StarTool® FDM 7.8.1 Readme

Updated 19 June 2017

General Information

This Readme file contains information you should know before you install or use StarTool FDM 7.8.1. For your convenience, this Readme file provides cumulative information for FDM 7.8, 7.8.0.01, 7.8.0.02, 7.8.0.03, and 7.8.1. The information that pertains only to FDM 7.8.1 is marked with change bars in the left margin.

Product Description

StarTool® FDM (File and Data Manager) is a multi-purpose file and data management utility for IBM mainframe systems. It provides an ISPF-based, menu-driven, integrated interface to a variety of file editors and data management tools for PDS, PDSE (V1 and V2), VSAM, IAM, sequential, extended sequential, direct-access, IMS, and DB2 files. A TSO command-line interface is supported concurrently, and bulk file updates in batch mode are also supported.

Licensing Considerations

License SERtificates from versions prior to FDM 7.7 are not compatible with StarTool FDM 7.8 and later releases.

To obtain the XML licensing key needed to make the software operational, please open an incident on the Micro Focus SupportLine website at:

`https://supportline.microfocus.com/`

As part of your request, please copy and paste your CPU information into the Support incident. To obtain the required CPU information, run the following command from the console on all LPARs hosting your product's software:

```
/D M=CPU
```

Accessing the Documentation

- All StarTool FDM documentation is provided in Adobe® Portable Document Format (PDF) and may be downloaded from the Documentation page of the Micro Focus SupportLine website at:

  `https://supportline.microfocus.com/productdoc.aspx`

  No user ID or password is needed to access the Documentation page.

- Adobe® Reader® is available for free download at `http://get.adobe.com/reader/`. Version 9 or higher is recommended.

- Additions and corrections to documentation can be issued after the product release date. Updated Readme files and user manuals are published on the Documentation page of the Micro Focus SupportLine website with the rest of the documentation for the product release. Updated Readme files and user manuals are flagged with an **Updated** date in the file name or description.

NOTE Change bars in the left margin of StarTool FDM documentation identify substantive changes from the last published version, unless otherwise noted.

Searching the Documentation

There is no cross-book index for the StarTool FDM documentation suite. You can use the Advanced Search facility in Adobe Reader to search all FDM manuals in the same or subordinate folders for information that you want.

Customer Support

- **Online Support** — The Micro Focus SupportLine website at `http://supportline.microfocus.com` provides an easy way for customers to manage their support cases, download software patches, or search the Knowledgebase for the latest technical information about our products at any time.

- A user ID and password are required to access the Micro Focus SupportLine website. To obtain a user ID and password, sign up online or contact Customer Care by email or telephone.

- **Telephone Support** — Telephone contact numbers for Customer Care are posted by region at the following URL:

  `https://www.microfocus.com/serena/support/`

  No user ID or password is needed to access Customer Care contact information.

- **Email Support** — Customers can send support requests by email to `customercare@microfocus.com.`

- **Authorized Support Representative** — Some support contracts specify that support calls will be centrally managed within the customer organization. This practice enables related issues to be aggregated across multiple customer sites or functional groups, and it ensures accountability for the satisfactory resolution of open issues. You should determine whether you have such a support contract before contacting Customer Care. If you do have such a contract, submit your support request to the designated support manager within your organization and arrange for that person to contact Customer Care.

New Features

StarTool FDM 7.8.1

StarTool FDM 7.8.1 contains the following new features. (See "Fixes in FDM 7.8.1" on page 23 for a link to the defects that are fixed in StarTool FDM 7.8.1.)

FDM Supports IAM Data Sets

StarTool FDM 7.8.1 can display information about Innovation Access Method (IAM) data sets:

- If you have IAM data sets and you want to display information about them, you must include the IAM libraries in your FDM load library concatenation.
- For the TSO CLISTS for StarTool FDM and for the batch StarBat JCL, the SYSLIB DDN must be defined and use the same IAM load library.

When the feature to support IAM data sets is ON, the information about IAM data sets is displayed with a call to an IAM exit as if the file is a VSAM data set. This display leaves many fields unpopulated, but the basic information is displayed.

When the feature is OFF, IDCAMS is called and the call returns the data from the system catalog as if the data set is a physical sequential data set.

The following command acts like a toggle to turn the feature ON and OFF:

```
CONTROL TESTIAM
```

That is, if the feature is ON, the command turns the feature OFF. If the feature is OFF, the command turns it ON. The default for this feature is ON.

The status of the feature is displayed in the log. For example:

```
PDS030I Global operands: Test IAM is active.
```

VUSE Subcommand Is Updated for EAV Devices

The VUSE (volume usage) subcommand has been updated to display additional information about Extended Access Volume (EAV) devices.

FDM Supports COBOL Version 6.1

FDM 7.8.1 supports COBOL version 6.1.

Content and Format of the HISTORY Subcommand Are Enhanced

The HISTORY subcommand is enhanced to support up to three translators per CSECT. The format changes are illustrated below. In the examples below, notice that the translator name (shown in red) has been moved to the left in the display:

New Format with 3 Translators:

```
DSN=USERABC.C2912472.FEB16.ABOLOAD,VOL=SER=SRSM36  MEM=:  --------------------------------------
>------->HIST :

** HISTORY  TIG6733
PDS060I Translator history by CSECT -
B_TEXT  :TIG6733  COBOLzOS  2011/06/17 5655S7100  V42M00   2034/01/00 ..TIG6733  V09M18 ...
B_TEXT  :CEESTART ASMA90    2009/03/18 569623400  V01M06
B_TEXT  :CEEBLLST ASMA90    2009/03/18 569623400  V01M06
B_TEXT  :CEEBTRM  ASMA90    2009/03/18 569623400  V01M06
B_TEXT  :CEEBETBL ASMA90    2009/03/18 569623400  V01M06
B_TEXT  :CEEARLU  ASMA90    2009/03/18 569623400  V01M06   2009/03/18 PL/X-390   V02M04
B_TEXT  :CEEBPIRA ASMA90    2009/03/18 569623400  V01M06   2009/03/18 PL/X-390   V02M04
B_TEXT  :CEEBINT  ASMA90    2009/03/18 569623400  V01M06   2009/03/18 PL/X-390   V02M04
B_TEXT  :CEEBPUBT ASMA90    2009/03/18 569623400  V01M06
B_TEXT  :CEECPYRT ASMA90    2009/03/18 569623400  V01M06   2009/03/18 PL/X-390   V02M04
B_TEXT  :CEESG005 ASMA90    2009/03/17 569623400  V01M06   2009/03/17 PL/X-390   V02M04
B_TEXT  :IGZCBSO  ASMA90    2009/03/17 569623400  V01M06   2009/03/17 PL/X-390   V02M04
PDS260I CSECT____VER_TEST_SSRANG_OPT_CMPR2_ZWB_NUMPR_TRUNC_RES_RENT_DYNAM_DATA__ALL OPTIONS___
PDS260I TIG6733   OS  NO   NO     NO  NO    YES NOPFD (OPT) YES YES  YES   31
64AC740C1080000052000B0B8028405000000000000800000000008400
PDS061I AMASPZAP update history by CSECT -
PDS067I Member has 1 IDR blocks with space for 19 IDR entries
PDS068I 0 IDR entries are in use; 19 are available for use
PDS062I User-supplied update history by CSECT -
B_TEXT  :TIG6733  2011/06/17 CXTIG6733 .I...L........................
B_TEXT  :CEESG005 2009/03/17 RSI90762250
B_TEXT  :IGZCBSO  2009/03/17 RSI90762118
PDS064I Last link-edited on 2017/02/10 10:11 by 5695PMB01-BINDERz/OS V02 M01
******************************* Bottom of data ************************************************
```

New Format with Only 2 Translators:

```
DSN=USERDEF.C226877.LPDSE,VOL=SER=SRSMF2  MEM=:  -----------------------------------------------
>------->HISTORY  TIG6733
PDS060I Translator history by CSECT -
B_TEXT  :TIG6733  COBOLzOS  2011/06/17 5655S7100  V42M00   2034/01/00 ..TIG6733  V09M18
B_TEXT  :CEEBPUBT ASMA90    2009/03/18 569623400  V01M06
B_TEXT  :CEESTART ASMA90    2009/03/18 569623400  V01M06
B_TEXT  :CEEBINT  ASMA90    2009/03/18 569623400  V01M06   2009/03/18 PL/X-390   V02M04
B_TEXT  :CEEBETBL ASMA90    2009/03/18 569623400  V01M06
B_TEXT  :CEEBPIRA ASMA90    2009/03/18 569623400  V01M06   2009/03/18 PL/X-390   V02M04
B_TEXT  :CEECPYRT ASMA90    2009/03/18 569623400  V01M06   2009/03/18 PL/X-390   V02M04
B_TEXT  :CEEBTRM  ASMA90    2009/03/18 569623400  V01M06
B_TEXT  :CEEBLLST ASMA90    2009/03/18 569623400  V01M06
B_TEXT  :CEEARLU  ASMA90    2009/03/18 569623400  V01M06   2009/03/18 PL/X-390   V02M04
B_TEXT  :CEESG005 ASMA90    2009/03/17 569623400  V01M06   2009/03/17 PL/X-390   V02M04
B_TEXT  :IGZCBSO  ASMA90    2009/03/17 569623400  V01M06   2009/03/17 PL/X-390   V02M04
PDS260I CSECT____VER_TEST_SSRANG_OPT_CMPR2_ZWB_NUMPR_TRUNC_RES_RENT_DYNAM_DATA__ALL OPTIONS___
PDS260I TIG6733   OS  NO   NO     NO  NO    YES NOPFD (OPT) YES YES  YES   31
64AC740C1080000052000B0B8028405000000000000800000000008400
```

```
PDS061I AMASPZAP update history by CSECT -
PDS067I Member has (n/a) IDR blocks with space for (n/a) IDR entries
PDS068I (n/a) IDR entries are in use; (n/a) are available for use
PDS062I User-supplied update history by CSECT -
B_TEXT  :TIG6733  2011/06/17 CXTIG6733 .I ..L........
B_TEXT  :CEESG005 2009/03/17 RSI90762250
B_TEXT  :IGZCBSO  2009/03/17 RSI90762118
PDS064I Last link-edited on 2017/02/10 10:11 by 5695PMB01-BINDERz/OS V02 M01
****************************** Bottom of data *************************************************
```

Prior Format:

```
DSN=USERDEF.C226877.LPDSE,VOL=SER=SRSMF2  MEM=:  -------------------------------------
>------>HISTORY  TIG6733
PDS060I Translator history by CSECT -
B_TEXT  :TIG6733  2011/06/17 5655S7100-COBOLzOS  V42 M00 2034/01/00  ..TIG6733  V09M18
B_TEXT  :CEEBPUBT 2009/03/18 569623400-ASMA90    V01 M06
B_TEXT  :CEESTART 2009/03/18 569623400-ASMA90    V01 M06
B_TEXT  :CEEBINT  2009/03/18 569623400-ASMA90    V01 M06 2009/03/18  PL/X-390   V02M04
B_TEXT  :CEEBETBL 2009/03/18 569623400-ASMA90    V01 M06
B_TEXT  :CEEBPIRA 2009/03/18 569623400-ASMA90    V01 M06 2009/03/18  PL/X-390   V02M04
B_TEXT  :CEECPYRT 2009/03/18 569623400-ASMA90    V01 M06 2009/03/18  PL/X-390   V02M04
B_TEXT  :CEEBTRM  2009/03/18 569623400-ASMA90    V01 M06
B_TEXT  :CEEBLLST 2009/03/18 569623400-ASMA90    V01 M06
B_TEXT  :CEEARLU  2009/03/18 569623400-ASMA90    V01 M06 2009/03/18  PL/X-390   V02M04
B_TEXT  :CEESG005 2009/03/17 569623400-ASMA90    V01 M06 2009/03/17  PL/X-390   V02M04
B_TEXT  :IGZCBSO  2009/03/17 569623400-ASMA90    V01 M06 2009/03/17  PL/X-390   V02M04
PDS260I CSECT____VER_TEST_SSRANG_OPT_CMPR2_ZWB_NUMPR_TRUNC_RES_RENT_DYNAM_DATA__ALL OPTIONS___
PDS260I TIG6733  OS  NO   NO     NO  NO    YES NOPFD (OPT) YES YES  YES   31
64AC740C108000005200B0B802840500000000000008000
PDS061I AMASPZAP update history by CSECT -
PDS067I Member has (n/a) IDR blocks with space for (n/a) IDR entries
PDS068I (n/a) IDR entries are in use; (n/a) are available for use
PDS062I User-supplied update history by CSECT -
B_TEXT  :TIG6733  2011/06/17 CXTIG6733 .I ..L........
B_TEXT  :CEESG005 2009/03/17 RSI90762250
B_TEXT  :IGZCBSO  2009/03/17 RSI90762118
PDS064I Last link-edited on 2017/02/10 10:11 by 5695PMB01-BINDERz/OS V02 M01
*************************************** Bottom of data ****************************************
```

MODEL Subcommand Is Enhanced To Create a Non-VSAM Data Set from a VSAM Data Set Model

The MODEL subcommand enables you to define a new data set based on a current data set that is used as a model. Options can be used to reformat the new data set.

The MODEL subcommand can have the following formats:

MODEL RESET|PROFILE|BLKSIZE

where:

Parameter	Description
RESET	Clear and reset saved information for display. As you enter multiple MODEL requests for the same data set, the information is saved from request to request. You can use the RESET option to remove saved information on the MODEL panel. If you change the current data set, the saved data is reset.
PROFILE	Use the data from variables saved in your profile to populate the MODEL panel. You must have saved the PROFILE values from a prior MODEL display. To save PROFILE values, enter the RCHANGE (PF6) command to save information on the MODEL panel for a new MODEL request that specifies the PROFILE option.
BLKSIZE	Display the BLKSIZE panel and use the current allocations for the display.

or:

MODEL [*dsn*] [*type*]

where:

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td><i>dsn</i></td><td>Data set name (can be fully qualified). If entered, the specified data set name will be used in the target model panel to create the new data set.</td></tr>
<tr><td><i>type</i></td><td>The data set type that is to be created. The specified type is used to select the appropriate output panel to use to define the output data set. Valid conversions are:
<ul>
<li>If the current data set is a VSAM data set, you can create a new VSAM, PS, or PO type data set.</li>
<li>If the current data set is a PS or PO type data set, you can create other PS or PO type data sets.</li>
</ul>
The <i>type</i> parameter can have one of the following values. If <i>type</i> is omitted, the format of the current input data set is used for the output data set:
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>AIX</td><td>VSAM alternate index.</td></tr>
<tr><td>ESDS</td><td>VSAM entry sequenced data set.</td></tr>
<tr><td>F</td><td>Fixed unblocked data set.</td></tr>
<tr><td>FB</td><td>PS or PO fixed blocked data set.</td></tr>
<tr><td>KSDS</td><td>VSAM key sequenced data set.</td></tr>
<tr><td>LDS</td><td>VSAM linear data set.</td></tr>
<tr><td>RRDS</td><td>VSAM relative record data set.</td></tr>
<tr><td>VB</td><td>Variable blocked data set.</td></tr>
<tr><td>VBS</td><td>Variable spanned blocked data set.</td></tr>
<tr><td>VRRDS</td><td>Variable relative record data set.</td></tr>
<tr><td>U</td><td>Unblocked data set.</td></tr>
</table>
</td></tr>
</table>

The MODEL subcommand displays your current data set allocation; it also serves as a front end for the CREATE subcommand. You can overtype any of the input data fields to change allocation parameters. You can also use the BLKSIZE optimization service by entering B as a primary command from the target model panel.

The MODEL subcommand does not have a COPY option. However, the output panel that is displayed for the output data set type allows you to specify a COPY option after the data set is created. You can specify record-selection criteria when the COPY panel is displayed. Once created, the copy is displayed.

NOTE Copy operations are not supported for VSAM LDS.

The MODEL subcommand does not change the current data set that is used as the model.

FDM Supports PDSE Version 2 Data Sets

FDM 7.8.1 supports Partitioned Data Set Extended (PDSE) version 2 data sets. Many FDM panels have been modified to enable PDSE version 2 data sets to be defined and to display their attributes. Additionally, the following commands and subcommands have been enhanced to support PDSE version 2 data sets:

COMBINE subcommand
COMPRESS subcommand
CONTROL subcommand
COPY subcommand
CREATE subcommand
FIXPDS subcommand
MODEL command

Additional Copy Type Operations Have Been Added to the StarBat Data Set Prompt Panel

The following copy type operations have been added to the StarBat Data Set Prompt (PDSSTB1) panel:

Copy Type Operation	Description
COPYMBR	Copies one or more data set members based on their contents. This operation can create empty members if no records are selected.
UPDATEMBR	Updates one or more data set members based on their contents.

PBROWSE Function Is Enhanced

The Find command is now supported on a PBROWSE (parallel browse) function against a sequential data set with RECFM=U.

PDS051I Messaging Has Changed

The following message has changed:

```
PDS051I memname was {moved/copied/replaced/combined/separated}; [input=count] output=count
```

Explanation: Issued by the COMBINE, COPY, DUP, FIXPDS, REPRO or SEPARATE subcommand to identify members copied or moved, their status, and record counts.

Sample messages are:

```
PDS051I Data set was copied; input=16; output=16
PDS051I //DD010    Member:CR79520A was copied; output=16
PDS051I //DD010   was copied; output=16
```

where DD01O could be the default output DD name or the DD name used in a WRITE= parm. It is created when the data set is closed.

In the following example, the messages indicate that data from two input members was copied to a single member in the output data set:

```
PDS051I CR79520A was copied; input=16; output=16
PDS051I CR79520B was copied; input=16; output=16
PDS051I //DD010    Member:CR79520D was copied; output=32
```

BINDJCL Job Is Changed

The BINDJCL member of the STRFDM.V7R8M1.CNTL distribution library has been changed to bind a package instead of a plan. This change applies to DB2 version 10 and later.

StarTool FDM 7.8.0.03

StarTool FDM 7.8.0.03 contains the following new features. (See "Fixes in FDM 7.8.0.03" on page 24 for a link to the defects that are fixed in StarTool FDM 7.8.0.03.)

StarTool FDM Fully Supports IAM Data Sets

StarTool FDM fully supports Innovation Access Method (IAM) data sets from Innovation Data Processing.

StarBat Option Messages Are Enhanced When IAM Data Sets Are Accessed

The text of the PDS200I and STRR05I messages has been enhanced to identify the VSAM data set type when an IAM data set is accessed with the FDM StarBat Option.

Example message output:

```
DD01 COPYREC
 *** End of control statement


PDS200I DISP UNIT     RECFM LRECL BLKSIZE   ALLOCTRK FREETRK SECONDARY DSORG
PDS200I SHR  3390     IAM      80      80   1X    15      13      5 CYL VS-KSDS

STRB02I EXCP input is in use, access as data
STRB05I DDNAME=DD010 DSN=USER1.FDM.IAM.KSDSA2 opened for VSAM output IAM
STRB05I DCB=(VS-KSDS,LRECL=80,BLKSIZE=80),VOL=SER=
  *** End of input data set
PDS051I Data set was copied; input=23; output=23
```

Another example of message output:

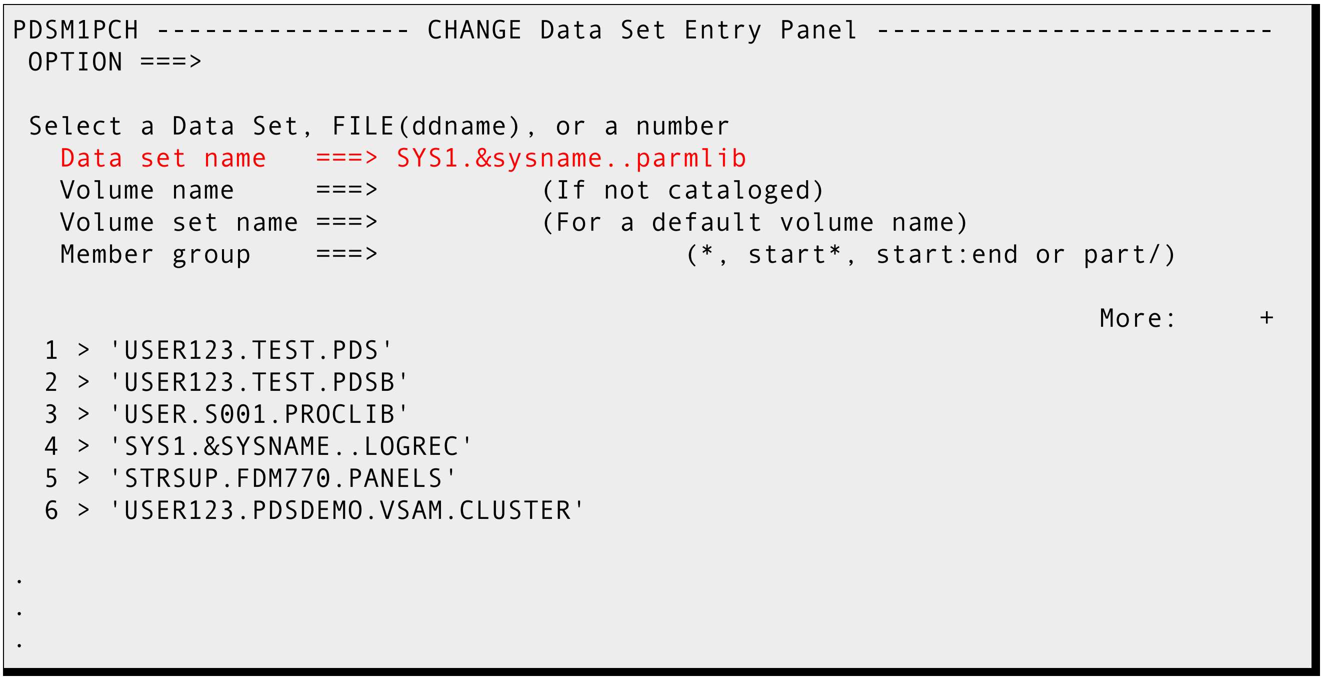
```
DD02 COPYREC
  *** End of control statement


PDS200I DISP UNIT     RECFM LRECL BLKSIZE   ALLOCTRK FREETRK SECONDARY DSORG
PDS200I SHR  3390     IAM      80      80   1X     15      13      1 CYL VS-ESDS

STRB02I EXCP input is in use, access as data
STRB05I DDNAME=DD02O DSN=USER1.FDM.IAM.RRDS opened for VSAM output IAM
STRB05I DCB=(VS-RRDS,LRECL=80,BLKSIZE=80),VOL=SER=

  *** End of input data set
PDS051I Data set was copied; input=23; output=23
```

StarTool FDM 7.8.0.02

StarTool FDM 7.8.0.02 contains the following new feature. (See "Fixes in FDM 7.8.0.02" on page 24 for a link to the defects that are fixed in StarTool FDM 7.8.0.02.)

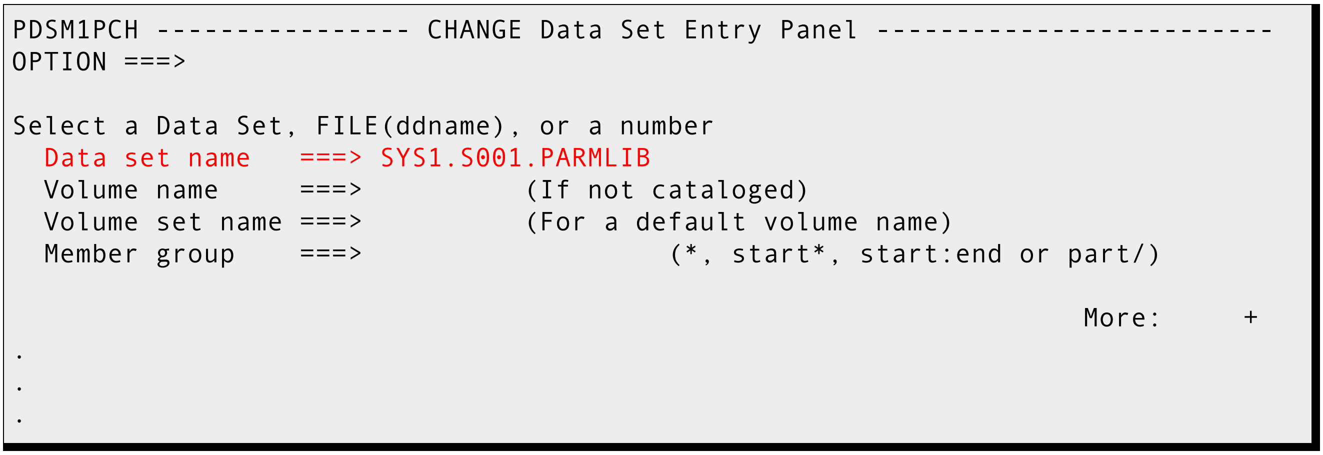
FDM Supports System and User ISPF Variables

FDM now supports the use of all system and user ISPF variables. For example, if you specify the system variable &SYSNAME when you enter a data set name on the Change Data Set Entry (PDSM1PCH) panel:



```
PDSM1PCH ---------------- CHANGE Data Set Entry Panel -------------------------
 OPTION ===>

 Select a Data Set, FILE(ddname), or a number
   Data set name   ===> SYS1.&sysname..parmlib
   Volume name     ===>          (If not cataloged)
   Volume set name ===>          (For a default volume name)
   Member group    ===>                   (*, start*, start:end or part/)

                                                                More:     +
  1 > 'USER123.TEST.PDS'
  2 > 'USER123.TEST.PDSB'
  3 > 'USER.S001.PROCLIB'
  4 > 'SYS1.&SYSNAME..LOGREC'
  5 > 'STRSUP.FDM770.PANELS'
  6 > 'USER123.PDSDEMO.VSAM.CLUSTER'

.
.
.
```

&SYSNAME is resolved to the system where you are currently logged on (S001 in this example) when you press Enter:



```
PDSM1PCH ---------------- CHANGE Data Set Entry Panel -------------------------
OPTION ===>

Select a Data Set, FILE(ddname), or a number
  Data set name   ===> SYS1.S001.PARMLIB
  Volume name     ===>          (If not cataloged)
  Volume set name ===>          (For a default volume name)
  Member group    ===>                   (*, start*, start:end or part/)

                                                                   More:     +
.
.
.
```

StarTool FDM 7.8.0.01

StarTool FDM 7.8.0.01 contains the following new features. (See "Fixes in FDM 7.8.0.01" on page 24 for a link to the defects that are fixed in StarTool FDM 7.8.0.01.)

Enhanced DECODE Subcommand

StarTool FDM 7.8 applied a new encryption/decryption algorithm such that files encrypted prior to FDM 7.8 could not be decrypted with the new decryption algorithms supplied with StarTool FDM 7.8.

StarTool FDM 7.8.0.01 supports an expanded decryption algorithm that allows you to decrypt files that were encrypted prior to FDM 7.8.

Code the key specification of the DECODE subcommand as follows to decrypt a file that was encrypted with a pre-FDM 7.8 ENCODE subcommand:

The key file contains exactly 2 records:

- Record 1 is the key that is used to process the input file.
- Record 2 is not used.

The format of the key in record 1 is:

Item	Column Position	Description
Key Type	1-3	Value is FDM. FDM means the file was encrypted with a pre-FDM 7.8 algorithm.

Item	Column Position	Description
Space	4	Value must be a single space.
Key	5-68	Original hexadecimal key that the pre-FDM 7.8 file was encrypted with, specified in hex characters 0-9, lowercase a-f, or uppercase A-F. Maximum key length is 32 hexadecimal digits (24 hexadecimal bytes). If the pre-FDM 7.8 file key was specified in character format, enclose the original character-format key (up to 16 alphanumeric characters) in single quotes.

Example Key Specifications

- For a character key:

  ```
  FDM ’CIPHER KEY’
  ```

- For a hexadecimal key:

  ```
  FDM aabbccddee0099
  ```

StarTool FDM 7.8

StarTool FDM 7.8 contains the following new features. (See “Fixes in FDM 7.8” on page 24 for a list of defects that have been fixed in FDM 7.8.)

FINDPROC Subcommand

The FINDPROC subcommand searches the active JES procedure library list and returns the names of the libraries that contain the specified JCL procedure.

Example

```
FINDPROC $AUTH LIST
```

returns the following:

```
  PROC00(0001)      DSN=SYS1.SERENA.TCPIP19.PROCLIB
  PROC00(0002)      DSN=SYS1.SERENA.PROCLIB
PDS293I Member found in PROC00(0002) DSNAME='SYS1.SERENA.PROCLIB'
  PROC00(0003)      DSN=SYS1.A004.PROCLIB
  PROC00(0004)      DSN=SYS1.CPZOS21.BLDJ.PROCLIB
  PROC00(0005)      DSN=SYS1.IBM.BLDJ.PROCLIB
  PROC00(0006)      DSN=ADCD.ZOSV1R3.PROCLIB
  PROC00(0007)      DSN=CEE.SCEEPROC
  PROC00(0008)      DSN=CSQ520.SCSQPROC
  PROC00(0009)      DSN=IOE.SIOEPROC
  PROC00(0010)      DSN=EOY.SEOYPROC
  PROC00(0011)      DSN=HLA.SASMSAM1
  PROC00(0012)      DSN=CBC.SCCNPRC
```

```
  PROC00(0013)      DSN=IGY310.SVSC.PROCLIB
  PROC00(0014)      DSN=IEL111.SVSC.PROCLIB
  PROC00(0015)      DSN=FAN130.SVSC.PROCLIB
  PROC00(0016)      DSN=SYS1.PROCLIB
  PROC00(0017)      DSN=USER.A004.PROCLIB
  PROC00(0018)      DSN=USER.PROCLIB
  PROC99(0001)      DSN=IMSV8.PROCLIB
  PROC99(0002)      DSN=IMSV7.PROCLIB
  PROC99(0003)      DSN=SYS1.SERENA.TCPIP19.PROCLIB
  PROC99(0004)      DSN=SYS1.SERENA.PROCLIB
PDS293I Member found in PROC99(0004) DSNAME='SYS1.SERENA.PROCLIB'
  PROC99(0005)      DSN=SYS1.A004.PROCLIB
  PROC99(0006)      DSN=SYS1.CPZOS21.BLDJ.PROCLIB
  PROC99(0007)      DSN=SYS1.IBM.BLDJ.PROCLIB
  PROC99(0008)      DSN=ADCD.ZOSV1R3.PROCLIB
  PROC99(0009)      DSN=CEE.SCEEPROC
  PROC99(0010)      DSN=CSQ520.SCSQPROC
  PROC99(0011)      DSN=IOE.SIOEPROC
  PROC99(0012)      DSN=EOY.SEOYPROC
  PROC99(0013)      DSN=HLA.SASMSAM1
  PROC99(0014)      DSN=CBC.SCCNPRC
  PROC99(0015)      DSN=IGY310.SVSC.PROCLIB
  PROC99(0016)      DSN=IEL111.SVSC.PROCLIB
  PROC99(0017)      DSN=FAN130.SVSC.PROCLIB
  PROC99(0018)      DSN=SYS1.PROCLIB
  PROC99(0019)      DSN=USER.A004.PROCLIB
  PROC99(0020)      DSN=USER.PROCLIB
```

Syntax

```
FINDPROC module [LIST]
                [CHANGE|GO LIB(n)]
```

Aliases

```
FINDP,FINDPR
```

Defaults

```
LIB(1)
```

Operands

module	Name of the target JCL procedure member.
LIST	Lists the procedure libraries that are searched.
CHANGE	If the member is found, perform a CHANGE to that library.
GO	If the member is found, initiate a GO session for that library.
LIB(*n*)	If there are multiple finds, LIB specifies which library to select for the CHANGE or GO action. If you specify a higher number for LIB than the number actually found, the last data set is selected.

> **NOTE** As part of FINDPROC processing, FDM creates and activates console SZFCONSL to issue the $D PROCLIB command, but then inactivates it.

Cylinder-Managed EAV Files Now Processed with EXCP

Files on Extended Address Volumes (EAV) are now processed with EXCP rather than other types of I/O processing.

HISTORY Command Now Recognizes Modules Coded in COBOL 5.1

The HISTORY command now recognizes modules coded in COBOL 5.1 and no longer returns *unknown.

FDM Now Reports if Non-Load and Load PDSs Are LLA-Managed

The Library Lookaside (LLA) facility can manage non-load as well as load PDSs. FDM now issues the following message when a CHANGE or USAGE subcommand is issued if the target data set is a non-load or load library and is LLA-managed:

```
   PDS189I This data set is managed by LLA
```

In previous FDM releases, FDM only issued the PDS189I message for LLA-managed load libraries.

FDM Encryption Processing Is Now Compliant with FIPS 197 AES

FDM 7.8 is compliant with the Advanced Encryption Standard (AES) set forth in Federal Information Processing Standard (FIPS) 197.

The syntax of the ENCODE and DECODE subcommands has been enhanced as follows.

ENCODE Subcommand

The ENCODE subcommand encrypts a member. The data encryption algorithm follows the Advanced Encryption Standard (AES) set forth in Federal Information Processing Standard (FIPS) 197.

Example

```
ENCODE 'USER1.C227693.CLEAR.VB' 'USER1.C227693.CODED.U' 'USER1.C227693.KEY' SHR
```

Syntax

ENCODE *input-dsn output-dsn key-dsn* [SHR|<u>OLD</u>]

Aliases

```
ENC, ENCO, ENCOD, ENCODE
```

Defaults

```
OLD
```

Required

input-dsn, output-dsn, key-dsn

Operands

input-dsn	Fully qualified data set name enclosed in single quotes.
output-dsn	Fully qualified data set name enclosed in single quotes.
key-dsn	Fully qualified data set name enclosed in single quotes.
`OLD`	This allocation is not to be shared with any other allocation for this data set.
`SHR`	This allocation is to be shared with other allocations for this data set.

Remarks

FDM supports both decryption and encryption of data sets with the National Institute of Standards and Technology (NIST) publication FIPS 197. Details are in publication:

http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

When you compare encrypted files without decrypting, that is at the binary level, a single byte difference within a record will cause the entire record to be different. This is a feature of the encryption algorithm.

Key Specification

The FDM users are custodians of their own keys. The data set that contains the keys can be RACF protected "for your eyes only". FDM will not show the keys in printed output or on the joblog or system console. FDM will erase the keys in memory so that they cannot be observed in program dumps and abends.

The key file contains exactly 2 records:

- Record 1 is the key that is used to process the input file.
- Record 2 is not used.

The key file is required for both decrypt and encrypt operations and is the same format in both cases. The actual key is expected to be a hexadecimal string. If you prefer a character form, enclose the key specification in quotes. The format of the key in record 1 is:

Item	Column Position	Description
Key Type	1-3	Values can be: 000, 128, 192, 256. 000 means no key.

Item	Column Position	Description
Space	4	Value must be a single space.
Key	5-68	Hexadecimal key specified in hex characters 0-9, lowercase a-f, or uppercase A-F. Maximum key length is: • 128 key type means key contains a maximum of 32 hexadecimal digits (16 hexadecimal bytes). • 192 key type means key contains a maximum of 48 hexadecimal digits (24 hexadecimal bytes). • 256 key type means key contains a maximum of 64 hexadecimal digits (32 hexadecimal bytes). Enclose character-format key in single quotes. • 128 key type means the key contains up to 16 alphanumeric characters. • 192 key type means the key contains from 17 to 24 alphanumeric characters. • 256 key type means the key contains from 25 to 32 alphanumeric characters.

Example Key Specifications

- For a character key:

  ```
  128 ’CIPHER KEY’
  ```

- For a hexadecimal key:

  ```
  192 aabbccddee00999ff0ff
  ```

Encode Operation

FDM will encrypt the input file. If the key type is 000, the results that are written to the output data set remain unaltered.

Verification

Take the following steps to verify that the ENCODE process works and that you can DECODE back to the original text:

- Create the output file; model it after the input data set.
- Create the key data set. It will contain 2 records as outlined above. Only the first record is used, but both must be in the data set or member. This file should be under RACF (or other security) so that a limited number of users have READ access.
- Enter the ENCODE command.
- Fill in the input, output, and key data set names. Use ENTER to start the process.
- Create a new, empty file for the DECODE output.

- Enter the DECODE command.
- Fill in the input (ENCODE output), output (new empty file), and key data set names. Use the same key data set that you used for the ENCODE process. Use ENTER to start the process.
- View all three data sets.
    - Verify that the ENCODE output data set is not readable.
    - Verify that the ENCODE input and the DECODE output data sets match.

DECODE Subcommand

The DECODE subcommand decrypts a member. The data encryption algorithm follows the Advanced Encryption Standard (AES) set forth in Federal Information Processing Standard (FIPS) 197.

Example

```
DECODE 'USER1.C227693.CODED.U' ’USER1.CLEAR.COPY' ’USER1.C227693.KEY' SHR
```

Syntax

`DECODE` *`input-dsn output-dsn key-dsn`* `[SHR|OLD]`

Aliases

`DEC, DECO, DECOD, DECODE`

Defaults

`OLD`

Required

`input-dsn, output-dsn, key-dsn`

Operands

input-dsn	Fully qualified data set name enclosed in single quotes.
output-dsn	Fully qualified data set name enclosed in single quotes.
key-dsn	Fully qualified data set name enclosed in single quotes.
`OLD`	This allocation is not to be shared with any other allocation for this data set.
`SHR`	This allocation is to be shared with other allocations for this data set.

Remarks

FDM supports both decryption and encryption of data sets with the National Institute of Standards and Technology (NIST) publication FIPS 197. Details are in publication:

`http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf`

Key Specification

The FDM users are custodians of their own keys. The data set that contains the keys can be RACF protected "for your eyes only". FDM will not show the keys in printed output or on the joblog or system console. FDM will erase the keys in memory so that they cannot be observed in program dumps and abends.

The key file contains exactly 2 records:

- Record 1 is the key that is used to process the input file.
- Record 2 is not used.

The key file is required for both decrypt and encrypt operations and is the same format in both cases. The actual key is expected to be a hexadecimal string. If you prefer a character form, enclose the key specification in quotes. The format of the key in record 1 is:

Item	Column Position	Description
Key Type	1-3	Values can be: 000, 128, 192, 256. 000 means no key.
Space	4	Value must be a single space.
Key	5-68	Hexadecimal key specified in hex characters 0-9, lowercase a-f, or uppercase A-F. Maximum key length is: • 128 key type means key contains a maximum of 32 hexadecimal digits (16 hexadecimal bytes). • 192 key type means key contains a maximum of 48 hexadecimal digits (24 hexadecimal bytes). • 256 key type means key contains a maximum of 64 hexadecimal digits (32 hexadecimal bytes). Enclose character-format key in single quotes. • 128 key type means the key contains up to 16 alphanumeric characters. • 192 key type means the key contains from 17 to 24 alphanumeric characters. • 256 key type means the key contains from 25 to 32 alphanumeric characters.

Example Key Specifications

- For a character key:

```
128 ’CIPHER KEY’
```

- For a hexadecimal key:

```
192 aabbccddee00999ff0ff
```

Decode Operation

FDM will decrypt the input file. If the key type is 000, the results that are written to the output data set remain unaltered.

Verification

Take the following steps to verify that the ENCODE process works and that you can DECODE back to the original text:

- Create the output file; model it after the input data set.
- Create the key data set. It will contain 2 records as outlined above. Only the first record is used, but both must be in the data set or member. This file should be under RACF (or other security) so that a limited number of users have READ access.
- Enter the ENCODE command.
- Fill in the input, output, and key data set names. Use ENTER to start the process.
- Create a new, empty file for the DECODE output.
- Enter the DECODE command.
- Fill in the input (ENCODE output), output (new empty file), and key data set names. Use the same key data set that you used for the ENCODE process. Use ENTER to start the process.
- View all three data sets.
    - Verify that the ENCODE output data set is not readable.
    - Verify that the ENCODE input and the DECODE output data sets match.

Pressing PF3 from any PDSSET* Panel Returns to PDSMENUS Panel

Inconsistencies in processing the PF3 key from a PDSSET* panel have been resolved. Pressing PF3 from any PDSSET* panel now returns you to the PDSMENUS panel.

PDS#DFLT Source and Load Modules Are Deleted in FDM 7.8

Member PDS#DFLT (which contained shipped defaults in earlier FDM releases) has been deleted from the FDM 7.8 source and load library distribution files. Refer to the section titled “Default Processing,” in Chapter 3, “Configuring PDS#OPT4 Options,” in the *StarTool FDM Installation Guide* for information about default option processing.

PDS#OPTR Source and Load Modules Are Deleted in FDM 7.8

Member PDS#OPTR has been deleted from the FDM 7.8 source and load library distribution files. The following combination of modules takes over the function that PDS#OPTR performed:

Member	Distribution Library	Function
PDS#OPT4	STRFDM.V7R8M0.COPY	Default control table that is distributed with FDM 7.8.
SAMPOPTR	STRFDM.V7R8M0.ASSEMBLE	Contains instructions for restricting a single set of FDM commands for all users. Instructions for changes in PDS#OPT4 can be found in STRFDM.V7R8M0.ASSEMBLE(SAMPOPTR).
SAMPOPT4	STRFDM.V7R8M0.ASSEMBLE	Contains instructions for setting up levels of access to commands based on individual user access. Requires a security package, such as RACF.
SMP#OPT4	STRFDM.V7R8M0.JCL	The JCL to assemble either SAMPOPTR or SAMPOPT4 resulting in a separate program that is loaded at execution time.
SMP#OPTJ	STRFDM.V7R8M0.JCL	Same as SMP#OPT4, but the result is link edited into StarTool FDM 7.8.

Refer to the *StarTool FDM Installation Guide* for more information about these modules and their functions.

Member SMPPTFRV in the JCL Library Has Changed

The JCL that appears in member SMPPTFRV in the STRFDM.V7R8M0.JCL distribution library is as follows:

```
//  jobcard                                    <== Your jobcard
//*-------------------------------------------------------------------*
//* BE SURE TO UPDATE ALL ITEMS BELOW:
//*   1) CHANGE ALL "HLQ~"  TO HI LEVEL QUALIFIER      (EX:SYS2.SERENA)
//*   2) CHANGE ALL "PDSE~~~"  TO CURRENT PRODUCT/VERSION (EX: PDSE780)
//*   3) CHANGE PTF~~~~~       TO THE PTF BEING APPLIED  (EX: P7800001)
//*   4) CHANGE "CDNODE"       TO THE 'HOST' HIGH LEVEL
//*                            QUALIFIER SPECIFIED DURING THE CD FILE
//*                            UPLOAD PROCESS
//*
//*-------------------------------------------------------------------*
//* RECEIVEs the PTF                                                  *
//*-------------------------------------------------------------------*
//RECEIVE  EXEC SMPPROC,
//         PARM='PROCESS=WAIT',
//         DYNAMNBR=120,REGION=6M
//SMPCSI   DD DISP=SHR,
//         DSN=HLQ~.PDSE~~~.CSI
//SMPPTFIN DD   DISP=SHR,
//         DSN=CDNODE.MCS
//SMPHOLD  DD   DUMMY
//SMPCNTL  DD   *
  SET BOUNDARY(GLOBAL) .
  RECEIVE S(PTF~~~~~) LIST SYSMODS .
/*
```

StarTool FDM Compiler Translator Table Can Now Be Customized

The compiler translator table (SZFTRNSL) that StarTool FDM uses in, for example, the HISTORY command can now be modified to add new compiler translator information. (For example, the translator information that IBM assigns to its Assembler H (ASMH) compilers is Translators 566896201 and 5734AS1).

The compiler translator table is coded in assembly language member SZFTRNSL. The SZFTRNSL source code member is located in library *somnode*.PDSE*vrm*.ASSEMBLE. SZFTRNSL calls the assembly language macro SZFTRNSM. The source member for this macro is located in library *somnode*.PDSE*vrm*.COPY. The macro is needed for assembly of SZFTRNSL if you customize the values in the translator table.

To customize the Startool FDM compiler translator table, perform the following steps:

1 Make an editable copy of SZFTRNSL by copying the source from *somnode*.PDSE*vrm*.COPY into *somnode*.PDSE*vrm*.ASSEMBLE(SZFTRNSL) in the location instructed in this member.

2 Edit the parameter values in the table as desired, following the instructions contained in the member.

3 Assemble and link the modified SZFTRNSL member with macro SZFTRNSM and place the resulting load module in the separate library where you keep your customized StarTool FDM load members. You can find the JCL to assemble and link the module in *somnode*.PDSE*vrm*.CNTL(SZFTRNST).

4 Add your custom load library to the JOBLIB or STEPLIB concatenation used to run StarTool FDM. At runtime, if FDM finds an SZFTRNSL load module in the run-time load library concatenation, it uses it. If not, FDM uses the translator table with default values that was pre-linked at the factory into the STARTOOL load module.

You Can Now Set a Default Value for the StarBat RDW Parameter

You can set a default value for the Record Descriptor Word (RDW) parameter by adding the RDW value (RDW=0,1,2, or 3) to the list of #INITIAL values that are specified in the PDS#OPT4 member of the *somnode*.PDSE*vrm*.COPY distribution library. The default value that is shipped with StarTool FDM is 0.

You can override the default with a StarBat SYSIN command at execution time.

Refer to the StarTool FDM Installation Guide for more information about configuring the PDS#OPT4 member of the *somnode*.PDSE*vrm*.COPY distribution library.

Changes to the Disassembler

The DISASM subcommand and DISASM: Disassemble Load Members panel have been changed in FDM 7.8 to accommodate new operation codes in z/OS 2.1.

DISASM Subcommand

The DISASM subcommand reconstructs assembler language instructions from a module. This procedure is often referred to as decoding or disassembling compiled or assembled instructions.

Example

```
DISASM MYMEMB OFFSET(1AF2)
```

Syntax

```
DISASM   memgroup [MODULE(name)]
                  [OFFSET(hexnum)]
                  [FLOAT|NOFLOAT]
                  [64BIT|NO64BIT]
                  [PRIV|SEMIPRIV|NOPRIV]
                  [REASM]
```

Aliases

```
DISA, DISAS
```

Defaults

memgroup, NOFLOAT, NOPRIV, NO64BIT

Required

None.

Operands

memgroup	Identifies the members to be disassembled. Default member names, member lists, member name ranges, and member name patterns are allowed.
MODULE(*name*)	Specifies a 1- to 8-byte partial external name that limits CSECT or ENTRY names to be displayed. If you enter both MODULE and OFFSET keywords, the offset applies to each module selected. The MODULE operand has several valid forms: MODULE(*) – uses the previous name entered on any MODULE keyword. MODULE(PAR) – formats only a CSECT or ENTRY named PAR. MODULE(PAR*) – formats only a CSECT or ENTRY who name begins with PAR.
OFFSET(*hexnum*)	Specifies a 1- to 6-digit hexadecimal module offset at which the display is to begin. If you enter both MODULE and OFFSET keywords, the offset applies to each module selected.
FLOAT	Include floating point instructions in the instruction set.
NOFLOAT	Do not include floating point instructions in the instruction set.
64BIT	Include instructions that deal with 64-bit registers and addresses.
NO64BIT	Include instructions that only deal with 32-bit registers and addresses.
PRIV	Include privileged and semiprivileged instructions in the instruction set.
SEMIPRIV	Include only semiprivileged instructions in the instruction set. Semi-privileged instructions are those generally not allowed in certain environments such as CICS and IMS. Examples are: PC and PR.

NOPRIV	Include neither privileged nor semiprivileged instructions in the instruction set.
REASM	Outputs the disassembled instructions in a format that can be submitted to recreate the decoded module. When you request this format, the module offset field and the CSECT offset field (the first two fields of output) are reversed in order and placed at the end of each decoded statement. PDS141I header messages are suppressed and JCL is created to assemble and link the decoded module and its aliases.

Remarks

The DISASM subcommand disassembles load members and formats its data in one of two ways:

- Non-REASM format:

```
000 0000 FBILDV00 CSECT
000 0000 FBILDV00 AMODE 31
000 0000 FBILDV00 RMODE 24
000 0000          B       32(,R15)              47F0 F020        *.00.*
004 0004          DC    X'1A',C'F'              1AC6             *.F*
```

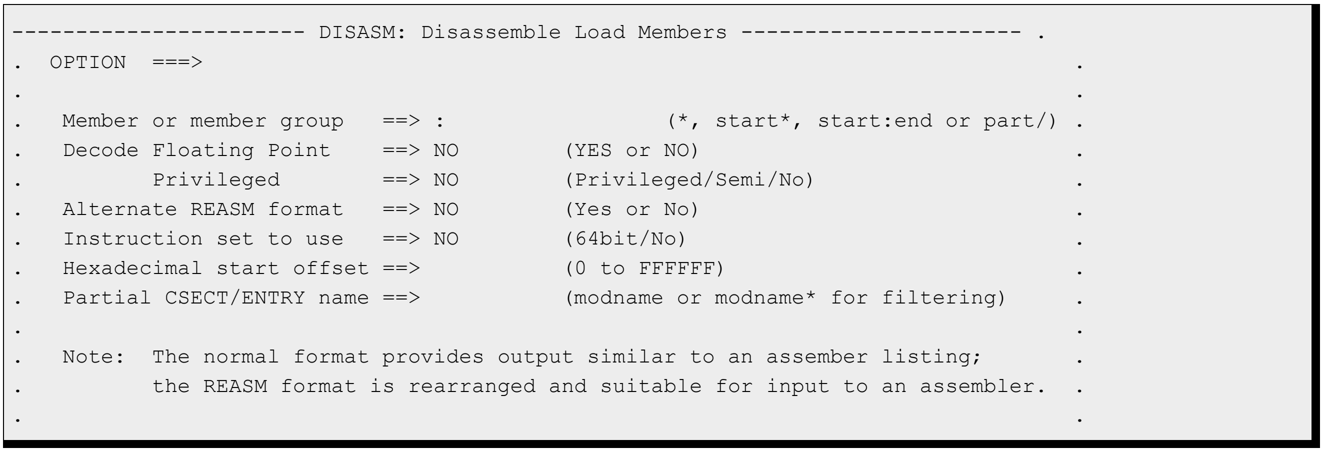
- REASM format (usable for reassembling):

```
//DIV       EXEC REASMLNK,OUTPUT='??REASM.OUTPUT',
//  PARM.LKED='NCAL,MAP,LIST,LET,RENT,OVLY,TEST'
//ASM.SYSIN   DD *
FBILDV00 CSECT                                                      000 0000
FBILDV00 AMODE 31                                                   000 0000
FBILDV00 RMODE 24                                                   000 0000
         B       32(,R15)              47F0 F020        *.00.*     000 0000
         DC    X'1A',C'F'              1AC6             *.F*       004 0004
         DC    X'0000000000000000'     0000000000000000  *........* 006 0006
```

The individual data columns are as follows:

module offset	A hexadecimal offset in the current module. With REASM format, this field is the last output field.
CSECT offset	A hexadecimal offset in the current CSECT. With REASM format, this field is just before the last output field.
name	An external name or generated internal name (the letter A followed by a 6-character offset in the module).
operation	1- to 7-byte reconstructed operation code.
operands	The reconstructed operand values.
comments	For SVC calls, the original MACRO (for example, GETMAIN or TPUT).
hex representation	1 to 3 columns of the original hexadecimal data.
character representation	1 to 8 bytes of the same data in character mode surrounded by asterisks.

The updated panel for the DISASM command looks like the following with the defaults filled in:



```
----------------------- DISASM: Disassemble Load Members ---------------------- .
.  OPTION  ===>                                                                    .
.                                                                                  .
.   Member or member group   ==> :                 (*, start*, start:end or part/) .
.   Decode Floating Point    ==> NO        (YES or NO)                             .
.          Privileged        ==> NO        (Privileged/Semi/No)                    .
.   Alternate REASM format   ==> NO        (Yes or No)                             .
.   Instruction set to use   ==> NO        (64bit/No)                              .
.   Hexadecimal start offset ==>           (0 to FFFFFF)                           .
.   Partial CSECT/ENTRY name ==>           (modname or modname* for filtering)     .
.                                                                                  .
.   Note:  The normal format provides output similar to an assember listing;       .
.          the REASM format is rearranged and suitable for input to an assembler.  .
.                                                                                  .
```

StarTool FDM Option 3.14--Search-For Utility Is Enhanced

You can now specify AND or OR search criteria in the second through last search string that you specify for this function on the StarBat Data Set Prompt (PDSSTB1) panel. In earlier FDM releases, multiple search string criteria was ANDed.

As part of this enhancement panel PDSSTB2 was deleted and the information from it was merged into panel PDSSTB1.

Disposition of TLOD Library in Later Releases

NOTE The *somnode*.PDSE*vrm*.TSRC and *somnode*.PDSE*vrm*.TLOD. libraries will be dropped from the StarTool FDM product in a future release. These libraries have not been updated in several years and, if needed, can be located by users. These programs and samples have never been officially supported. If you have questions or concerns about library contents, please contact Customer Support.

Fixes

Fixes in FDM 7.8.1

Click on the following link to access the list of defects that are fixed in FDM 7.8.1.

http://knowledgebase.serena.com/InfoCenter/index?page=content&cat=STRFDM_7.8.1_RESOLVED&channel=DEFECTS

Fixes in FDM 7.8.0.03

Click on the following link to access the list of defects that are fixed in FDM 7.8.0.03:

```
http://knowledgebase.serena.com/InfoCenter/
index?page=content&cat=STRFDM_7.8.0.03_RESOLVED&channel=DEFECTS
```

Fixes in FDM 7.8.0.02

Click on the following link to access the list of defects that are fixed in FDM 7.8.0.02:

```
http://knowledgebase.serena.com/InfoCenter/
index?page=content&cat=STRFDM_7.8.0.02_RESOLVED&channel=DEFECTS
```

Fixes in FDM 7.8.0.01

Click on the following link to access the list of defects that are fixed in FDM 7.8.0.01:

```
http://knowledgebase.serena.com/InfoCenter/
index?page=content&cat=STRFDM_7.8.0.01_RESOLVED&channel=DEFECTS
```

Fixes in FDM 7.8

Click on the following link to access the list of defects that are fixed in FDM 7.8:

```
http://knowledgebase.serena.com/InfoCenter/
index?page=content&cat=STRFDM_7.8_RESOLVED&channel=DEFECTS
```

Installation

The FDM 7.8.1 product distribution files contain the full FDM 7.8.1 product. You can download the FDM 7.8.1 distribution files from the Micro Focus SupportLine website at `https://supportline.microfocus.com/`.

You can install FDM 7.8.1 by using one of two methods:

- Perform an IEBCOPY (non-SMP/E) install.
- Perform an SMP/E install.

Refer to the *FDM Installation Guide* for the steps to follow for each install method.

Be sure to reapply any customizations you may have made to your production FDM libraries after you install FDM 7.8.1.

Software Compatibility

For a complete list of software requirements and product compatibility, refer to the Platform Matrix for StarTool FDM 7.8.1 at the following location on the Micro Focus SupportLine website:

```
https://supportline.microfocus.com/prodavail.aspx
```

Known Issues

The following are known issues for StarTool FDM 7.8.1:

- PBROWSE does not support multivolume files. Use PVIEW or PEDIT instead.
- VSAM linear data sets are limited by z/OS to a fixed CISIZE (Control Interval Size) of 4K. Attempts to use StarTool FDM with a linear data set of any other CISIZE results in an ABEND. This is a permanent restriction of the operating system, not a defect in StarTool FDM.
- The z/OS Binder issues an "internal binder error" when it encounters an object it doesn't recognize. When this happens, the binder returns an invalid number-of-ESD-entries value rather than an error code. This error can impact StarTool FDM. FDM customers should apply the IBM fix (APAR OW52128) to correct this error.
- IMS Version 8 macro libraries have been used to generate the PSBs and DBDs that are distributed with FDM 7.8.1. Newer versions of the IMS macro libraries may generate results not supported by FDM 7.8.1.

Documentation Updates

StarTool FDM 7.7.1 manuals are distributed with FDM 7.8.1. Refer to “New Features” on page 3 for a description new features in FDM 7.8.0 and later releases.