the Use view option on the Browse Entry panel is selected. When this option is not selected, this check is not required because, as with Edit, FM/IMS now checks whether the database is empty immediately prior to the display of the Database Positioning panel. If either of these checks find that the database is empty, FM/IMS returns the user to the Browse Entry panel.

The FM/IMS IEB function has been modified as follows: 24. The PRINT call has been modified to use a copy of the concatenated key.

25. You may now specify either the NEXT parameter or the FIRST parameter on a GETIMS call with a WHERE clause. With the NEXT parameter, the call retrieves the next segment in the database that satisfies the WHERE clause. With the FIRST parameter, the call retrieves the first segment in the database that satisfies the WHERE clause. The default has been changed to NEXT.

26. You may now specify either the NEXT parameter or the FIRST parameter on a REPLIMS/DELIMS call with a WHERE clause. With the NEXT parameter, the call replaces/deletes the next segment in the database that satisfies the WHERE clause. With the FIRST parameter, the call replaces/deletes the first segment in the database that satisfies the WHERE clause. The default has been changed to NEXT.

Please note that with modifications 25 and 26, existing EXECs may behave differently. But by adding the FIRST parameter to all GETIMS, REPLIMS and DELIMS calls with a WHERE clause, you can preserve the behavior that they had prior to this APAR fix.

27. When the preceding GETIMS call has the USING_VIEW parameter, a REPLIMS/DELIMS call with no WHERE clause will replace/delete the segment that the GETIMS call retrieved (as it should). But when the preceding GETIMS call does not have the USING_VIEW parameter and the segment you are attempting to replace/delete or one of its parents has a nonunique key or no key, you will be required to add the new HOLD parameter to the GETIMS call (see 35 below). IEB will let you know if this parameter is required.

28. If you position on the parent of the segment that you want to insert and then issue an ISRTIMS call without a WHERE clause, the segment will be inserted under this parent.

29. A GETIMS call with the ROOT parameter and the USING_VIEW parameter will retrieve the root segment that the current segment is under (as it should). But when a GETIMS call with the ROOT parameter does not have the USING_VIEW parameter and the root segment has a nonunique key or no key, the GETIMS call will issue a diagnostic and the EXEC will be terminated.

30. A GETIMS call with the TOP and the USING_VIEW parameters will retrieve the first segment in the database that is in the view.

31. An ISRTIMS call without a SET clause will obtain the data for the new segment from the OUTREC variable.

32. A REPLIMS call without a SET clause will replace the segment in the database with the data from the OUTREC variable.

33. The dynamic PSB that is generated for the IEB function will have one database PCB. Before this APAR fix, the dynamic PSB had two database PCBs -- a read PCB that the GETIMS call used and an update PCB that the REPLIMS, DELIMS and ISRTIMS calls used.

34. The GETIMS, REPLIMS, DELIMS and ISRTIMS calls all use the same PCB.

35. You may now specify the HOLD parameter on a GETIMS call that does not have the USING_VIEW parameter. A GETIMS with a HOLD parameter will retrieve the segment with a hold. Although in most cases it is not required, if you add a HOLD parameter to each GETIMS call that precedes a REPLIMS or DELIMS call your EXEC will execute less DL/I calls and run more efficiently.

36. When a GETIMS call with a HOLD parameter precedes a REPLIMS/DELIMS call without a WHERE clause, the REPLIMS/DELIMS call skips the GET HOLD DL/I call that, previously, it always executed.

37. An ISRTIMS call with no WHERE clause will only reposition on the current segment when the previous IMS call was a GETIMS call with a USING_VIEW parameter. In all other cases, repositioning before the insert is no longer required because all the calls now use the same PCB.

Please note that with modifications 27 through 37, existing EXECs may behave differently.

**Documentation impact**

This APAR requires changes to be made to the User's Guide and Reference for IMS Data (SC18-9657-01).

## Changes to the User's Guide and Reference for IMS Data

### Chapter 11. Functions

In the section, "General tips for using the REXX functions defined by IEB":

- Delete the section titled "Locating a segment with non-unique keys or no keys".
- At the end of the section titled "Locating a segment by specifying its key values", add the following text:
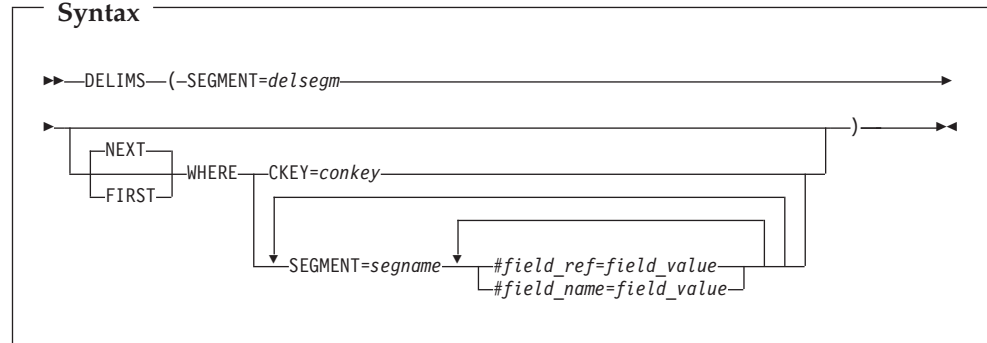
"When the WHERE clause is on a GETIMS, DELIMS or REPLIMS call, you may specify the FIRST parameter or NEXT parameter before it. When the FIRST

parameter is specified, the call locates the first segment in the database that satisfies the WHERE clause. When the NEXT parameter is specified, the call locates the next segment in the database after the current segment that satisfies the WHERE clause. The default is NEXT.

When the WHERE clause is on an ISRTIMS call, the call locates the first segment in the database that satisfies the WHERE clause."

In the section, "DELIMS (Delete an IMS segment)":

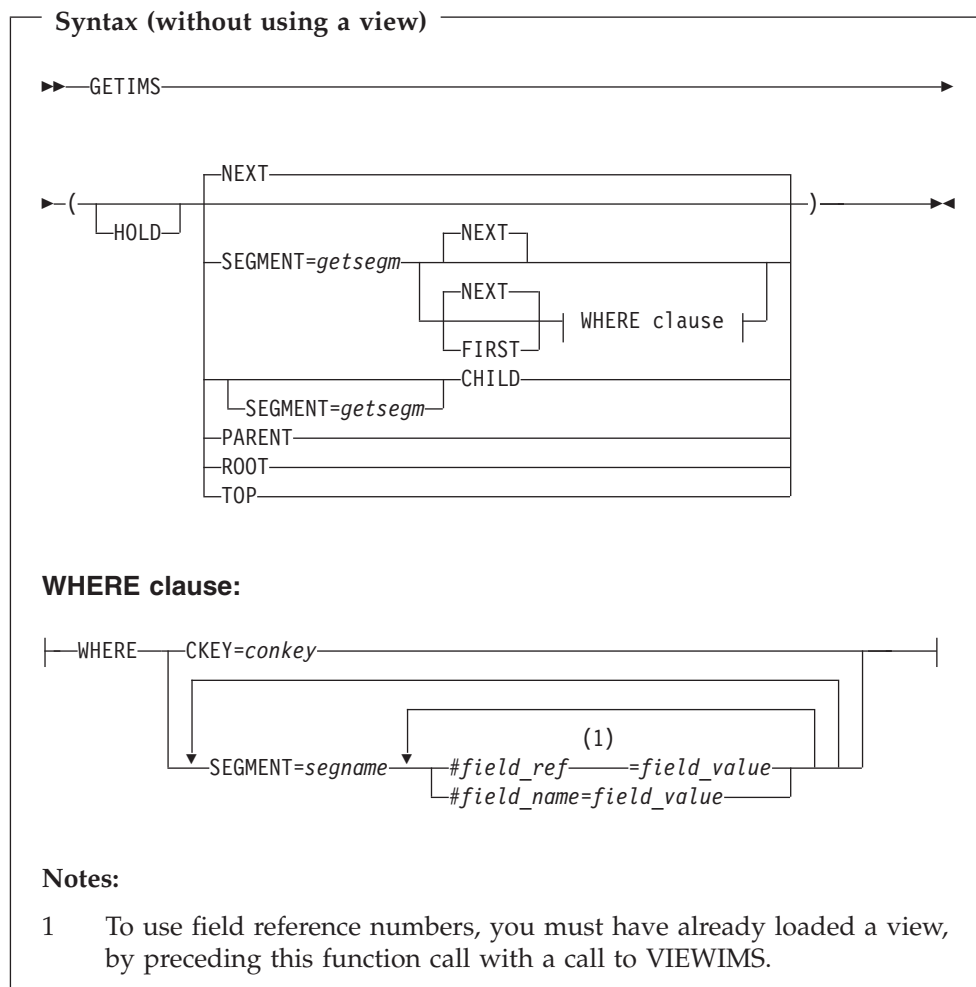• Replace the syntax diagram with the following:



**Syntax**

```
►►──DELIMS──(─SEGMENT=delsegm─────────────────────────────────────────►

►──┬───────┬────────────────────────────────────────────────────┬──)──►◄
   ├─NEXT──┤                                                      │
   │       ─WHERE──┬─CKEY=conkey──────────────────────────────┐  │
   └─FIRST─┘       │                                           │  │
                   └──►──SEGMENT=segname──┬─#field_ref=field_value──┤
                                          └─#field_name=field_value─┘
```

• After the description for the parameter, **delsegm**, add:

**NEXT**  Deletes the next segment in the database after the current segment that satisfies the WHERE clause.

**FIRST**  Deletes the first segment in the database that satisfies the WHERE clause.

In the section, "GETIMS (Retrieve an IMS segment)":

• Replace the syntax (without using a view) diagram with the following:

**Syntax (without using a view)**

```
►►──GETIMS────────────────────────────────────────────────────────►

          ┌─NEXT──────────────────────────────────┐
►─(──┬──────┬──┼───────────────────────────────────────┼──)──►◄
     └─HOLD─┘  │            ┌─NEXT─┐                     │
              ├─SEGMENT=getsegm──┼──────┼──────────────────┤
              │            │ ┌─NEXT─┐               │
              │            │ ├──────┤─ WHERE clause ─┤
              │            └─┴FIRST─┘               │
              │            ┌─CHILD──────────────────┤
              ├─┬────────────────┬─┴                │
              │ └─SEGMENT=getsegm─┘                   │
              ├─PARENT────────────────────────────────┤
              ├─ROOT──────────────────────────────────┤
              └─TOP───────────────────────────────────┘
```

**WHERE clause:**

```
                                                            (1)
├──WHERE──┬─CKEY=conkey───────────────────────────────────────┬──┤
          │                                     ┌──────────┐  │
          └─SEGMENT=segname─┬─#field_ref────=field_value─┴─┘
                            └─#field_name=field_value──────┘
```

**Notes:**

1  To use field reference numbers, you must have already loaded a view, by preceding this function call with a call to VIEWIMS.

- Add the following to the list of parameters that follows the above syntax diagram:

**HOLD**

Retrieves the segment with a hold. Code this parameter when the GETIMS call is followed by a REPLIMS call without a WHERE clause, or a DELIMS call without a WHERE clause.
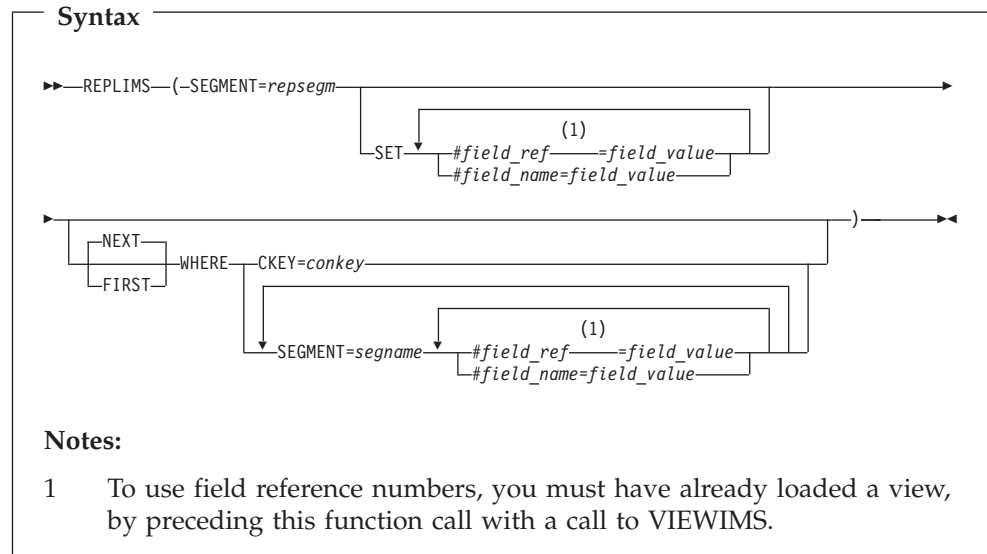
If the segment or one of its parents has a nonunique key or no key, this parameter is mandatory. Without it, the REPLIMS call or DELIMS call fails. The REPLIMS call or DELIMS call will not fail if the segment and its parents have unique keys but, if you do code it, your EXEC executes less DL/I calls and runs more efficiently.

- After the description for the parameter NEXT, add:

**FIRST**  Retrieves the first segment in the database that satisfies the WHERE clause.

In the section, "REPLIMS (Update an IMS segment)":

- Replace the syntax diagram with the following:

**PK25463**

```
┌─ Syntax ──────────────────────────────────────────────────────────────────┐
│                                                                            │
│  ►►──REPLIMS──(─SEGMENT=repsegm──────────────────────────────────────►     │
│                                              (1)                           │
│                            ┌──────#field_ref──────=field_value──┐          │
│                    └─SET─▼─┴──#field_name=field_value───────────┘          │
│                                                                            │
│  ►─────────────────────────────────────────────────────────────)──►◄      │
│      ┌─NEXT─┐                                                              │
│      ┼──────┼──WHERE──┬─CKEY=conkey────────────────────────┐              │
│      └─FIRST┘         │                                     │              │
│                       │                          (1)        │              │
│                       │              ┌──#field_ref──=field_value─┐         │
│                       └─SEGMENT=segname─┴─#field_name=field_value─┘         │
│                                                                            │
│  Notes:                                                                    │
│                                                                            │
│  1    To use field reference numbers, you must have already loaded a view, │
│       by preceding this function call with a call to VIEWIMS.              │
│                                                                            │
└────────────────────────────────────────────────────────────────────────────┘
```

- After the description for the parameter repsegm, add:

  **NEXT**    Replaces the next segment in the database after the current segment that satisfies the WHERE clause.

  **FIRST**    Replaces the first segment in the database that satisfies the WHERE clause.

# Part 2. General Documentation Changes

This section describes enhancements and updates in the documentation for File Manager for z/OS Version 6 Release 1. These changes are not associated with individual APAR or PTF numbers, as they do not require the application of any code updates.

The changes are grouped by manual and listed within each section in reverse date order. That is, the most recent documentation change appears at the beginning of each manual section.

# Customization Guide (SC18-9650-01)

## October 2007

**Chapter 1,** ″**Preparing to Customize File Manager**″

Section, ″Modifying the TSO logon procedure″

Add the following at the end of this section:

If you plan to use options 8.1 or 8.2 (HFS browse or edit), ensure that the following BCP target libraries are concatenated, as follows, when you invoke File Manager.

```
SBPXEXEC concatenated to SYSEXEC DDname
SBPXMENU concatenated to ISPMLIB DDname
SBPXPENU concatenated to ISPPLIB DDname
SBPXTENU concatenated to ISPTLIB DDname
```

If you do not have these libraries available, you may get 'panel not found' errors when you invoke 8.1 or 8.2.

## February 2007

- **Chapter 4,** ″**Customizing the File Manager security environment**″

  In the section: ″Controlling access to File Manager functions with SAF″ replace the opening sentence and two bullet points with the following new paragraphs and box.

  SAF controls access to File Manager functions as follows:

  – If access to the profile FACILITY(FILEM.FUNCTION.*fc*) in the FACILITY class is defined (where *fc* is the function code), this controls access to the function.

  – If access to the profile FACILITY(FILEM.FUNCTION.*fc*) in the FACILITY class is not defined, the profile name shown in the table ″File Manager function to profile name cross-reference″ (in the form FILEM.*group.name*) is used.

  – If no profile name as shown in the table ″File Manager function to profile name cross-reference″ is defined, then FILEM.OTHER.ALL is used. If this does not permit access then access is denied.

  Some File Manager functions are protected, by default, by the FILEM.OTHER.ALL profile. These functions are listed in the table ″File Manager functions protected by FILEM.OTHER.ALL″.

> **Important information for users of non-IBM security products**
>
> File Manager issues a RACROUTE TYPE=AUTH for the required profile, and if the required profile is not defined, RACF returns RC=4. If RC=4 is returned, File Manager issues a RACROUTE for FILEM.OTHER.ALL. If a higher return code is returned, no RACROUTE is issued for FILEM.OTHER.ALL and the access request fails immediately.
>
> You should be aware that not all non-IBM security products issue RC=4 when a RACROUTE TYPE=AUTH for a profile fails; in this case no RACROUTE is issued for FILEM.OTHER.ALL and the access request fails immediately. If this applies to your security product, you will have to provide individual profiles for all File Manager functions listed in the table "File Manager functions protected by FILEM.OTHER.ALL".

- **Chapter 4,** "**Customizing the File Manager security environment**"

  Replace the section: "Unprotected functions and profile names for protected functions", including all text and the two tables, with the following new text, and three tables.

  Table 1, Table 2 on page 43, and Table 3 on page 44, show File Manager functions with their descriptions and (where applicable) the equivalent File Manager panel you can use under ISPF.

  Table 1 lists the functions that cannot be protected by RACF or an equivalent security product, or by the FMNSECUR exit. Functions not listed in Table 1 are protected, and Table 3 shows the profile name used in the FACILITY class to protect a specific function.

  All protected functions not listed in Table 3 are protected by the FILEM.OTHER.ALL profile.

  In particular, the File Manager functions listed in Table 2 are protected by default when you receive File Manager. They are protected by the profile "FILEM.OTHER.ALL", and if you want to use these functions, you must first give the users access to these functions, by defining the profile "FILEM.OTHER.ALL" to RACF or your equivalent security product.

  For information on File Manager profiles, and defining them to your security product, refer to the section "Setting up the security environment by using RACF or an equivalent security product" in chapter 4 of the File Manager Customization Guide. In particular, if you use a non-IBM security product, see the warning in "Important information for users of non-IBM security products".

  **Note:** In Table 3, where more than one function code is listed for the same function, and the first function code is in **bold** letters, you should use this function code when defining the relevant function profile.

*Table 1. File Manager unprotected functions*

| Function | Equivalent panel | Description |
|----------|------------------|-------------|
| AF | | Template Workbench |
| APB | 3.7 | AFP™ Print Browse |
| AUD | | Print Audit Trail |
| DSB | 1 | Data Set Browse |
| DSC, COPY | 3.3 | Copy Utility |

*Table 1. File Manager unprotected functions  (continued)*

| Function | Equivalent panel | Description |
|---|---|---|
| DSE | 2 | Data Set Edit |
| DSEB | | Data Set Edit (batch) |
| DSG | 3.1 | Data Create Utility |
| DSM, DSCMP | 3.11 | Data Set Compare |
| DSP | 3.2 | Print Utility |
| DSU | | Data Set Update (batch) |
| DVT | 3.5 | Display VTOC |
| DX | | Decimal to hex conversion |
| FCH | 3.6 | Find/Change Utility |
| FMT | 3.0 | Set DBCS Format |
| LVL | | Show File Manager Release and PTF Level |
| MB, TST | 3.8 | Memory Browse |
| NEW, NEWS | | Show File Manager Release News |
| NODF, NOSORT | | Do not use DFSORT |
| PB | 3.9 | Print Browse |
| PBK | | Print/view copybook |
| SCS | 3.4 | Catalog Services |
| SET | | Set File Manager Processing Options |
| VER | | Show File Manager Release and PTF Level |
| VLM | | View Load Module |
| XD | | Hex to decimal conversion |
| (none) | | Template Functions |

*Table 2. File Manager functions protected by FILEM.OTHER.ALL*

| Function | Equivalent panel | Description | Profile |
|---|---|---|---|
| BSF | 4.15.1 | Backward space file (tape) | FILEM.OTHER.ALL |
| BSR | 4.15.3 | Backward space record (tape) | FILEM.OTHER.ALL |
| DVB | 5.8 | Data in virtual browse | FILEM.OTHER.ALL |
| FSF | 4.15.2 | Forward space file (tape) | FILEM.OTHER.ALL |
| FSR | 4.15.4 | Forward space record (tape) | FILEM.OTHER.ALL |
| OB | 6.2 | Object browse | FILEM.OTHER.ALL |
| ODL | 6.1 | Object directory list | FILEM.OTHER.ALL |
| OP | 6.3 | Object print | FILEM.OTHER.ALL |
| OQ, OS | 6.6.2 | Object to sequential data | FILEM.OTHER.ALL |
| OV | 6.6.1 | Object to VSAM | FILEM.OTHER.ALL |
| REW | 4.15.5 | Rewind (tape) | FILEM.OTHER.ALL |
| RUN | 4.15.6 | Rewind and unload (tape) | FILEM.OTHER.ALL |
| VX | | VSAM to REXX stem | FILEM.OTHER.ALL |

*Table 2. File Manager functions protected by FILEM.OTHER.ALL  (continued)*

| Function | Equivalent panel | Description | Profile |
|---|---|---|---|
| XV | | REXX stem to VSAM | FILEM.OTHER.ALL |

*Table 3. File Manager function to profile name cross-reference*

| Function | Equivalent panel | Description | Profile |
|---|---|---|---|
| (none) | | Using BLP for tape processing | FILEM.TAPE.BLP |
| (none) | | Handling full disk packs | FILEM.DISK.FULLPACK |
| BT | 4.7 | Create Tape File | FILEM.TAPE.OUTPUT |
| DB | 5.1 | Disk Browse | FILEM.DISK.INPUT |
| DCN | | Disk to console | FILEM.DISK.INPUT |
| DP | 5.3 | Disk Print | FILEM.DISK.INPUT |
| DRS | 5.4 | Disk Record Scan | FILEM.DISK.INPUT |
| DSX | 5.6 | Data Set Extents | FILEM.DISK.INPUT |
| DTE | 5.2 | Disk Track Edit | FILEM.DISK.UPDATE |
| EOF | 5.5 | Write EOF Record | FILEM.DISK.UPDATE |
| ERT | 4.13 | Erase Tape | FILEM.TAPE.UPDATE |
| EVC | 4.2.8 | Exported Stacked Volume Copy | FILEM.TAPE.DUPLICATE |
| EVL | 4.14 | Exported Stacked Volume List | FILEM.TAPE.INPUT |
| HT | | HFS to tape | FILEM.TAPE.OUTPUT |
| INT | 4.12 | Initialize Tape | FILEM.TAPE.UPDATE |
| LMU | 2 | Load module edit/update | FILEM.LOADMOD.UPDATE |
| OE | 6.5 | Object Erase | FILEM.OAM.UPDATE |
| OO | 6.6.5 | Object to Object | FILEM.OAM.OUTPUT |
| OU | 6.4 | Object Update | FILEM.OAM.UPDATE |
| **QO**, SO, TLO | 6.6.4 | Sequential Data to Object | FILEM.OAM.OUTPUT |
| **QT**, ST, STP | 4.2.7 | Sequential Data to Tape | FILEM.TAPE.OUTPUT |
| TB | 4.1 | Tape Browse | FILEM.TAPE.INPUT |
| TC | | Tape to card | FILEM.TAPE.INPUT |
| TCN | | Tape to console | FILEM.TAPE.INPUT |
| TDL | | Tape data set list | FILEM.TAPE.INPUT |
| TH | | Tape to HFS | FILEM.TAPE.INPUT |
| TLB | 4.8 | Tape Label Display | FILEM.TAPE.INPUT |
| TLT | 4.2.3 | Tape to Labeled Tape | FILEM.TAPE.DUPLICATE |
| TMP | 4.6 | Tape Map | FILEM.TAPE.INPUT |
| TP | 4.5 | Tape Print | FILEM.TAPE.INPUT |
| **TQ**, TS, TSQ | 4.2.5 | Tape to Sequential Data | FILEM.TAPE.INPUT |
| TRL | 4.4 | Tape Record Load | FILEM.TAPE.UPDATE |
| TRS | 4.10 | Tape Record Scan | FILEM.TAPE.INPUT |
| TT | 4.2.1 | Tape to Tape (copy) | FILEM.TAPE.DUPLICATE |
| TTC | 4.9 | Tape to Tape Compare | FILEM.TAPE.INPUT |

*Table 3. File Manager function to profile name cross-reference  (continued)*

| Function | Equivalent panel | Description | Profile |
|---|---|---|---|
| TTR | 4.2.2 | Tape to Tape Reblocked | FILEM.TAPE.DUPLICATE |
| TU | 4.3 | Tape Update | FILEM.TAPE.UPDATE |
| **TV**, TVS | 4.2.4 | Tape to VSAM | FILEM.TAPE.INPUT |
| TX | | Tape to REXX Variable | FILEM.TAPE.INPUT |
| VO | 6.6.3 | VSAM to Object | FILEM.OAM.OUTPUT |
| VRU | | Update VSAM record | FILEM.VSAM.UPDATE |
| **VT**, VTP | 4.2.6 | VSAM to Tape | FILEM.TAPE.OUTPUT |
| VU | 5.7 | VSAM Update | FILEM.VSAM.UPDATE |
| WTM | 4.11 | Write Tape Mark | FILEM.TAPE.UPDATE |
| XT | | REXX Variable to Tape | FILEM.TAPE.OUTPUT |
| Others | | Protects all functions not otherwise listed in Table 1 or this table. | FILEM.OTHER.ALL |

- **Chapter 19,** ″**Customizing the File Manager/IMS security environment**″

  In the final section: ″Protected functions in File Manager/IMS″ replace the first paragraph with the following new paragraphs and box. (The table remains the same.)

  The File Manager/IMS functions listed in the table ″File Manager/IMS protected functions″ are protected by default when you receive File Manager/IMS. They are protected by the profile ″FILEM.OTHER.ALL″, and if you want to use these functions, you must first give the users access to these functions, by defining the profile ″FILEM.OTHER.ALL″ to RACF or your equivalent security product. For information on File Manager profiles, and defining them to your security product, refer to chapter 4 of the File Manager Customization Guide.

  > **Important information for users of non-IBM security products**
  > File Manager issues a RACROUTE TYPE=AUTH for the required profile, and if the required profile is not defined, RACF returns RC=4. If RC=4 is returned, File Manager issues a RACROUTE for FILEM.OTHER.ALL. If a higher return code is returned, no RACROUTE is issued for FILEM.OTHER.ALL and the access request fails immediately.
  >
  > You should be aware that not all non-IBM security products issue RC=4 when a RACROUTE TYPE=AUTH for a profile fails; in this case no RACROUTE is issued for FILEM.OTHER.ALL and the access request fails immediately. If this applies to your security product, you will have to provide individual profiles for all File Manager/IMS functions listed in the table ″File Manager/IMS Protected functions″.

  Functions described as **(batch)** are only available in batch processing.

# November 2006

Add the following section to Chapter 12, ″Customizing File Manager/DB2″, after the section ″Binding DB2 (required)″:

## Running multiple versions of File Manager/DB2

If you intend to install multiple (different) versions of File Manager/DB2, you need to ensure that each File Manager/DB2 installation is **COMPLETELY INDEPENDENT** of the other(s).

In particular, you must ensure that:
- Different (and unique) plan and package names are used for each File Manager/DB2 installation.
- The File Manager/DB2 plan and packages are bound using the appropriate DBRM modules for each version of File Manager/DB2.
- Each version of File Manager/DB2 uses a unique FMN2POPT module, and that module must specify the correct plan name for that version of File Manager/DB2.
- When File Manager/DB2 is used to access remote DB2 systems, the plan and packages for each File Manager/DB2 version must be bound on every DB2 system that might be accessed using File Manager/DB2.

The notes below cover the situation where an existing version (say version n) of File Manager/DB2 is installed. A new version (say n+1) of File Manager/DB2 is available and will be installed. The two versions of File Manager/DB2 are to be run concurrently while acceptance testing is completed for the new version of File Manager/DB2.

The default plan names are used for the existing version of File Manager/DB2 that is already installed:
- FMN2PLAN
- FMN2GEN

Similarly, the default package names are used for the existing version of File Manager/DB2 that is already installed:
- FMN2PLPK
- FMN2GNPK

For the new version of File Manager/DB2, select new plan and package names, for example:
- FMN2PLNA (instead of FMN2PLAN)
- FMN2GENA (instead of FMN2GEN)
- FMN2PLPA (instead of FMN2PLPK)
- FMN2GNPA (instead of FMN2GNPK)

Note that any new plan/package names can be selected, as long as the new names are different to all other File Manager/DB2 plan/package names already in use.

For the new version of File Manager/DB2, modify the sample FMN2BIND job to specify the amended plan and package names, as shown above. Run the FMN2BIND job on every DB2 system that the new version of File Manager/DB2 will have access to, either via direct or remote connection. Ensure that the DBRM library specified in the new bind job is for the new version of File Manager/DB2.

Modify the sample FMN2POPT job for the new version of File Manager/DB2. Specify the new plan names in the FMN2SSDM macro definition for each DB2 system that File Manager/DB2 will access, either directly or remotely, as shown below:

```
FMN2SSDM  SSID=XXXX,PLAN=FMN2PLNA,PLAN2=FMN2GENA
```

Assemble the new FMN2POPT module, and ensure that is available when a user executes the new version of File Manager/DB2.

## August 2006

### Chapter 7,″Customizing File Manager to use library management system libraries″

Completely replace the sections in this chapter, from the start, up to ″Capabilities provided by File Manager via FMNCRAEX″, with the following:

File Manager provides the facility to allow you to access source code stored in either library management (LM) or source code management (SCM) systems.

The difference between library management systems and source code management systems is not well defined, but in general, SCM systems tend to provide software packaging, version management of packages, and build management in addition to the typical LM functions. LM systems tend to be simpler and focus on efficiently storing data (files) in a library and managing versions or levels of the individual files. An example of a library management system is CA-Panvalet. An example of a source code management system is IBM's SCLM.

In this chapter both LMs and SCMs are referred to as library management systems or LMSs.

There are three possible scenarios to accessing source code in LMSs:
* the LMS is CA-Panvalet.
* the LMS (not CA-Panvalet) provides a SUBSYS interface.
* the LMS (not CA-Panvalet) does not provide a SUBSYS interface.

If you want to use File Manager to access COBOL copybooks and PL/I include books stored in LMS libraries, the customization tasks you must perform are different, depending on which of the above scenarios applies to your LMS. The possible customization tasks are listed in Table 4. Perform only the tasks that apply to your LMS.

*Table 4. Summary of steps for customizing File Manager to use LMS libraries*

| Step | Description |
| --- | --- |
| __ 1 | Determine if your LMS provides a SUBSYS interface. |
| __ 2 | Set the LMS and LMSUBSYS options, as necessary. |
| __ 3 | Ensure that the correct release of Language Environment is available to File Manager. |
| __ 4 | Determine if your LMS requires your exit to run below the 16MB line. |
| __ 5 | Write your own File Manager exit and link edit it into a load library available to File Manager, as load module FMNCRAEX. |

**Accessing source code in CA-Panvalet libraries**:

File Manager provides an interface to CA-Panvalet, enabling you to use File Manager to work with COBOL copybooks and PL/I include books stored in CA-Panvalet libraries. Before you can do this, you must change the default options to specify one of:
* **LMS=PANVALET**

- **LMS=(USERLMS,PANVALET)**
- **LMS=(PANVALET,USERLMS)**

If you want to access only CA-Panvalet libraries, specify **LMS=PANVALET**.

For information about the LMS option, see Appendix A. For more information on how to change the default options, see ″Changing the default options″, in chapter 3.

If you do not change the LMS option to one of the above values, you will receive an error message if you attempt to access copybooks in a CA-Panvalet library.

**Notes:**

1. This support is provided for CA-Panvalet release 14 and above.
2. Ensure that your CA-Panvalet library is defined with DSORG=DA.

**Accessing source code in an LMS with a SUBSYS interface**

If your LMS provides a SUBSYS interface, you need to perform tasks #2 and #3 from Table 4 on page 47, as described here.

- Set the LMSUBSYS option to the value for the SUBSYS interface provided by your LMS vendor, (for example, LMSUBSYS=LAM).
- Ensure that the LMS option is set to NO.
- Ensure Language Environment is available to File Manager, if you have not already done so.

  You can do this by one of the following methods:
  - Add the Language Environment runtime library, SCEERUN, to your concatenated LINKLIST.
  - Place SCEERUN in the STEPLIB DD statement of the logon proc.
  - Your users can use the TSO command, TSOLIB, to add SCEERUN to the search list.

**Note:** You cannot access source code from an LMS with a SUBSYS interface and also from CA-Panvalet. The two are mutually exclusive.

**Accessing source code in an LMS without a SUBSYS interface**

If your LMS does not provide a SUBSYS interface, you need to perform tasks #2, #3, #4, and #5, from Table 4 on page 47, as described here.

- Set the LMSUBSYS option to a blank (the IBM-supplied default in FMN0POPT.)
- Set the LMS option to USERLMS.
- Ensure Language Environment is available to File Manager, if you have not already done so.

  You can do this by one of the following methods:
  - Add the Language Environment runtime library, SCEERUN, to your concatenated LINKLIST.
  - Place SCEERUN in the STEPLIB DD statement of the logon proc.
  - Your users can use the TSO command, TSOLIB, to add SCEERUN to the search list.
- Code and link your own user exit, a load module named FMNCRAEX, into the File Manager load library. If you have installed File Manager in the default libraries this will be FMN.SFMNMOD1. You can access source code in multiple LMSs through a single FMNCRAEX exit.

If you are using an LMS without a SUBSYS interface, you cannot run File Manager APF-authorized.

Sample user code for FMNCRAEX, written in COBOL, is distributed in the File Manager sample library, FMN.SFMNSAM1. The following sections discuss the functions and requirements for FMNCRAEX. Appendix D, discusses the sample exit itself.

# Appendix A, File Manager options

### LMS option

Replace the description of the LMS option with the following:

LMS
Specifies whether or not File Manager will use COBOL copybooks or PL/I include books stored in CA-Panvalet libraries or other Library Management System libraries.

NO
File Manager will not use any library management system libraries.

PANVALET
File Manager will use COBOL copybooks or PL/I include books stored in CA-Panvalet libraries.

USERLMS
File Manager will use COBOL copybooks or PL/I include books stored in a library management system, other than CA-Panvalet, that does **not** provide a SUBSYS interface.

(PANVALET,USERLMS) or (USERLMS,PANVALET)
File Manager can use COBOL copybooks or PL/I include books stored in CA-Panvalet together with any other library management system that does **not** provide a SUBSYS interface. (PANVALET,USERLMS) and (USERLMS,PANVALET) are equivalent.

**Note:** You cannot specify LMS=PANVALET together with LMSUBSYS=xxxx, since you cannot access source from CA-Panvalet together with source from an LMS that provides a SUBSYS interface.

### LMSUBSYS option

Replace the description of the LMSUBSYS option with the following:

**LMSUBSYS**
specifies that File Manager will, when accessing COBOL copybooks or PL/I include books, attempt the access via an I/O subsystem using the DFSMSdfp SUBSYS=xxxx allocation parameter.

This access is only available to copybooks and include books that exist in data sets controlled by the LMS. Data set organizations of BDAM, VSAM, and PS (sequential) are supported.

This facility is used to access COBOL copybooks or PL/I include books in OEM library management system data sets.

Specify *xxxx* as provided by your library management system vendor.

**Note:** You cannot specify LMSUBSYS=xxxx if you have specified LMS=PANVALET, LMS=(PANVALET,USERLMS), or

LMS=(USERLMS,PANVALET), since you cannot specify a combination of LMS and LMSUBSYS that equates to using CA-Panvalet with another LMS that provides a SUBSYS interface.

## June 2006

### Chapter 2, ″**Customizing the Operating Environment for File Manager**″

Section, ″Customizing to use DFSORT to improve File Manager performance″

This section describes the requirements for DFSORT when used with File Manager to improve performance.

All references in this section to releases of DFSORT and required PTFs should include both the following:
- DFSORT R14 with PTFs UQ90054 and UQ84945 applied.
- z/OS DFSORT V1R5 with PTF UQ84946 applied.

### Appendix C, ″**File Manager/IMS options**″

Macro FMN1POPI, option DYNPRFN

In the last paragraph describing the use of DYNPRFN, remove the first sentence, so that this paragraph now reads:

″If you are not using dynamic PSBs in this IMS subsystem, delete this option, and also the DYNACB option, from this FMN1POPI macro.″

# User's Guide and Reference (SC18-9655-01)

## December 2008

Chapter 16, "Functions", section "External REXX functions"

- In the list of functions at the beginning of the section:
  - In the description for RSTR_OUT, add the sentence:
    "(Can be used in FASTREXX condition expressions.)"
  - In the description for SAVE_OUT, add the sentence:
    "(Can be used in FASTREXX condition expressions.)"
- In the subsection, "RSTR_OUT", after the syntax diagram add the sentence:
  "(Can be used in FASTREXX condition expressions.)"
- In the subsection, "SAVE_OUT", after the syntax diagram add the sentence:
  "(Can be used in FASTREXX condition expressions.)"

## April 2007

**Chapter 17, "Functions"**

Section, "SET (Set Processing Options)"

- In the description for **ASCII**, change the last paragraph

"This processing option applies to all tape input and output functions except: Initialize Tape, Tape Browse, Tape Label Display, Tape to Labeled Tape, Tape to Tape Compare and Tape Update. For Tape Label Display, ASCII translation is automatically performed when needed. For Initialize Tape and Tape Browse, you can specify ASCII input or output when invoking the function. For the other functions, translation is not supported by File Manager."

to

"This processing option applies to all tape input and output functions except: Initialize Tape, Tape Browse, Tape Label Display, and Tape Update. For Tape Label Display, ASCII translation is automatically performed when needed. For Initialize Tape and Tape Browse, you can specify ASCII input or output when invoking the function. For the other (non-tape) functions, ASCII translation is not supported by File Manager."

## March 2007

**Chapter 15, "Panels and fields"**

In the section "Edit Entry panel", under "**Inplace edit**" replace

"You must select **Inplace edit** when using any of the following:..."

with

"**Inplace edit** is implied when using any of the following:..."

**51**

# December 2006

**Chapter 18,** ″**File Manager messages**″

Add the following new message:

**FMN0052I**
> The template has been corrupted and must be created again before it can be used. Reason code rc.

**Explanation:**
> The template has been corrupted and cannot be used.

**User Response:**
> Recreate the template. If the problem still occurs, save any system error information and contact your system support.

# November 2006

**Chapter 17,** ″**Functions**″

Section, ″DSU (Data Set Update) - batch only″

- In the description for **KEY=key (VSAM only)**, change the following text

  ″A key for KSDS records, or a slot number for RRDS records. The maximum key length is 30 characters. The first record with a key or slot value greater than or equal to key is the first record printed. If you omit the key and skip values, printing begins with the first record in the data set.″

  to

  ″A key for KSDS records, or a slot number for RRDS records. The maximum key length is 30 characters. The first record with a key or slot value greater than or equal to key is the first record updated. If you omit the key and skip values, updating begins with the first record in the data set.″

- In the description for **PROC=proc**, change the following text

  ″Member name of a REXX procedure that you want to use to process each record before it is printed, or an asterisk (*) to indicate the REXX procedure is inline. If you specify a member name, you must define an FMNEXEC ddname that identifies the PDS containing the member. If you specify *, the procedure is read from SYSIN immediately following the control statement for the current function. The inline procedure is terminated by a record containing a slash and a plus sign (/+) in columns 1-2. For more information about using REXX procedures to process records before they are printed, see...″

  to

  ″Member name of a REXX procedure that you want to use to process each record before it is updated, or an asterisk (*) to indicate the REXX procedure is inline. If you specify a member name, you must define an FMNEXEC ddname that identifies the PDS containing the member. If you specify *, the procedure is read from SYSIN immediately following the control statement for the current function. The inline procedure is terminated by a record containing a slash and a plus sign (/+) in columns 1-2. For more information about using REXX procedures to process records before they are updated, see...″

**Chapter 12,** ″**Using UNIX System Services and the Hierarchical File System**″

Section, ″Using File Manager functions in batch jobs″

*Move* the first paragraph (starting "Some functions which do not require user interaction...") and the example that follows, up to but excluding the paragraph starting "To use a File Manager function in a batch job...", *to* the end of the previous section, "Invoking File Manager panels from REXX procedures or TSO clists" under a new heading, "**Invoking File Manager functions from a REXX procedures or CLIST in /BATCH mode**".

**August 2006**

**Chapter 6,** "**Changing records in data sets**"

Section, "**Editing records within data sets**":

After the section, "Splitting and joining lines", add a new section:

# September 2006

**Chapter 6,** "**Changing records in data sets**"

Section, "Editing records within data sets"

Add the following new section at the end:

## Changing the length of a record

For those files that support variable-length records, you can change the length of the record by entering the zoom mode from a CHAR, HEX, or LHEX display format.

In zoom mode, the length of the record is displayed which you can change to the required value.

When you increase the record length in this way, the new bytes added to the record are blanks.

# User's Guide and Reference for DB2 Data (SC18-9656-01)

**May 2006**

- Chapter 13, "**Working with SQL statements**"

  Section, "**Should you use basic or advanced SQL prototyping?**", subsection, "**Using basic SQL prototyping**":

  Replace the first paragraph with:

  You use the Basic SELECT Prototyping (primary) panel to specify up to 15 DB2 objects (for a definition of "DB2 object" in this context, see Chapter 16, "FM/DB2 panels and fields", section "Basic SELECT Prototyping panel") to be used to construct a FROM clause as the starting point for prototyping a SELECT statement.

- Chapter 16, "**FM/DB2 panels and fields**"

  Section, "**Basic SELECT Prototyping panel**":

  Replace the first paragraph with:

  You use the Basic SELECT Prototyping (primary) panel to specify up to 15 DB2 objects (see note below) to be used to construct a FROM clause as the starting point for prototyping a SELECT statement.

  **Note:** In the context of the Basic SQL prototyper, a "DB2 object" can be a table, a view, a synonym, or an alias that refers to a DB2 object at the same location. Aliases that refer to DB2 objects at remote locations are not supported by the Basic SQL prototyper. In what follows, "table" is used as a generic term for any DB2 object that matches this definition.

## User's Guide and Reference for IMS Data (SC18-9657-01)

None applicable.

# Part 3. Appendixes

# Index

## P

**IBM** ®

Printed in USA

Spine information:

IBM

File Manager for z/OS V6R1

**Addenda to V6R1 User's Guides and Customization Guide**