Program Directory for

IBM z/OS Debugger

V14.2.0

Program Number 5655-AC5, 5724-T07, 5737-I22, 5655-Q50

FMIDs HADRE20, JADRE2J, JADRE2K

for Use with
z/OS V02.02.00 or later

Document Date: September 2019

GI13-4540-03

Note

Before using this information and the product it supports, be sure to read the general information under 7.0, “Notices” on page 55.

A form for reader's comments appears at the back of this publication. When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Figures

1.0 Introduction

This program directory is intended for system programmers who are responsible for program installation and maintenance. It contains information about the material and procedures associated with the installation of IBM z/OS Debugger. This publication refers to IBM z/OS Debugger as z/OS Debugger.

z/OS Debugger is available as a component of multiple products. Depending on which product you acquired, different functions of z/OS Debugger will be enabled or disabled.

- IBM Developer for z/OS Enterprise Edition V14.2.0, program number 5655-AC5 (Shopz orderable)
- IBM Developer for z/OS V14.2.0, program number 5724-T07 (web download)
- IBM Z Open Development V2.0.0, program number 5737-I22 (web download)
- IBM Debug for z/OS V14.2.0, program number 5655-Q50 (Shopz orderable)

The Program Directory contains the following sections:

- 2.0, “Program Materials” on page 3 identifies the basic program materials and documentation for z/OS Debugger.
- 3.0, “Program Support” on page 8 describes the IBM support available for z/OS Debugger.
- 4.0, “Program and Service Level Information” on page 10 lists the APARs (program level) and PTFs (service level) that have been incorporated into z/OS Debugger.
- 5.0, “Installation Requirements and Considerations” on page 11 identifies the resources and considerations that are required for installing and using z/OS Debugger.
- 6.0, “Installation Instructions” on page 29 provides detailed installation instructions for z/OS Debugger. It also describes the procedures for activating the functions of z/OS Debugger, or refers to appropriate publications.

The rest of this section only applies when z/OS Debugger is ordered via Shopz. It does not apply when you downloaded it from the web.
Select this link to go to the next section: 1.1, “z/OS Debugger Description” on page 2

Before installing z/OS Debugger, read the *CBPDO Memo To Users* and the *CBPDO Memo To Users Extension* that are supplied with this program in softcopy format and this program directory; after which, keep the documents for your reference. Section 3.2, “Preventive Service Planning” on page 8 tells you how to find any updates to the information and procedures in this program directory.

z/OS Debugger is supplied in a Custom-Built Product Delivery Offering (CBPDO, 5751-CS3). The program directory that is provided in softcopy format on the CBPDO is identical to the hardcopy format if one was included with your order. All service and HOLDDATA for z/OS Debugger are included on the CBPDO.

Do not use this program directory if you install z/OS Debugger with a SystemPac or ServerPac. When you use one of those offerings, use the jobs and documentation supplied with the offering. The offering will point you to specific sections of this program directory as needed.

1.1 z/OS Debugger Description

IBM z/OS Debugger helps z/OS application developers debug programs running in a variety of environments, such as batch, TSO, CICS, IMS, DB2, DB2 Stored Procedures, and z/OS UNIX System Services.

1.2 z/OS Debugger FMIDs

z/OS Debugger consists of the following FMIDs:

HADRE20
JADRE2J
JADRE2K

The rest of this section only applies when z/OS Debugger is ordered via Shopz. It does not apply when you downloaded it from the web.
Select this link to go to the next section: 2.0, "Program Materials" on page 3

You will receive the following materials:

- If you order only the base feature, you will receive the following materials:
  - z/OS Debugger Base (FMID HADRE20)
- If you order both the base and Japanese features, you will receive the following materials:
  - z/OS Debugger Base (FMID HADRE20)
  - z/OS Debugger Japanese (FMID JADRE2J)
- If you order both the base and Korean features, you will receive the following materials:
  - z/OS Debugger Base (FMID HADRE20)
  - z/OS Debugger Korean (FMID JADRE2K)
- If you order all--the base, Japanese, and Korean features, you will receive the following materials:
  - z/OS Debugger Base (FMID HADRE20)
  - z/OS Debugger Japanese (FMID JADRE2J)
  - z/OS Debugger Korean (FMID JADRE2K)

2.0 Program Materials

An IBM program is identified by a program number.

z/OS Debugger is available as a component of multiple products. Depending on which product you acquired, different functions of z/OS Debugger will be enabled or disabled.

- IBM Developer for z/OS Enterprise Edition V14.2.0, program number 5655-AC5 (Shopz orderable)
- IBM Developer for z/OS V14.2.0, program number 5724-T07 (web download)
- IBM Z Open Development V2.0.0, program number 5737-I22 (web download)
- IBM Debug for z/OS V14.2.0, program number 5655-Q50 (Shopz orderable)

Basic Machine-Readable Materials are materials that are supplied under the base license and are required for the use of the product.

The program announcement material describes the features supported by z/OS Debugger. Ask your IBM representative for this information if you have not already received a copy.

2.1 Basic Machine-Readable Material

The distribution medium for this program is physical media or downloadable files. This program is in SMP/E RELFILE format and is installed by using SMP/E. See 6.0, “Installation Instructions” on page 29 for more information about how to install the program.

The rest of this section only applies when z/OS Debugger is ordered via Shopz. It does not apply when you downloaded it from the web.
Select this link to go to the next section: 2.2, “Program Publications” on page 5

You can find information about the physical media for the basic machine-readable materials for z/OS Debugger in the *CBPDO Memo To Users Extension*.

Figure 1 describes the program file content for z/OS Debugger. You can refer to the *CBPDO Memo To Users Extension* to see where the files reside on the image.

Notes:

1. The data set attributes in this table must be used in the JCL of jobs that read the data sets. However, because the data sets are in IEBCOPY unloaded format, their actual attributes might be different.
2. If any RELFILEs are identified as PDSEs, ensure that SMPTLIB data sets are allocated as PDSEs.

Figure 1. Program File Content -- Base

Name	ORG	RECFM	LRECL	BLKSIZE
SMPMCS	SEQ	FB	80	8800
IBM.HADRE20.F1	PDSE	FB	80	8800
IBM.HADRE20.F2	PDSE	FB	80	8800
IBM.HADRE20.F3	PDSE	U	0	6144
IBM.HADRE20.F4	PDSE	FB	80	8800
IBM.HADRE20.F5	PDSE	FB	80	8800
IBM.HADRE20.F6	PDSE	FB	80	8800
IBM.HADRE20.F7	PDSE	FB	80	8800
IBM.HADRE20.F8	PDSE	FB	80	8800
IBM.HADRE20.F9	PDSE	FB	80	8800
IBM.HADRE20.F10	PDSE	FB	80	8800
IBM.HADRE20.F11	PDSE	VB	6995	6999

Figure 2. Program File Content -- Japanese

Name	ORG	RECFM	LRECL	BLKSIZE
SMPMCS	SEQ	FB	80	8800
IBM.JADRE2J.F1	PDSE	FB	80	8800
IBM.JADRE2J.F2	PDSE	FB	80	8800
IBM.JADRE2J.F3	PDSE	U	0	6144
IBM.JADRE2J.F4	PDSE	FB	80	8800
IBM.JADRE2J.F5	PDSE	FB	80	8800
IBM.JADRE2J.F6	PDSE	FB	80	8800

Figure 3. Program File Content -- Korean

Name	ORG	RECFM	LRECL	BLK SIZE
SMPMCS	SEQ	FB	80	8800
IBM.JADRE2K.F1	PDSE	FB	80	8800
IBM.JADRE2K.F2	PDSE	FB	80	8800
IBM.JADRE2K.F3	PDSE	U	0	6144
IBM.JADRE2K.F4	PDSE	FB	80	8800
IBM.JADRE2K.F5	PDSE	FB	80	8800
IBM.JADRE2K.F6	PDSE	FB	80	8800

2.2 Program Publications

The following sections identify the basic publications for z/OS Debugger.

Figure 4 identifies the basic unlicensed program publications for z/OS Debugger. Those that are in softcopy format publications can be obtained from the IBM Publications Center website at https://www-05.ibm.com/e-business/linkweb/publications/servlet/pbi.wss.

Figure 4 (Page 1 of 2). Basic Material: Unlicensed Publications

Publication Title	Form Number	Media Format
IBM z/OS Debugger User's Guide	SC27-9580	See note[1]
IBM z/OS Debugger Reference Summary	SC27-9581	See note[1]
IBM z/OS Debugger Reference and Messages	SC27-9582	See note[1]
IBM z/OS Debugger Customization Guide	SC27-9583	See note[1]

Figure 4 (Page 2 of 2). Basic Material: Unlicensed Publications		
Publication Title	**Form Number**	**Media Format**
IBM z/OS Debugger API User's Guide and Reference	SC27-9584	See note[1]

Note:

1. These, and other, publications can be obtained from the online library of the product through which you aquired z/OS Debugger.
   - IBM Developer for z/OS Enterprise Edition V14.2.0, program number 5655-AC5, https://www-01.ibm.com/support/docview.wss?uid=swg27048563
   - IBM Developer for z/OS V14.2.0, program number 5724-T07, https://www-01.ibm.com/support/docview.wss?uid=swg27048563
   - IBM Z Open Development V2.0.0, program number 5737-I22, https://www-01.ibm.com/support/docview.wss?uid=ibm10738975
   - IBM Debug for z/OS V14.2.0, program number 5655-Q50, https://www-01.ibm.com/support/docview.wss?uid=swg27050482

Note: Requisite information can be obtained by creating a report for z/OS Debugger on https://www.ibm.com/software/reports/compatibility/clarity/index.html.

2.2.1 Optional Program Publications

No optional publications are provided for z/OS Debugger.

2.3 Program Source Materials

No program source materials or viewable program listings are provided for z/OS Debugger.

2.4 Publications Useful During Installation

You might want to use the publications listed in Figure 5 during the installation of z/OS Debugger.

Figure 5 (Page 1 of 2). Publications Useful During Installation		
Publication Title	**Form Number**	**Media Format**
IBM SMP/E for z/OS User's Guide	SA23-2277	See note[1]
IBM SMP/E for z/OS Commands	SA23-2275	See note[1]
IBM SMP/E for z/OS Reference	SA23-2276	See note[1]

Figure 5 (Page 2 of 2). Publications Useful During Installation

Publication Title	Form Number	Media Format
IBM SMP/E for z/OS Messages, Codes, and Diagnosis	GA32-0883	See note[1]

Note:

1. https://www-05.ibm.com/e-business/linkweb/publications/servlet/pbi.wss

3.0 Program Support

This section describes the IBM support available for z/OS Debugger.

3.1 Program Services

Contact your IBM representative for specific information about available program services.

3.2 Preventive Service Planning

Before you install z/OS Debugger, make sure that you have reviewed the current Preventive Service Planning (PSP) information. Review the PSP Bucket for General Information, Installation Documentation, and the Cross Product Dependencies sections. For the Recommended Service section, instead of reviewing the PSP Bucket, it is recommended you use the IBM.PRODUCTINSTALL-REQUIREDSERVICE fix category in SMP/E to ensure you have all the recommended service installed. Use the **FIXCAT(IBM.PRODUCTINSTALL-REQUIREDSERVICE)** operand on the **APPLY CHECK** command. See 6.1.16, “Perform SMP/E APPLY” on page 46 for a sample APPLY command

If you obtained z/OS Debugger as part of a CBPDO, HOLDDATA is included.

If the CBPDO for z/OS Debugger is older than two weeks by the time you install the product materials, you can obtain the latest PSP Bucket information by going to the following website:

http://www14.software.ibm.com/webapp/set2/psearch/search?domain=psp

You can also use S/390 SoftwareXcel or contact the IBM Support Center to obtain the latest PSP Bucket information.

For program support, access the Software Support Website at http://www.ibm.com/support/.

PSP Buckets are identified by UPGRADEs, which specify product levels; and SUBSETs, which specify the FMIDs for a product level. The UPGRADE and SUBSET values for z/OS Debugger are included in Figure 6.

Figure 6. PSP Upgrade and Subset ID

UPGRADE	SUBSET	Description
DEBUGE20	HADRE20	z/OS Debugger Base
DEBUGE20	JADRE2J	z/OS Debugger JPN
DEBUGE20	JADRE2K	z/OS Debugger KOR

3.3 Statement of Support Procedures

Report any problems which you feel might be an error in the product materials to your IBM Support Center. You may be asked to gather and submit additional diagnostics to assist the IBM Support Center in their analysis.

Figure 7 on page 9 identifies the component IDs (COMPID) for z/OS Debugger.

Figure 7. Component IDs

FMID	COMPID	Component Name	RETAIN Release
HADRE20	5724T0713	z/OS Debugger Base	E20
JADRE2J	5724T0713	z/OS Debugger JPN	E2J
JADRE2K	5724T0713	z/OS Debugger KOR	E2K

4.0 Program and Service Level Information

This section identifies the program and relevant service levels of z/OS Debugger. The program level refers to the APAR fixes that have been incorporated into the program. The service level refers to the PTFs that have been incorporated into the program.

4.1 Program Level Information

All resolved APARs of previous releases of z/OS Debugger have been incorporated into z/OS Debugger.

4.2 Service Level Information

No PTFs against this release of z/OS Debugger have been incorporated into the product package.

Frequently check the z/OS Debugger PSP Bucket for HIPER and SPECIAL attention PTFs against all FMIDs that you must install. You can also receive the latest HOLDDATA, then add the **FIXCAT(IBM.PRODUCTINSTALL-REQUIREDSERVICE)** operand on your **APPLY CHECK** command. This will allow you to review the recommended and critical service that should be installed with your FMIDs.

5.0 Installation Requirements and Considerations

The following sections identify the system requirements for installing and activating z/OS Debugger. The following terminology is used:

- *Driving system*: the system on which SMP/E is executed to install the program.

  The program might have specific operating system or product level requirements for using processes, such as binder or assembly utilities during the installation.

- *Target system*: the system on which the program is configured and run.

  The program might have specific product level requirements, such as needing access to the library of another product for link-edits. These requirements, either mandatory or optional, might directly affect the element during the installation or in its basic or enhanced operation.

In many cases, you can use a system as both a driving system and a target system. However, you can make a separate IPL-able clone of the running system to use as a target system. The clone must include copies of all system libraries that SMP/E updates, copies of the SMP/E CSI data sets that describe the system libraries, and your PARMLIB and PROCLIB.

Use separate driving and target systems in the following situations:

- When you install a new level of a product that is already installed, the new level of the product will replace the old one. By installing the new level onto a separate target system, you can test the new level and keep the old one in production at the same time.
- When you install a product that shares libraries or load modules with other products, the installation can disrupt the other products. By installing the product onto a separate target system, you can assess these impacts without disrupting your production system.

5.1 Driving System Requirements

This section describes the environment of the driving system required to install z/OS Debugger.

5.1.1 Machine Requirements

The driving system can run in any hardware environment that supports the required software.

5.1.2 Programming Requirements

Figure 8. Driving System Software Requirements

Program Number	Product Name	Minimum VRM	Minimum Service Level will satisfy these APARs	Included in the shipped product?
5650-ZOS	z/OS	V02.02.00	N/A	No

Note: The minimum z/OS level is either what is listed in the table, or the currently minimum supported OS level, whichever is the most recent.

Note: SMP/E is a requirement for Installation and is an element of z/OS but can also be ordered as a separate product, 5655-G44, minimally V03.06.00.

Note: Installation might require migration to new z/OS releases to be service supported. See https://www-01.ibm.com/software/support/lifecycle/index_z.html.

5.2 Target System Requirements

This section describes the environment of the target system required to install and use z/OS Debugger.

z/OS Debugger installs in the z/OS (Z038) SREL.

5.2.1 Machine Requirements

The target system can run in any hardware environment that supports the required software.

5.2.2 Programming Requirements

5.2.2.1 Installation Requisites

Installation requisites identify products that are required and *must* be present on the system or products that are not required but *should* be present on the system for the successful installation of this product.

Mandatory installation requisites identify products that are required on the system for the successful installation of this product. These products are specified as PREs or REQs.

z/OS Debugger has no mandatory installation requisites.

Conditional installation requisites identify products that are *not* required for successful installation of this product but can resolve such things as certain warning messages at installation time. These products are specified as IF REQs.

z/OS Debugger has no conditional installation requisites.

5.2.2.2 Operational Requisites

Operational requisites are products that are required and *must* be present on the system or products that are not required but *should* be present on the system for this product to operate all or part of its functions.

Mandatory operational requisites identify products that are required for this product to operate its basic functions.

Conditional operational requisites identify products that are *not* required for this product to operate its basic functions but are required at run time for this product to operate specific functions. These products are specified as IF REQs.

Figure 9 lists the required service for z/OS operational requisites. Install the appropriate PTFs according to the level of your operating system.

Figure 9. Service for z/OS Operational Requisites

FMID	Product Name	PTF
z/OS V2.3		
HLE77B0	(Language Environment)	UI50167 UI53807 UI53820 UI56511 UI61244
z/OS V2.2		
HLE77A0	(Language Environment)	UI29282 UI30573 UI31702 UI33265 UI43053 UI45429 UI43458 UI49699 UI53511 UI53889 UI54726 UI56507 UI61245

Figure 10 lists the conditional operational requisites.

Figure 10 (Page 1 of 2). Target System Conditional Operational Requisites

Program Number	Product Name and Minimum VRM/Service Level	Function
5696-234	High Level Assembler for MVS & VM & VSE V1.6	Debug of Assembler programs
5650-ZOS	z/OS V2.2 through V2.4 C/C++ Element	Debug of C/C++ programs
5655-Q42	File Manager for z/OS V14.1	Tools for working with z/OS data sets, DB2, CICS and IMS data
5655-Q41	Fault Analyzer for z/OS V14.1	Application abend analysis
Any **one** of the following:		
5655-AC5	FMID HHOPxxx - Developer for z/OS Enterprise Edition V14.2	Remote debug on workstation
5724-T07	FMID HHOPxxx - Developer for z/OS V14.2	Remote debug on workstation
5655-AC5	FMID HHOPxxx - Developer for z Systems Enterprise Edition V14.0 through V14.1	Remote debug on workstation

Figure 10 (Page 2 of 2). Target System Conditional Operational Requisites

Program Number	Product Name and Minimum VRM/Service Level	Function
5724-T07	FMID HHOPxxx - Developer for z Systems V14.0 through V14.1	Remote debug on workstation
5724-T07	FMID HHOPxxx - Rational Developer for z Systems V9.5	Remote debug on workstation
5737-J31	FMID HHOPxxx - Z Open Unit Test V1.0.0 through V2.0.0	Remote debug on workstation
5737-I22	FMID HHOPxxx - Z Open Development V1.0.0 through V2.0.0	Remote debug on workstation
Any **one** of the following:		
5655-EX1	Explorer for z/OS V3.0.1 through V3.1.1	Remote debug on workstation
5655-EXP	Explorer for z/OS V3.0	Remote debug on workstation
Any **one** of the following:		
5722-DFJ	CICS TS for z/OS Value Unit Edition, V5.2 through V5.5	CICS support
5655-Y04	CICS TS for z/OS V5.2 through V5.5	CICS support
Any **one** of the following:		
5770-AF3	DB2 12 for z/OS Value Unit Edition	DB2 support
5659-DB2	DB2 12 for z/OS	DB2 support
5697-P43	DB2 11 for z/OS Value Unit Edition	DB2 support
5615-DB2	DB2 11 for z/OS	DB2 support
Any **one** of the following:		
5655-DS5	IMS Database Value Unit Edition, V15	IMS support
5635-A06	IMS V15	IMS support
5655-DSE	IMS Database Value Unit Edition, V14	IMS support
5635-A05	IMS V14	IMS support
Any **one** of the following:		
5655-EC6	Enterprise COBOL for z/OS V6.1 through V6.3	Debug of COBOL programs
5655-W32	Enterprise COBOL for z/OS V5	Debug of COBOL programs
5655-S71	Enterprise COBOL for z/OS V4.2	Debug of COBOL programs
Any **one** of the following:		
5655-PL5	Enterprise PL/I for z/OS V5.1 through V5.3	Debug of PL/I programs
5655-W67	Enterprise PL/I for z/OS V4.5	Debug of PL/I programs

Figure 11 on page 15 lists the required service for conditional operational requisites. Install the appropriate PTFs according to the level of your product.

Figure 11. Service for Conditional Operational Requisites

FMID	Product Name	PTF
sub-systems		
HCI6900	CICS TS for z/OS V5.2	UI22206 UI30410
compilers		
HMQ4160	High Level Assembler for MVS & VM & VSE V1.6	UK47103 UK59311
HADB620	Enterprise COBOL for z/OS V6.2	UI57342
HADB610	Enterprise COBOL for z/OS V6.1	UI43370 UI48286 UI57900
HADB520	Enterprise COBOL for z/OS V5.2	UI42823 UI47619 UI62754 UI63478 UI63640
HADB510	Enterprise COBOL for z/OS V5.1	UI62769 UI63486 UI63637
miscellaneous		
HVWR180	Application Development Facility for z/OS Common Components V1.8	UI52558 UI54704 UI56366 UI56588 UI57380 UI58500 UI60690 UI61780 UI63179 UI63549
H09F210	COBOL and CICS Command Level Conversion Aid for OS/390 and MVS and VM V2.1	UK18923 UK93163 UK95729 UI19930 UI25023 UI36219

Figure 12 on page 16 lists old releases of conditional operational requisite products that are no longer in service.

IBM will undertake to fix any problem with z/OS Debugger that you might encounter when running z/OS Debugger with these unsupported levels. However, IBM must be able to reproduce the problem using a supported level of the same product.

If the problem can be reproduced and fixed, the fix for z/OS Debugger will be developed and tested using the supported levels of the product. If a fix is made available, it is likely to work on the unsupported product level. However, IBM cannot guarantee that the fix will work in this case.

Figure 12 (Page 1 of 2). Out-of-support Conditional Operational Requisites

Program Number	**Product Name and VRM/Service Level**	**Function**
sub-systems		
5655-147	CICS TS for OS/390 V1.3	CICS support
5697-E93	CICS TS for z/OS V2.2 through V2.3	CICS support
5655-M15	CICS TS for z/OS V3	CICS support
5655-S97	CICS TS for z/OS V4	CICS support
5722-DFJ	CICS TS for z/OS Value Unit Edition, V5.1	CICS support
5655-Y04	CICS TS for z/OS V5.1	CICS support
5675-DB2	DB2 UDB for z/OS and OS/390 V7	DB2 support
5625-DB2	DB2 UDB for z/OS V8	DB2 support
5635-DB2	DB2 V9 for z/OS	DB2 support
5697-P31	DB2 10 for z/OS Value Unit Edition	DB2 support
5605-DB2	DB2 10 for z/OS	DB2 support
5655-J38	IMS V9	IMS support
5635-A01	IMS V10	IMS support
5635-A02	IMS V11	IMS support
5655-DSQ	IMS Database Value Unit Edition, V12	IMS support
5635-A03	IMS V12	IMS support
5655-DSM	IMS Database Value Unit Edition, V13	IMS support
5635-A04	IMS V13	IMS support
compilers		
5696-234	High Level Assembler for MVS & VM & VSE V1.5	Debug of Assembler programs via the disassembly view

Figure 12 (Page 2 of 2). Out-of-support Conditional Operational Requisites

Program Number	Product Name and VRM/Service Level	Function
5688-216	AD/Cycle C/370 V1.2	Debug of C/370 programs
5655-121	C/C++ for MVS/ESA V3	Debug of C/C++ programs
5647-A01	OS/390 V2.10 C/C++ Element	Debug of C/C++ programs
5694-A01	z/OS V1 C/C++ Element	Debug of C/C++ programs
5650-ZOS	z/OS V2.1 through V2.2 C/C++ Element	Debug of C/C++ programs
5740-CB1	OS/VS COBOL V1.2.4	Debug of COBOL programs (with limitations)
5668-958, 5688-023	VS COBOL II V1.3.1, V1.3.2, V1.4	Debug of COBOL programs (with limitations)
5688-197	IBM COBOL for MVS & VM V1	Debug of COBOL programs
5648-A25	COBOL for OS/390 & VM V2.2	Debug of COBOL programs
5655-G53	IBM Enterprise COBOL for z/OS V3	Debug of COBOL programs
5655-S71	IBM Enterprise COBOL for z/OS V4.1	Debug of COBOL programs
5668-909, 5668-910	OS PL/I V2.1, V2.2, V2.3	Debug of PL/I programs (with limitations)
5655-B22	IBM VisualAge PL/I for OS/390 V2	Debug of PL/I programs
5688-235	IBM PL/I for MVS and VM V1.1.1	Debug of PL/I programs
5655-H31	Enterprise PL/I for z/OS V3	Debug of PL/I programs
5655-W67	Enterprise PL/I for z/OS V4.1 through V4.4	Debug of PL/I programs

Figure 13 lists the required service for the old releases of conditional operational requisite products in Figure 12 on page 16. Install the appropriate PTFs according to the level of your product.

Figure 13 (Page 1 of 3). Service for out-of-support Conditional Operational Requisites

FMID	Product Name	PTF
sub-systems		
HCI6800	CICS TS for z/OS V5.1	UI13727 UI22205
HCI6600	CICS TS for z/OS V4.1	UK48726 UK62906
HCI6500	CICS TS for z/OS V3.2	UK27838 UK31567 UK33556 UK39009 UK40995 UK48737
HCI6400	CICS TS for z/OS V3.1	UK11717 UK18764 UK21469 UK25495 UK31566 UK33555 UK39008 UK40994
HCI6300	CICS TS for z/OS V2.3	UQ83866 UQ83946 UK22257 UK25447 UK31617
HCI6200	CICS TS for z/OS V2.2	UQ81557 UQ82628
HCI5300	CICS TS for OS/390 V1.3	UQ81716 UQ82557
HDB7710	DB2 UDB for z/OS and OS/390 V7	UQ57178
compilers		
HMQ4150	High Level Assembler for MVS & VM & VSE V1.5	UK33757 UK40904 UK42558 UK59303
H26L330	Enterprise COBOL for z/OS V3.3	UK02153
H26L320	Enterprise COBOL for z/OS and OS/390 V3.2	UK02152 UQ84365
H26L310	Enterprise COBOL for z/OS and OS/390 V3.1	UQ72888
H249200	COBOL for OS/390 & VM V2.2	UQ71482 UQ72385

Figure 13 (Page 2 of 3). Service for out-of-support Conditional Operational Requisites

FMID	Product Name	PTF
H249100	COBOL for OS/390 & VM V2.1	UQ46089 UQ71481 UQ72384
J249101	COBOL for OS/390 & VM V2.1 (Mixed-case English)	UQ46090
J249102	COBOL for OS/390 & VM V2.1 (JPN)	UQ46091
H270440	Enterprise PL/I for z/OS V4.4	UI22280
H270430	Enterprise PL/I for z/OS V4.3	UI22279
H270420	Enterprise PL/I for z/OS V4.2	UI22276
H270380	Enterprise PL/I for z/OS V3.8	UK44457
H270370	Enterprise PL/I for z/OS V3.7	UK31095 UK39550 UK44456
H270360	Enterprise PL/I for z/OS V3.6	UK20098 UK21916 UK22225 UK29178 UK30600 UK31028 UK39549
H270350	Enterprise PL/I for z/OS V3.5	UK20097 UK20187 UK22224 UK29177 UK30599
H270340	Enterprise PL/I for z/OS V3.4	UK00175 UK01048 UK01926 UK01955 UK30598 UQ96871
H270330	Enterprise PL/I for z/OS V3.3	UK01925 UQ83141 UQ83550 UQ84230 UQ85850 UQ91552 UQ96870

Figure 13 (Page 3 of 3). Service for out-of-support Conditional Operational Requisites

FMID	Product Name	PTF
H270320	Enterprise PL/I for z/OS and OS/390 V3.2	UK01924 UQ71704 UQ82334 UQ83140 UQ83549 UQ84229 UQ96869
H270310	Enterprise PL/I for z/OS and OS/390 V3.1	UQ71463 UQ83139 UQ83548 UQ84228

5.2.2.3 Toleration/Coexistence Requisites

Toleration/coexistence requisites identify products that must be present on sharing systems. These systems can be other systems in a multisystem environment (not necessarily sysplex), a shared DASD environment (such as test and production), or systems that reuse the same DASD environment at different time intervals.

z/OS Debugger has no toleration/coexistence requisites.

5.2.2.4 Incompatibility (Negative) Requisites

Negative requisites identify products that must *not* be installed on the same system as this product.

z/OS Debugger has no negative requisites.

5.2.3 DASD Storage Requirements

z/OS Debugger libraries can reside on all supported DASD types.

Figure 14 lists the total space that is required for each type of library.

Figure 14 (Page 1 of 2). Total DASD Space Required by z/OS Debugger -- Base

Library Type	Total Space Required in 3390 Trks	Description
Target	2111 Tracks	
Distribution	2806 Tracks	
File System(s)	1200 Tracks	

Figure 14 (Page 2 of 2). Total DASD Space Required by z/OS Debugger -- Base

Library Type	Total Space Required in 3390 Trks	Description
Web Download	6349 Tracks	This row only applies when z/OS Debugger is downloaded from the web. It does not apply when you ordered it via Shopz. These are temporary data sets, which can be removed after the SMP/E install.

Figure 15. Total DASD Space Required by z/OS Debugger -- Japanese

Library Type	Total Space Required in 3390 Trks	Description
Target	123 Tracks	
Distribution	123 Tracks	
Web Download	260 Tracks	This row only applies when z/OS Debugger is downloaded from the web. It does not apply when you ordered it via Shopz. These are temporary data sets, which can be removed after the SMP/E install.

Figure 16. Total DASD Space Required by z/OS Debugger -- Korean

Library Type	Total Space Required in 3390 Trks	Description
Target	123 Tracks	
Distribution	123 Tracks	
Web Download	245 Tracks	This row only applies when z/OS Debugger is downloaded from the web. It does not apply when you ordered it via Shopz. These are temporary data sets, which can be removed after the SMP/E install.

Notes:

1. For non-RECFM U data sets, IBM recommends using system-determined block sizes for efficient DASD utilization. For RECFM U data sets, IBM recommends using a block size of 32760, which is most efficient from the performance and DASD utilization perspective.
2. Abbreviations used for data set types are shown as follows.

   U Unique data set, allocated by this product and used by only this product. This table provides all the required information to determine the correct storage for this data set. You do not need to refer to other tables or program directories for the data set size.

S Shared data set, allocated by this product and used by this product and other products. To determine the correct storage needed for this data set, add the storage size given in this table to those given in other tables (perhaps in other program directories). If the data set already exists, it must have enough free space to accommodate the storage size given in this table.

E Existing shared data set, used by this product and other products. This data set is *not* allocated by this product. To determine the correct storage for this data set, add the storage size given in this table to those given in other tables (perhaps in other program directories). If the data set already exists, it must have enough free space to accommodate the storage size given in this table.

If you currently have a previous release of this product installed in these libraries, the installation of this release will delete the old release and reclaim the space that was used by the old release and any service that had been installed. You can determine whether these libraries have enough space by deleting the old release with a dummy function, compressing the libraries, and comparing the space requirements with the free space in the libraries.

For more information about the names and sizes of the required data sets, see 6.1.12, “Allocate SMP/E Target and Distribution Libraries” on page 44.

3. All target and distribution libraries listed have the following attributes:
   - The default name of the data set can be changed.
   - The default block size of the data set can be changed.
   - The data set can be merged with another data set that has equivalent characteristics.
   - The data set can be either a PDS or a PDSE, with some exceptions. If the value in the "ORG" column specifies "PDS", the data set must be a PDS. If the value in "DIR Blks" column specifies "N/A", the data set must be a PDSE.
4. All target libraries listed have the following attributes:
   - These data sets can be SMS-managed, but they are not required to be SMS-managed.
   - These data sets are not required to reside on the IPL volume.
   - The values in the "Member Type" column are not necessarily the actual SMP/E element types that are identified in the SMPMCS.
5. All target libraries that are listed and contain load modules have the following attributes:
   - These data sets can not be in the LPA, with some exceptions. If the value in the "Member Type" column specifies "LPA", it is advised to place the data set in the LPA.
   - These data sets can be in the LNKLST.
   - These data sets are not required to be APF-authorized, with some exceptions. If the value in the "Member Type" column specifies "APF", the data set must be APF-authorized.

The following figures describe the target and distribution libraries required to install z/OS Debugger. The storage requirements of z/OS Debugger must be added to the storage required by other programs that have data in the same library.

Note: Use the data in these tables to determine which libraries can be merged into common data sets. In addition, since some ALIAS names may not be unique, ensure that no naming conflicts will be introduced before merging libraries.

Figure 17. Storage Requirements for z/OS Debugger Target Libraries -- Base

Library DDNAME	Member Type	Target Volume	TYPE	ORG	RECFM	LRECL	No. of 3390 Trks	No. of DIR Blks
SEQALPA	LPA	ANY	U	PDS	U	0	3	2
SEQAAUTH	APF	ANY	U	PDSE	U	0	14	N/A
SEQAMOD	LMOD	ANY	U	PDSE	U	0	1600	N/A
SEQABMOD	APF	ANY	U	PDSE	U	0	80	N/A
SEQASAMP	Sample	ANY	U	PDSE	FB	80	91	25
SEQAEXEC	EXEC	ANY	U	PDSE	FB	80	144	15
SEQAMENU	Message	ANY	U	PDSE	FB	80	2	2
SEQAPENU	Panel	ANY	U	PDSE	FB	80	112	55
SEQASENU	Skeleton	ANY	U	PDSE	FB	80	9	5
SEQATLIB	Table	ANY	U	PDSE	FB	80	5	2
SEQAMENP	Message	ANY	U	PDSE	FB	80	2	2
SEQAPENP	Panel	ANY	U	PDSE	FB	80	112	55
SEQASENP	Skeleton	ANY	U	PDSE	FB	80	9	5

- SEQALPA must be a PDS.
- SEQAAUTH, SEQAMOD, and SEQABMOD must be a PDS/E.
- SEQAMENU, SEQAPENU, and SEQASENU are ISPF libraries in mixed-case English.
- SEQAMENP, SEQAPENP, and SEQASENP are ISPF libraries in uppercase English.
- See section "Choosing a method to start z/OS Debugger Utilities" in chapter "Customizing z/OS Debugger Utilities" in the *IBM z/OS Debugger Customization Guide* (SC27-9583) for additional considerations if you want to merge these ISPF libraries into the common system libraries.

Figure 18 (Page 1 of 2). Storage Requirements for z/OS Debugger Target Libraries -- Japanese

Library DDNAME	Member Type	Target Volume	TYPE	ORG	RECFM	LRECL	No. of 3390 Trks	No. of DIR Blks
SEQAMOD	LMOD	ANY	S	PDSE	U	0	10	N/A
SEQABMOD	APF	ANY	S	PDSE	U	0	4	N/A
SEQASAMP	Sample	ANY	S	PDSE	FB	80	4	3

Figure 18 (Page 2 of 2). Storage Requirements for z/OS Debugger Target Libraries -- Japanese

Library DDNAME	Member Type	Target Volume	TYPE	ORG	RECFM	LRECL	No. of 3390 Trks	No. of DIR Blks
SEQAEXEC	EXEC	ANY	S	PDSE	FB	80	6	3
SEQAMJPN	Message	ANY	U	PDSE	FB	80	2	2
SEQAPJPN	Panel	ANY	U	PDSE	FB	80	112	70
SEQASJPN	Skeleton	ANY	U	PDSE	FB	80	9	3

- SEQAMOD, and SEQABMOD must be a PDS/E.
- SEQAMJPN, SEQAPJPN, and SEQASJPN are ISPF libraries in Japanese.
- See section "Choosing a method to start z/OS Debugger Utilities" in chapter "Customizing z/OS Debugger Utilities" in the *IBM z/OS Debugger Customization Guide* (SC27-9583) for additional considerations if you want to merge these ISPF libraries into the common system libraries.

Figure 19. Storage Requirements for z/OS Debugger Target Libraries -- Korean

Library DDNAME	Member Type	Target Volume	TYPE	ORG	RECFM	LRECL	No. of 3390 Trks	No. of DIR Blks
SEQAMOD	LMOD	ANY	S	PDSE	U	0	10	N/A
SEQABMOD	APF	ANY	S	PDSE	U	0	4	N/A
SEQASAMP	Sample	ANY	S	PDSE	FB	80	4	3
SEQAEXEC	EXEC	ANY	S	PDSE	FB	80	6	3
SEQAMKOR	Message	ANY	U	PDSE	FB	80	2	2
SEQAPKOR	Panel	ANY	U	PDSE	FB	80	112	70
SEQASKOR	Skeleton	ANY	U	PDSE	FB	80	9	3

- SEQAMOD, and SEQABMOD must be a PDS/E.
- SEQAMKOR, SEQAPKOR, and SEQASKOR are ISPF libraries in Korean.
- See section "Choosing a method to start z/OS Debugger Utilities" in chapter "Customizing z/OS Debugger Utilities" in the *IBM z/OS Debugger Customization Guide* (SC27-9583) for additional considerations if you want to merge these ISPF libraries into the common system libraries.

Figure 20. z/OS Debugger File System Paths

DDNAME	TYPE	Path Name
SEQAZFS	X	/usr/lpp/IBM/debug/IBM

Figure 21. Storage Requirements for z/OS Debugger Distribution Libraries -- Base

Library DDNAME	TYPE	ORG	RECFM	LRECL	No. of 3390 Trks	No. of DIR Blks
AEQAMOD	U	PDSE	U	0	1600	N/A
AEQASAMP	U	PDSE	FB	80	91	25
AEQAEXEC	U	PDSE	FB	80	144	15
AEQAMENU	U	PDSE	FB	80	2	2
AEQAPENU	U	PDSE	FB	80	112	55
AEQASENU	U	PDSE	FB	80	9	5
AEQATLIB	U	PDSE	FB	80	5	2
AEQAMENP	U	PDSE	FB	80	2	2
AEQAPENP	U	PDSE	FB	80	112	55
AEQASENP	U	PDSE	FB	80	9	5
AEQAZFS	U	PDSE	VB	6995	720	2

Figure 22. Storage Requirements for z/OS Debugger Distribution Libraries -- Japanese

Library DDNAME	TYPE	ORG	RECFM	LRECL	No. of 3390 Trks	No. of DIR Blks
AEQAMOD	S	PDSE	U	0	14	N/A
AEQASAMP	S	PDSE	FB	80	4	3
AEQAEXEC	S	PDSE	FB	80	6	3
AEQAMJPN	U	PDSE	FB	80	2	2
AEQAPJPN	U	PDSE	FB	80	112	70
AEQASJPN	U	PDSE	FB	80	9	3

Figure 23. Storage Requirements for z/OS Debugger Distribution Libraries -- Korean

Library DDNAME	TYPE	ORG	RECFM	LRECL	No. of 3390 Trks	No. of DIR Blks
AEQAMOD	S	PDSE	U	0	14	N/A
AEQASAMP	S	PDSE	FB	80	4	3
AEQAEXEC	S	PDSE	FB	80	6	3
AEQAMKOR	U	PDSE	FB	80	2	2
AEQAPKOR	U	PDSE	FB	80	112	70
AEQASKOR	U	PDSE	FB	80	9	3

The rest of this section only applies when z/OS Debugger is downloaded from the web. It does not apply when you ordered it via Shopz.
Select this link to go to the next section: 5.3, "FMIDs Deleted" on page 28

The following figures list data sets that are not used by z/OS Debugger, but are required as input for SMP/E.

Figure 24 (Page 1 of 2). Storage Requirements for z/OS Debugger Web Download Data Sets -- Base

Data Set Name	TYPE	ORG	RECFM	LRECL	No. of 3390 Trks	No. of DIR Blks
hlq.IBM.HADRE20.F1	U	PDSE	FB	80	96	N/A
hlq.IBM.HADRE20.F2	U	PDSE	FB	80	119	N/A
hlq.IBM.HADRE20.F3	U	PDSE	U	0	1357	N/A
hlq.IBM.HADRE20.F4	U	PDSE	FB	80	2	N/A
hlq.IBM.HADRE20.F5	U	PDSE	FB	80	108	N/A
hlq.IBM.HADRE20.F6	U	PDSE	FB	80	7	N/A
hlq.IBM.HADRE20.F7	U	PDSE	FB	80	7	N/A
hlq.IBM.HADRE20.F8	U	PDSE	FB	80	2	N/A
hlq.IBM.HADRE20.F9	U	PDSE	FB	80	108	N/A
hlq.IBM.HADRE20.F10	U	PDSE	FB	80	7	N/A
hlq.IBM.HADRE20.F11	U	PDSE	VB	6995	549	N/A
hlq.IBM.HADRE20.SMPMCS	U	SEQ	FB	80	12	N/A

Figure 24 (Page 2 of 2). Storage Requirements for z/OS Debugger Web Download Data Sets -- Base

Data Set Name	TYPE	ORG	RECFM	LRECL	No. of 3390 Trks	No. of DIR Blks
z/OS UNIX file system	U	zFS	N/A	N/A	3975	N/A

Note: These are temporary data sets, which can be removed after the SMP/E install.

Figure 25. Storage Requirements for z/OS Debugger Web Download Data Sets -- Japanese

Data Set Name	TYPE	ORG	RECFM	LRECL	No. of 3390 Trks	No. of DIR Blks
hlq.IBM.JADRE2J.F1	U	PDSE	FB	80	5	N/A
hlq.IBM.JADRE2J.F2	U	PDSE	FB	80	5	N/A
hlq.IBM.JADRE2J.F3	U	PDSE	U	0	9	N/A
hlq.IBM.JADRE2J.F4	U	PDSE	FB	80	2	N/A
hlq.IBM.JADRE2J.F5	U	PDSE	FB	80	108	N/A
hlq.IBM.JADRE2J.F6	U	PDSE	FB	80	7	N/A
hlq.IBM.JADRE2J.SMPMCS	U	SEQ	FB	80	4	N/A
z/OS UNIX file system	U	zFS	N/A	N/A	120	N/A

Note: These are temporary data sets, which can be removed after the SMP/E install.

Figure 26 (Page 1 of 2). Storage Requirements for z/OS Debugger Web Download Data Sets -- Korean

Data Set Name	TYPE	ORG	RECFM	LRECL	No. of 3390 Trks	No. of DIR Blks
hlq.IBM.JADRE2K.F1	U	PDSE	FB	80	5	N/A
hlq.IBM.JADRE2K.F2	U	PDSE	FB	80	5	N/A
hlq.IBM.JADRE2K.F3	U	PDSE	U	0	9	N/A
hlq.IBM.JADRE2K.F4	U	PDSE	FB	80	2	N/A
hlq.IBM.JADRE2K.F5	U	PDSE	FB	80	108	N/A
hlq.IBM.JADRE2K.F6	U	PDSE	FB	80	7	N/A
hlq.IBM.JADRE2K.SMPMCS	U	SEQ	FB	80	4	N/A

Figure 26 (Page 2 of 2). Storage Requirements for z/OS Debugger Web Download Data Sets -- Korean

Data Set Name	TYPE	ORG	RECFM	LRECL	No. of 3390 Trks	No. of DIR Blks
z/OS UNIX file system	U	zFS	N/A	N/A	105	N/A

Note: These are temporary data sets, which can be removed after the SMP/E install.

5.3 FMIDs Deleted

Installing z/OS Debugger might result in the deletion of other FMIDs. To see which FMIDs will be deleted, examine the ++VER statement in the SMPMCS of the product.

If you do not want to delete these FMIDs at this time, install z/OS Debugger into separate SMP/E target and distribution zones.

Note: These FMIDs are not automatically deleted from the Global Zone. If you want to delete these FMIDs from the Global Zone, use the SMP/E REJECT NOFMID DELETEFMID command. See the SMP/E Commands book for details.

5.4 Special Considerations

Target libraries that contain load modules have some special considerations. After you have successfully installed the product, do the following steps:

- Place SEQALPA in the LPA list by adding it to an LPALSTxx member of parmlib that will be used for IPL.
- Place SEQABMOD in the link list by adding a LNKLST ADD statement for it to a PROGxx member of parmlib that will be used for IPL.
- APF-authorize SEQAAUTH, and SEQABMOD by adding APF ADD statements for them to a PROGxx member of parmlib that will be used for IPL. The other load module data sets can be APF-authorized, but it is not required.
- Ensure that the LPA, and APF-authorized data sets are access controlled. This to prevent unauthorized execution of the programs within, and to prevent programs from being added to the data sets.
- For ease of access by the users, you can place SEQAMOD in the link list by adding a LNKLST ADD statement for it to a PROGxx member of parmlib that will be used for IPL.

6.0 Installation Instructions

This chapter describes the installation method and the step-by-step procedures to install and to activate the functions of z/OS Debugger.

Please note the following points:

- If you want to install z/OS Debugger into its own SMP/E environment, consult the SMP/E manuals for instructions on creating and initializing the SMPCSI and the SMP/E control data sets.
- You can use the sample jobs that are provided to perform part or all of the installation tasks. The SMP/E jobs assume that all DDDEF entries that are required for SMP/E execution have been defined in appropriate zones.
- You can use the SMP/E dialogs instead of the sample jobs to accomplish the SMP/E installation steps.

6.1 Installing z/OS Debugger

6.1.1 SMP/E Considerations for Installing z/OS Debugger

Use the SMP/E RECEIVE, APPLY, and ACCEPT commands to install this release of z/OS Debugger.

6.1.2 SMP/E Options Subentry Values

The recommended values for certain SMP/E CSI subentries are shown in Figure 27. Using values lower than the recommended values can result in failures in the installation. DSSPACE is a subentry in the GLOBAL options entry. PEMAX is a subentry of the GENERAL entry in the GLOBAL options entry. See the SMP/E manuals for instructions on updating the global zone.

Figure 27. SMP/E Options Subentry Values

Subentry	Value	Comment
DSSPACE	(1200,1200,1400)	Space allocation
PEMAX	SMP/E Default	IBM recommends using the SMP/E default for PEMAX.

6.1.3 SMP/E CALLLIBS Processing

z/OS Debugger uses the CALLLIBS function provided in SMP/E to resolve external references during installation. When z/OS Debugger is installed, ensure that DDDEFs exist for the following libraries:

- SCEELKED
- CSSLIB
- SEZATCP

Note: CALLLIBS uses the previous DDDEFs only to resolve the link-edit for z/OS Debugger. These data sets are not updated during the installation of z/OS Debugger.

6.1.4 Overview of the installation steps

Overview of steps required to install IBM z/OS Debugger.

1. Allocate file system to hold web download package
   (only applicable for web download, not Shopz orders)
2. Upload the web download package to the host
   (only applicable for web download, not Shopz orders)
3. Extract and expand the compressed SMPMCS and RELFILEs
   (only applicable for web download, not Shopz orders)

   Select this link to skip the web download specific steps: 6.1.8, "Sample Jobs" on page 39
4. Create SMP/E environment (optional)
5. Perform SMP/E RECEIVE
6. Allocate SMP/E target and distribution libraries
7. Allocate and mount z/OS UNIX file system (optional)
8. Allocate z/OS UNIX paths
9. Create DDDEF entries
10. Perform SMP/E APPLY
11. Enable/Register z/OS Debugger
12. Remove old registrations
13. Run the Installation Verification Programs (IVPs)
14. Verify the z/OS Debugger Utilities Setup Utility Function
15. Perform SMP/E ACCEPT
16. Run REPORT CROSSZONE

6.1.5 Allocate file system to hold web download package

This section only applies when z/OS Debugger is downloaded from the web. It does not apply when it is ordered via Shopz.
Select this link to skip the web download specific steps: 6.1.8, "Sample Jobs" on page 39

The SMP/E input data sets to install IBM z/OS Debugger are provided as compressed files in archives HADRE20.pax.Z, JADRE2J.pax.Z, and JADRE2K.pax.Z, which must be uploaded to z/OS as a z/OS UNIX file. You can either create a new z/OS UNIX file system (zFS) or create a new directory in an existing file system to place the download package.

You can use the following sample JCL to create a new file system, and directory, for the download package. The sample assumes that you will upload only the base feature (FMID HADRE20). The size of the file system must be increased if you also include other features:

- base feature, provided in HADRE20.pax.Z: 3975 tracks
- Japanese feature, provided in JADRE2J.pax.Z: 120 tracks
- Korean feature, provided in JADRE2K.pax.Z: 105 tracks

```
//FILESYS  JOB <job information>
//*
//* - Provide valid job card information
//* - Change:
//*   @zfs_path@
//*   ----+----1----+----2----+----3----+----4----+----5
//*              - To the absolute z/OS UNIX path for the
//*                download package (starting with /)
//*              - Maximum length is 50 characters
//*              - Do not include a trailing /
//*   @zfs_dsn@
//*              - To your file system data set name
//*
//* Your userid MUST be defined as a SUPERUSER to successfully
//* run this job
//*
//CREATE   EXEC PGM=IDCAMS,REGION=0M,COND=(0,LT)
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  DEFINE CLUSTER ( -
         NAME(@zfs_dsn@) -
         TRK(3975 10) -
         LINEAR -
         SHAREOPTIONS(3) -
         )
//*
//         SET ZFSDSN='@zfs_dsn@'
//FORMAT   EXEC PGM=IOEAGFMT,REGION=0M,COND=(0,LT),
//            PARM='-aggregate &ZFSDSN -compat'
//*STEPLIB  DD DISP=SHR,DSN=IOE.SIOELMOD        before z/OS 1.13
//*STEPLIB  DD DISP=SHR,DSN=SYS1.SIEALNKE       from z/OS 1.13
//SYSPRINT DD SYSOUT=*
//*
//MOUNT    EXEC PGM=IKJEFT01,REGION=0M,COND=(0,LT)
//SYSEXEC  DD DISP=SHR,DSN=SYS1.SBPXEXEC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD *
  PROFILE MSGID WTPMSG
  oshell umask 0022; +
    mkdir -p @zfs_path@
  MOUNT +
    FILESYSTEM('@zfs_dsn@') +
    MOUNTPOINT('@zfs_path@') +
    MODE(RDWR) TYPE(ZFS) PARM('AGGRGROW')
//*
```

Expected Return Codes and Messages: You will receive a return code of 0 if this job runs correctly.

6.1.6 Upload the web download package to the host

This section only applies when z/OS Debugger is downloaded from the web. It does not apply when it is ordered via Shopz.
Select this link to skip the web download specific steps: 6.1.8, “Sample Jobs” on page 39

Upload the HADRE20.readme.txt file, and JADRE2J.readme.txt or JADRE2K.readme.txt if required, in text format and the HADRE20.pax.Z file, and JADRE2J.pax.Z or JADRE2K.pax.Z, in binary format from your workstation to the z/OS UNIX file system.

In the following sample dialog, we use FTP from a Microsoft Windows command line to do the transfer. Commands or other information entered by the user are in bold, and the following values are assumed:

Figure 28. User Entered Values

User enters:	**Values**
mvsaddr	TCP/IP address or hostname of the z/OS system
tsouid	Your TSO user ID
tsopw	Your TSO password
d:	Location of the downloaded files
@zfs_path@	z/OS UNIX path where to store the files. This matches the @zfs_path@ variable you specified in the previous step.

Issue these commands to upload the z/OS Debugger Base:

```
C:\>ftp mvsaddr
Connected to mvsaddr.
220-FTPD1 IBM FTP CS %version% at mvsaddr, %time% on %date%.
220 Connection will close if idle for more than 5 minutes.
User (mvsaddr:(none)): tsouid
331 Send password please.
Password: tsopw
230 tsouid is logged on.  Working directory is "tsouid.".
ftp> cd @zfs_path@
250 HFS directory @zfs_path@ is the current working directory
ftp> ascii
200 Representation type is Ascii NonPrint
ftp> put d:\HADRE20.readme.txt
200 Port request OK.
125 Storing data set @zfs_path@/HADRE20.readme.txt
250 Transfer completed successfully.
ftp: 20331 bytes sent in 0.01 sec. (1366.67 Kb/s)
ftp> binary
200 Representation type is Image
ftp> put d:\HADRE20.pax.Z
```

```
200 Port request OK.
125 Storing data set @zfs_path@/HADRE20.pax.Z
250 Transfer completed successfully.
ftp: 73866240 bytes sent in 1.26 sec. (1040.52 Kb/s)
ftp> quit
221 Quit command received. Goodbye.
```

Issue these commands to upload the z/OS Debugger Japanese feature:

```
C:\>ftp mvsaddr
Connected to mvsaddr.
220-FTPD1 IBM FTP CS %version% at mvsaddr, %time% on %date%.
220 Connection will close if idle for more than 5 minutes.
User (mvsaddr:(none)): tsouid
331 Send password please.
Password: tsopw
230 tsouid is logged on.  Working directory is "tsouid.".
ftp> cd @zfs_path@
250 HFS directory @zfs_path@ is the current working directory
ftp> ascii
200 Representation type is Ascii NonPrint
ftp> put d:\JADRE2J.readme.txt
200 Port request OK.
125 Storing data set @zfs_path@/JADRE2J.readme.txt
250 Transfer completed successfully.
ftp: 19521 bytes sent in 0.01 sec. (1323.80 Kb/s)
ftp> binary
200 Representation type is Image
ftp> put d:\JADRE2J.pax.Z
200 Port request OK.
125 Storing data set @zfs_path@/JADRE2J.pax.Z
250 Transfer completed successfully.
ftp: 870912 bytes sent in 1.26 sec. (1115.22 Kb/s)
ftp> quit
221 Quit command received. Goodbye.
```

Issue these commands to upload the z/OS Debugger Korean feature:

```
C:\>ftp mvsaddr
Connected to mvsaddr.
220-FTPD1 IBM FTP CS %version% at mvsaddr, %time% on %date%.
220 Connection will close if idle for more than 5 minutes.
User (mvsaddr:(none)): tsouid
331 Send password please.
Password: tsopw
230 tsouid is logged on.  Working directory is "tsouid.".
ftp> cd @zfs_path@
250 HFS directory @zfs_path@ is the current working directory
ftp> ascii
200 Representation type is Ascii NonPrint
```

```
ftp> put d:\JADRE2K.readme.txt
200 Port request OK.
125 Storing data set @zfs_path@/JADRE2K.readme.txt
250 Transfer completed successfully.
ftp: 19521 bytes sent in 0.01 sec. (1371.34 Kb/s)
ftp> binary
200 Representation type is Image
ftp> put d:\JADRE2K.pax.Z
200 Port request OK.
125 Storing data set @zfs_path@/JADRE2K.pax.Z
250 Transfer completed successfully.
ftp: 870912 bytes sent in 1.26 sec. (1201.43 Kb/s)
ftp> quit
221 Quit command received. Goodbye.
```

6.1.7 Extract and expand the compressed SMPMCS and RELFILEs

This section only applies when z/OS Debugger is downloaded from the web. It does not apply when it is ordered via Shopz.
Select this link to skip the web download specific steps: 6.1.8, “Sample Jobs” on page 39

The HADRE20.readme.txt, JADRE2J.readme.txt, and JADRE2K.readme.txt files uploaded in the previous step each hold a sample JCL to expand the compressed SMPMCS and RELFILEs from the uploaded HADRE20.pax.Z, JADRE2J.pax.Z, and JADRE2K.pax.Z files into data sets for use by the SMP/E RECEIVE job. The JCLs are repeated here for your convenience.

- @zfs_path@ matches the variable you specified in the previous step.
- If the 'oshell' command gets a RC=256 and message "pax: checksum error on tape (got ee2e, expected 0)", then the archive file was not uploaded to the host in binary format.
- GIMUNZIP allocates data sets to match the definitions of the original data sets. You may encounter errors if your SMS ACS routines alter the attributes used by GIMUNZIP. If this occurs, specify a non-SMS managed volume for the GIMUNZIP allocation of the data sets. For example:

```
<ARCHDEF archid="..."
        storclas="storage_class" volume="data_set_volume"
        newname="..."/>
```

Customize and submit this JCL to expand the z/OS Debugger Base:

```
//EXTRACT  JOB <job information>
//*
//*   - Provide valid job card information
//*   - Change:
//*       @PREFIX@
//*       ----+----1----+----2----+
```

```
//*                  - To your desired data set name prefix
//*                  - Maximum length is 25 characters
//*                  - This value is used for the names of the
//*                    data sets extracted from the download-package
//*       @zfs_path@
//*       ----+----1----+----2----+----3----+----4----+----5
//*                  - To the absolute z/OS UNIX path for the
//*                    download-package (starting with /)
//*                  - Maximum length is 50 characters
//*                  - Do not include a trailing /
//*
//UNPAX    EXEC PGM=IKJEFT01,REGION=0M,COND=(0,LT)
//SYSEXEC  DD DISP=SHR,DSN=SYS1.SBPXEXEC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD *
  oshell cd @zfs_path@/ ; +
    pax -rvf HADRE20.pax.Z
//*
//GIMUNZIP EXEC PGM=GIMUNZIP,REGION=0M,COND=(0,LT)
//*STEPLIB  DD DISP=SHR,DSN=SYS1.MIGLIB
//SYSUT3   DD UNIT=SYSALLDA,SPACE=(CYL,(50,10))
//SYSUT4   DD UNIT=SYSALLDA,SPACE=(CYL,(25,5))
//SMPOUT   DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SMPDIR   DD PATHDISP=KEEP,
// PATH='@zfs_path@/'
//SYSIN    DD *
```

```
<GIMUNZIP>
<ARCHDEF archid="SMPMCS"
         newname="@PREFIX@.IBM.HADRE20.SMPMCS"/>
<ARCHDEF archid="IBM.HADRE20.F1"
         newname="@PREFIX@.IBM.HADRE20.F1"/>
<ARCHDEF archid="IBM.HADRE20.F2"
         newname="@PREFIX@.IBM.HADRE20.F2"/>
<ARCHDEF archid="IBM.HADRE20.F3"
         newname="@PREFIX@.IBM.HADRE20.F3"/>
<ARCHDEF archid="IBM.HADRE20.F4"
         newname="@PREFIX@.IBM.HADRE20.F4"/>
<ARCHDEF archid="IBM.HADRE20.F5"
         newname="@PREFIX@.IBM.HADRE20.F5"/>
<ARCHDEF archid="IBM.HADRE20.F6"
         newname="@PREFIX@.IBM.HADRE20.F6"/>
<ARCHDEF archid="IBM.HADRE20.F7"
         newname="@PREFIX@.IBM.HADRE20.F7"/>
<ARCHDEF archid="IBM.HADRE20.F8"
         newname="@PREFIX@.IBM.HADRE20.F8"/>
<ARCHDEF archid="IBM.HADRE20.F9"
         newname="@PREFIX@.IBM.HADRE20.F9"/>
<ARCHDEF archid="IBM.HADRE20.F10"
         newname="@PREFIX@.IBM.HADRE20.F10"/>
<ARCHDEF archid="IBM.HADRE20.F11"
         newname="@PREFIX@.IBM.HADRE20.F11"/>
</GIMUNZIP>
//*
```

Customize and submit this JCL to expand the z/OS Debugger Japanese feature:

```
//EXTRACT  JOB <job information>
//*
//*   - Provide valid job card information
//*   - Change:
//*       @PREFIX@
//*       ----+----1----+----2----+
//*                   - To your desired data set name prefix
//*                   - Maximum length is 25 characters
//*                   - This value is used for the names of the
//*                     data sets extracted from the download-package
//*       @zfs_path@
//*       ----+----1----+----2----+----3----+----4----+----5
//*                   - To the absolute z/OS UNIX path for the
//*                     download package (starting with /)
//*                   - Maximum length is 50 characters
//*                   - Do not include a trailing /
//*
//UNPAX    EXEC PGM=IKJEFT01,REGION=0M,COND=(0,LT)
//SYSEXEC  DD DISP=SHR,DSN=SYS1.SBPXEXEC
```

```
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD *
  oshell cd @zfs_path@/ ; +
    pax -rvf JADRE2J.pax.Z
//*
//GIMUNZIP EXEC PGM=GIMUNZIP,REGION=0M,COND=(0,LT)
//*STEPLIB  DD DISP=SHR,DSN=SYS1.MIGLIB
//SYSUT3   DD UNIT=SYSALLDA,SPACE=(CYL,(50,10))
//SYSUT4   DD UNIT=SYSALLDA,SPACE=(CYL,(25,5))
//SMPOUT   DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SMPDIR   DD PATHDISP=KEEP,
// PATH='@zfs_path@/'
//SYSIN    DD *
<GIMUNZIP>
<ARCHDEF archid="SMPMCS"
         newname="@PREFIX@.IBM.JADRE2J.SMPMCS"/>
<ARCHDEF archid="IBM.JADRE2J.F1"
         newname="@PREFIX@.IBM.JADRE2J.F1"/>
<ARCHDEF archid="IBM.JADRE2J.F2"
         newname="@PREFIX@.IBM.JADRE2J.F2"/>
<ARCHDEF archid="IBM.JADRE2J.F3"
         newname="@PREFIX@.IBM.JADRE2J.F3"/>
<ARCHDEF archid="IBM.JADRE2J.F4"
         newname="@PREFIX@.IBM.JADRE2J.F4"/>
<ARCHDEF archid="IBM.JADRE2J.F5"
         newname="@PREFIX@.IBM.JADRE2J.F5"/>
<ARCHDEF archid="IBM.JADRE2J.F6"
         newname="@PREFIX@.IBM.JADRE2J.F6"/>
</GIMUNZIP>
//*
```

Customize and submit this JCL to expand the z/OS Debugger Korean feature:

```
//EXTRACT  JOB <job information>
//*
//*   - Provide valid job card information
//*   - Change:
//*       @PREFIX@
//*       ----+----1----+----2----+
//*                   - To your desired data set name prefix
//*                   - Maximum length is 25 characters
//*                   - This value is used for the names of the
//*                     data sets extracted from the download-package
//*       @zfs_path@
//*       ----+----1----+----2----+----3----+----4----+----5
//*                   - To the absolute z/OS UNIX path for the
//*                     download package (starting with /)
//*                   - Maximum length is 50 characters
```

```
//*                  - Do not include a trailing /
//*
//UNPAX    EXEC PGM=IKJEFT01,REGION=0M,COND=(0,LT)
//SYSEXEC  DD DISP=SHR,DSN=SYS1.SBPXEXEC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD *
  oshell cd @zfs_path@/ ; +
    pax -rvf JADRE2K.pax.Z
//*
//GIMUNZIP EXEC PGM=GIMUNZIP,REGION=0M,COND=(0,LT)
//*STEPLIB  DD DISP=SHR,DSN=SYS1.MIGLIB
//SYSUT3   DD UNIT=SYSALLDA,SPACE=(CYL,(50,10))
//SYSUT4   DD UNIT=SYSALLDA,SPACE=(CYL,(25,5))
//SMPOUT   DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SMPDIR   DD PATHDISP=KEEP,
// PATH='@zfs_path@/'
//SYSIN    DD *
<GIMUNZIP>
<ARCHDEF archid="SMPMCS"
         newname="@PREFIX@.IBM.JADRE2J.SMPMCS"/>
<ARCHDEF archid="IBM.JADRE2K.F1"
         newname="@PREFIX@.IBM.JADRE2K.F1"/>
<ARCHDEF archid="IBM.JADRE2K.F2"
         newname="@PREFIX@.IBM.JADRE2K.F2"/>
<ARCHDEF archid="IBM.JADRE2K.F3"
         newname="@PREFIX@.IBM.JADRE2K.F3"/>
<ARCHDEF archid="IBM.JADRE2K.F4"
         newname="@PREFIX@.IBM.JADRE2K.F4"/>
<ARCHDEF archid="IBM.JADRE2K.F5"
         newname="@PREFIX@.IBM.JADRE2K.F5"/>
<ARCHDEF archid="IBM.JADRE2K.F6"
         newname="@PREFIX@.IBM.JADRE2K.F6"/>
</GIMUNZIP>
//*
```

6.1.8 Sample Jobs

The following sample installation jobs are provided as part of the product to help you install z/OS Debugger:

Figure 29 (Page 1 of 2). Sample Installation Jobs -- BASE

Job Name	Job Type	Description	RELFILE
EQAWEDIT	MACRO	ISPF Editor macro to aid in updating the sample jobs (optional)	IBM.HADRE20.F1

Figure 29 (Page 2 of 2). Sample Installation Jobs -- BASE

Job Name	Job Type	Description	RELFILE
EQAWSMPE	SMP/E	Sample job to create an SMP/E environment (optional)	IBM.HADRE20.F1
EQAWRECV	RECEIVE	Sample SMP/E RECEIVE job	IBM.HADRE20.F1
EQAWALOC	ALLOCATE	Sample job to allocate target and distribution libraries	IBM.HADRE20.F1
EQAWZFS	ALLOMZFS	Sample job to allocate, create mountpoint, and mount zFS data sets	IBM.HADRE20.F1
EQAWMKD	MKDIR	Sample job to invoke the supplied EQAWMKDIR EXEC to allocate file system paths	IBM.HADRE20.F1
EQAWDDEF	DDDEF	Sample job to define SMP/E DDDEFs	IBM.HADRE20.F1
EQAWAPLY	APPLY	Sample SMP/E APPLY job	IBM.HADRE20.F1
EQAWACPT	ACCEPT	Sample SMP/E ACCEPT job	IBM.HADRE20.F1
EQAWRPXZ	SMP/E	Sample REPORT CROSSZONE job	IBM.HADRE20.F1

Note: When z/OS Debugger is downloaded from the web, the RELFILE data set name will be prefixed by your chosen high level qualifier, as documented in section 6.1.7, “Extract and expand the compressed SMPMCS and RELFILEs” on page 35.

Figure 30. Sample Installation Jobs -- Japanese

Job Name	Job Type	Description	RELFILE
EQAWEDIJ	MACRO	ISPF Editor macro to aid in updating the sample jobs (optional)	IBM.JADRE2J.F1
EQAWRECJ	RECEIVE	Sample SMP/E RECEIVE job	IBM.JADRE2J.F1
EQAWALOJ	ALLOCATE	Sample job to allocate target and distribution libraries	IBM.JADRE2J.F1
EQAWDEFJ	DDDEF	Sample job to define SMP/E DDDEFs	IBM.JADRE2J.F1
EQAWAPLJ	APPLY	Sample SMP/E APPLY job	IBM.JADRE2J.F1
EQAWACPJ	ACCEPT	Sample SMP/E ACCEPT job	IBM.JADRE2J.F1

Note: When z/OS Debugger is downloaded from the web, the RELFILE data set name will be prefixed by your chosen high level qualifier, as documented in section 6.1.7, “Extract and expand the compressed SMPMCS and RELFILEs” on page 35.

Figure 31 (Page 1 of 2). Sample Installation Jobs -- Korean

Job Name	Job Type	Description	RELFILE
EQAWEDIK	MACRO	ISPF Editor macro to aid in updating the sample jobs (optional)	IBM.JADRE2K.F1

Figure 31 (Page 2 of 2). Sample Installation Jobs -- Korean

Job Name	Job Type	Description	RELFILE
EQAWRECK	RECEIVE	Sample SMP/E RECEIVE job	IBM.JADRE2K.F1
EQAWALOK	ALLOCATE	Sample job to allocate target and distribution libraries	IBM.JADRE2K.F1
EQAWDEFK	DDDEF	Sample job to define SMP/E DDDEFs	IBM.JADRE2K.F1
EQAWAPLK	APPLY	Sample SMP/E APPLY job	IBM.JADRE2K.F1
EQAWACPK	ACCEPT	Sample SMP/E ACCEPT job	IBM.JADRE2K.F1

Note: When z/OS Debugger is downloaded from the web, the RELFILE data set name will be prefixed by your chosen high level qualifier, as documented in section 6.1.7, “Extract and expand the compressed SMPMCS and RELFILEs” on page 35.

The rest of this section only applies when z/OS Debugger is ordered via Shopz. It does not apply when you downloaded it from the web.
Select this link to go to the next section: 6.1.9, “Set up ISPF Editor Macro (optional)” on page 43

You can access the sample installation jobs by performing an SMP/E RECEIVE (refer to 6.1.11, “Perform SMP/E RECEIVE” on page 44) then copy the jobs from the RELFILES to a work data set for editing and submission. See Figure 29 on page 39 to find the appropriate relfile data set.

You can also copy the sample installation jobs from the product files by submitting the following job. Before you submit the job, add a job card and change the lowercase parameters to uppercase values to meet the requirements of your site.

To copy the sample jobs for z/OS Debugger Base, use this JCL:

```
//STEP1    EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*
//IN       DD DSN=IBM.HADRE20.F1,
//            DISP=SHR,
//*           VOL=SER=filevol,
//            UNIT=SYSALLDA
//OUT      DD DSNAME=jcl-library-name,
//            DISP=(NEW,CATLG,DELETE),
//            SPACE=(TRK,(15,5,5)),
//            DSNTYPE=LIBRARY,
//*           VOL=SER=dasdvol,
//            UNIT=SYSALLDA
//SYSUT3   DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))
//SYSIN    DD *
    COPY INDD=IN,OUTDD=OUT
    SELECT MEMBER=(EQAWEDIT,EQAWSMPE,EQAWRECV,EQAWALOC)
    SELECT MEMBER=(EQAWDDEF,EQAWAPLY,EQAWACPT,EQAWRPXZ)
    SELECT MEMBER=(EQAWZFS,EQAWMKD)
/*
```

To copy the sample jobs for z/OS Debugger Japanese feature, use this JCL:

```
//STEP1    EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*
//IN       DD DSN=IBM.JADRE2J.F1,
//            DISP=SHR,
//*           VOL=SER=filevol,
//            UNIT=SYSALLDA
//OUT      DD DSNAME=jcl-library-name,
//            DISP=(NEW,CATLG,DELETE),
//            SPACE=(TRK,(15,5,5)),
//            DSNTYPE=LIBRARY,
//*           VOL=SER=dasdvol,
//            UNIT=SYSALLDA
//SYSUT3   DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))
//SYSIN    DD *
    COPY INDD=IN,OUTDD=OUT
    SELECT MEMBER=(EQAWEDIJ,EQAWRECJ,EQAWALOJ)
    SELECT MEMBER=(EQAWDEFJ,EQAWAPLJ,EQAWACPJ)
/*
```

To copy the sample jobs for z/OS Debugger Korean feature, use this JCL:

```
//STEP1    EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*
//IN       DD DSN=IBM.JADRE2K.F1,
//            DISP=SHR,
//*           VOL=SER=filevol,
//            UNIT=SYSALLDA
//OUT      DD DSNAME=jcl-library-name,
//            DISP=(NEW,CATLG,DELETE),
//            SPACE=(TRK,(15,5,5)),
//            DSNTYPE=LIBRARY,
//*           VOL=SER=dasdvol,
//            UNIT=SYSALLDA
//SYSUT3   DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))
//SYSIN    DD *
    COPY INDD=IN,OUTDD=OUT
    SELECT MEMBER=(EQAWEDIK,EQAWRECK,EQAWALOK)
    SELECT MEMBER=(EQAWDEFK,EQAWAPLK,EQAWACPK)
/*
```

See the following information to update the statements in the previous sample:

IN:

filevol is the volume serial of the DASD device where the downloaded files reside.

OUT:

jcl-library-name is the name of the output data set where the sample jobs are stored.

dasdvol is the volume serial of the DASD device where the output data set resides. Uncomment the statment if a volume serial must be provided.

6.1.9 Set up ISPF Editor Macro (optional)

An ISPF editor macro is provided to aid you in making changes to z/OS Debugger SMP/E installation jobs. The macro lets you substitute proper values for all of the required variables in those jobs instead of having you make the changes repeatedly by hand.

- EQAWEDIT, provided with the SMP/E jobs for the base product, updates the SMP/E jobs for the base product:
  EQAWSMPE, EQAWRECV, EQAWALOC, EQAWZFS, EQAWMKD,
  EQAWDDEF, EQAWAPLY, EQAWACPT
- EQAWEDIJ, provided with the SMP/E jobs for the Japanese feature, updates the SMP/E jobs for the base product and the Japanese feature:
  EQAWSMPE, EQAWRECV, EQAWALOC, EQAWZFS, EQAWMKD,
  EQAWDDEF, EQAWAPLY, EQAWACPT,
  EQAWRECJ, EQAWALOJ,
  EQAWDEFJ, EQAWAPLJ, EQAWACPJ
- EQAWEDIK, provided with the SMP/E jobs for the Korean feature, updates the SMP/E jobs for the base product and the Korean feature:
  EQAWSMPE, EQAWRECV, EQAWALOC, EQAWZFS, EQAWMKD,
  EQAWDDEF, EQAWAPLY, EQAWACPT,
  EQAWRECK, EQAWALOK,
  EQAWDEFK, EQAWAPLK, EQAWACPK

Edit the macro and provide the proper values. Consult the instructions in the macro for more information.

After making the changes, either copy the macro to any data set in your TSO logon procedure SYSEXEC concatenation, or issue the commands below to make the macro immediately accessible to your current ISPF session:

```
From ISPF option 6, issue:

ALLOCATE FI(SYSUEXEC) DA('dsn') SHR REU
  where dsn is the data set holding the macro
ALTLIB ACTIVATE USER(EXEC)
```

Consult the instructions in the macro on how to invoke it.

6.1.10 Create SMP/E environment (optional)

If you are using an existing CSI, do not run the sample job EQAWSMPE.

If you choose to create a new SMP/E environment for this install, a sample job is provided or you may choose to use your own JCL. If you choose to use the sample job provided, edit and submit EQAWSMPE. Consult the instructions in the sample job for more information.

Expected Return Codes and Messages: You will receive a return code of 0 if this job runs correctly.

6.1.11 Perform SMP/E RECEIVE

If you have obtained z/OS Debugger as part of a CBPDO, use the RCVPDO job in the CBPDO RIMLIB data set to receive the z/OS Debugger FMIDs, service, and HOLDDATA that are included on the CBPDO package. For more information, see the documentation that is included in the CBPDO.

You can also choose to edit and submit sample job EQAWRECV to perform the SMP/E RECEIVE for z/OS Debugger. Consult the instructions in the sample job for more information.

If your order contains the Japanese feature, edit and submit sample job EQAWRECJ to perform the SMP/E RECEIVE for that feature. Consult the instructions in the sample job for more information.

If your order contains the Korean feature, edit and submit sample job EQAWRECK to perform the SMP/E RECEIVE for that feature. Consult the instructions in the sample job for more information.

Expected Return Codes and Messages: You will receive a return code of 0 if this job runs correctly.

6.1.12 Allocate SMP/E Target and Distribution Libraries

Edit and submit sample job EQAWALOC to allocate the SMP/E target and distribution libraries for z/OS Debugger. Consult the instructions in the sample job for more information.

If your order contains the Japanese feature, edit and submit sample job EQAWALOJ to allocate the SMP/E target and distribution libraries for that feature. Consult the instructions in the sample job for more information.

If your order contains the Korean feature, edit and submit sample job EQAWALOK to allocate the SMP/E target and distribution libraries for that feature. Consult the instructions in the sample job for more information.

Expected Return Codes and Messages: You will receive a return code of 0 if this job runs correctly.

6.1.13 Allocate, create and mount ZFS Files (Optional)

This job allocates, creates a mountpoint, and mounts zFS data sets.

If you plan to install z/OS Debugger into a new z/OS UNIX file system, you can edit and submit the optional EQAWZFS job to perform the following tasks:

- Create the z/OS UNIX file system
- Create a mountpoint
- Mount the z/OS UNIX file system on the mountpoint

Consult the instructions in the sample job for more information.

The recommended z/OS UNIX file system type is *zFS*. The recommended mountpoint is */usr/lpp/IBM/debug*.

Before running the sample job to create the z/OS UNIX file system, you must ensure that OMVS is active on the driving system. zFS must be active on the driving system if you are installing z/OS Debugger into a file system that is zFS.

If you create a new file system for this product, consider updating the BPXPRMxx PARMLIB member to mount the new file system at IPL time. This action can be helpful if an IPL occurs before the installation is completed.

```
MOUNT FILESYSTEM('#dsn')
 MOUNTPOINT('/usr/lpp/IBM/debug')
 MODE(RDWR)        /* can be MODE(READ) */
 TYPE(ZFS) PARM('AGGRGROW') /* zFS, with extents */
```

See the following information to update the statements in the previous sample:

#dsn is the name of the data set holding the z/OS UNIX file system.
/usr/lpp/IBM/debug is the name of the mountpoint where the z/OS UNIX file system will be mounted.

Expected Return Codes and Messages: You will receive a return code of 0 if this job runs correctly.

6.1.14 Allocate File System Paths

The target system HFS or zFS data set must be mounted on the driving system when running the sample EQAWMKD job since the job will create paths in the HFS or zFS.

Before running the sample job to create the paths in the file system, you must ensure that OMVS is active on the driving system and that the target system's HFS or zFS file system is mounted to the driving system. zFS must be active on the driving system if you are installing z/OS Debugger into a file system that is zFS.

If you plan to install z/OS Debugger into a new HFS or zFS file system, you must create the mountpoint and mount the new file system to the driving system for z/OS Debugger.

The recommended mountpoint is */usr/lpp/IBM/debug*.

Edit and submit sample job EQAWMKD to allocate the HFS or zFS paths for z/OS Debugger. Consult the instructions in the sample job for more information.

If you create a new file system for this product, consider updating the BPXPRMxx PARMLIB member to mount the new file system at IPL time. This action can be helpful if an IPL occurs before the installation is completed.

Expected Return Codes and Messages: You will receive a return code of 0 if this job runs correctly.

6.1.15 Create DDDEF Entries

Edit and submit sample job EQAWDDEF to create DDDEF entries for the SMP/E target and distribution libraries for z/OS Debugger. Consult the instructions in the sample job for more information.

If your order contains the Japanese feature, edit and submit sample job EQAWDEFJ to create DDDEF entries for the SMP/E target and distribution libraries for that feature. Consult the instructions in the sample job for more information.

If your order contains the Korean feature, edit and submit sample job EQAWDEFK to create DDDEF entries for the SMP/E target and distribution libraries for that feature. Consult the instructions in the sample job for more information.

Expected Return Codes and Messages: You will receive a return code of 0 if this job runs correctly.

6.1.16 Perform SMP/E APPLY

1. Ensure that you have the latest HOLDDATA; then edit and submit sample job EQAWAPLY to perform an SMP/E APPLY CHECK for z/OS Debugger. Consult the instructions in the sample job for more information.

   If your order contains the Japanese feature, edit and submit sample job EQAWAPLJ to perform an SMP/E APPLY CHECK for that feature. Consult the instructions in the sample job for more information.

   If your order contains the Korean feature, edit and submit sample job EQAWAPLK to perform an SMP/E APPLY CHECK for that feature. Consult the instructions in the sample job for more information.

   The latest HOLDDATA is available through several different portals, including http://service.software.ibm.com/holdata/390holddata.html. The latest HOLDDATA may identify HIPER and FIXCAT APARs for the FMIDs you will be installing. An APPLY CHECK will help you determine if any HIPER or FIXCAT APARs are applicable to the FMIDs you are installing. If there are any applicable HIPER or FIXCAT APARs, the APPLY CHECK will also identify fixing PTFs that will resolve the APARs, if a fixing PTF is available.

   You should install the FMIDs regardless of the status of unresolved HIPER or FIXCAT APARs. However, do not deploy the software until the unresolved HIPER and FIXCAT APARs have been analyzed to determine their applicability. That is, before deploying the software either ensure fixing PTFs are applied to resolve all HIPER or FIXCAT APARs, or ensure the problems reported by all HIPER or FIXCAT APARs are not applicable to your environment.

   To receive the full benefit of the SMP/E Causer SYSMOD Summary Report, do *not* bypass the PRE, ID, REQ, and IFREQ on the APPLY CHECK. The SMP/E root cause analysis identifies the cause only of *errors* and not of *warnings* (SMP/E treats bypassed PRE, ID, REQ, and IFREQ conditions as warnings, instead of errors).

   Here are sample APPLY commands:

a. To ensure that all recommended and critical service is installed with the FMIDs, receive the latest HOLDDATA and use the APPLY CHECK command as follows

```
APPLY S(fmid,fmid,...) CHECK
FORFMID(fmid,fmid,...)
SOURCEID(RSU*)
FIXCAT(IBM.PRODUCTINSTALL-REQUIREDSERVICE)
GROUPEXTEND .
```

Some HIPER APARs might not have fixing PTFs available yet. You should analyze the symptom flags for the unresolved HIPER APARs to determine if the reported problem is applicable to your environment and if you should bypass the specific ERROR HOLDs in order to continue the installation of the FMIDs.

This method requires more initial research, but can provide resolution for all HIPERs that have fixing PTFs available and are not in a PE chain. Unresolved PEs or HIPERs might still exist and require the use of BYPASS.

b. To install the FMIDs without regard for unresolved HIPER APARs, you can add the BYPASS(HOLDCLASS(HIPER)) operand to the APPLY CHECK command. This will allow you to install FMIDs even though one or more unresolved HIPER APARs exist. After the FMIDs are installed, use the SMP/E REPORT ERRSYSMODS command to identify unresolved HIPER APARs and any fixing PTFs.

```
APPLY S(fmid,fmid,...) CHECK
FORFMID(fmid,fmid,...)
SOURCEID(RSU*)
FIXCAT(IBM.PRODUCTINSTALL-REQUIREDSERVICE)
GROUPEXTEND
BYPASS(HOLDCLASS(HIPER)) .
 ..any other parameters documented in the program directory
```

This method is quicker, but requires subsequent review of the Exception SYSMOD report produced by the REPORT ERRSYSMODS command to investigate any unresolved HIPERs. If you have received the latest HOLDDATA, you can also choose to use the REPORT MISSINGFIX command and specify Fix Category IBM.PRODUCTINSTALL-REQUIREDSERVICE to investigate missing recommended service.

If you bypass HOLDs during the installation of the FMIDs because fixing PTFs are not yet available, you can be notified when the fixing PTFs are available by using the APAR Status Tracking (AST) function of ServiceLink or the APAR Tracking function of ResourceLink.

2. After you take actions that are indicated by the APPLY CHECK, remove the CHECK operand and run the job again to perform the APPLY.

Note: The GROUPEXTEND operand indicates that SMP/E applies all requisite SYSMODs. The requisite SYSMODS might be applicable to other functions.

Expected Return Codes and Messages from APPLY CHECK: You will receive a return code of 0 if this job runs correctly.

Expected Return Codes and Messages from APPLY: You will receive a return code of 0 if this job runs correctly.

6.1.17 Enable/Register z/OS Debugger

Product registration information for z/OS Debugger can be found in the *IBM z/OS Debugger Customization Guide* (SC27-9583) publication.

6.1.18 Run the Installation Verification Programs (IVPs)

z/OS Debugger is available as a component of multiple products. Depending on which product you acquired, z/OS Debugger might only offer access via a GUI interface that is part of the workstation client. This implies that in this situation, you cannot run the 3270-based IVPs listed here, except EQAWIVPB. EQAWIVPB has instructions describing how to run it in remote mode (rather than full-screen using the Terminal Interface Manager).

- IBM Developer for z/OS Enterprise Edition V14.2.0, program number 5655-AC5, can run all IVPs
- IBM Developer for z/OS V14.2.0, program number 5724-T07, can run only EQAWIVPB
- IBM Z Open Development V2.0.0, program number 5737-I22, can run only EQAWIVPB
- IBM Debug for z/OS V14.2.0, program number 5655-Q50, can run all IVPs

Some IVPs require that certain PTFs be applied before you can successfully run them. To find the requisite PTFs, run the REPORT CROSSZONE job as shown in section 6.1.21, “Run REPORT CROSSZONE” on page 53.

Some IVPs also require certain customization to the product before you can successfully run them. See the specific information in each table below.

Depending on the compiler languages (COBOL, C, PL/I, or assembler) that your site uses, you can run one or all of the IVPs. Consult the instructions in each IVP for the expected return code and output. All of the IVPs are in the SEQASAMP library.

Figure 32 lists the IVPs for batch mode.

Figure 32 (Page 1 of 2). z/OS Debugger IVPs for Batch Mode

IVP Name	Description
EQAWIVPA[1]	Dynamic Debug facility and Language Environment Assembler
EQAWIVPC[1]	Dynamic Debug facility and Non-Language Environment Assembler
EQAWIVPF[1]	Dynamic Debug facility and Enterprise PL/I TEST(ALL,SYM,NOHOOK)
EQAWIVPG	z/OS C DEBUG(FORMAT(DWARF),HOOK(LINE,NOBLOCK,PATH),SYMBOL)
EQAWIVPI[1]	Dynamic Debug facility and Enterprise PL/I TEST(ALL,SYM,NOHOOK,SEPARATE)
EQAWIVPJ[1]	Dynamic Debug facility and LangX Language Environment COBOL IVP
EQAWIVPM	z/OS C DEBUG(FORMAT(DWARF),HOOK(LINE,NOBLOCK,PATH),SYMBOL), XPLINK

Figure 32 (Page 2 of 2). z/OS Debugger IVPs for Batch Mode

IVP Name	Description
EQAWIVPP[1]	Dynamic Debug facility and COBOL TEST(NONE,SYM,SEPARATE) COBOL for OS/390 and VM Enterprise COBOL for z/OS and OS/390 V3 Enterprise COBOL for z/OS V3 and V4
EQAWIVPS[1]	Dynamic Debug facility and Disassembly
EQAWIVPT[1]	Dynamic Debug facility and COBOL TEST Enterprise COBOL for z/OS V5 and V6
EQAWIVPV[1]	Dynamic Debug facility and OS/VS COBOL
EQAWIVPX[1]	Dynamic Debug facility and Non-Language Environment VS COBOL II
EQAWIVP1	COBOL TEST(ALL) COBOL for MVS and VM COBOL for OS/390 and VM Enterprise COBOL for z/OS and OS/390 V3 Enterprise COBOL for z/OS V3 and V4
EQAWIVP2	C TEST(ALL)
EQAWIVP3	PL/I for MVS and VM TEST(ALL)
EQAWIVP4[1]	Dynamic Debug facility and COBOL TEST(NONE,SYM) COBOL for OS/390 and VM Enterprise COBOL for z/OS and OS/390 V3 Enterprise COBOL for z/OS V3 and V4
EQAWIVP8	VisualAge or Enterprise PL/I TEST(ALL)

Note:

1. See chapter "Installing the z/OS Debugger SVCs" in the *IBM z/OS Debugger Customization Guide* (SC27-9583) before running these IVPs.

Figure 33 lists the IVPs for Remote debug in standard mode.

Figure 33 (Page 1 of 2). z/OS Debugger IVPs for Remote debug in standard mode

IVP Name	Description
EQAZIVPF[1]	Language Environment High Level Assembler Version 1 Release 6 ADATA
EQAZIVPI[1]	Enterprise PL/I for z/OS Version 4 or Version 5 31-bit TEST(NOHOOK,SEPARATE)
EQAZIVPP[1]	Enterprise PL/I for z/OS Version 5 64-bit TEST
EQAZIVPT[1]	Enterprise COBOL for z/OS Version 5 or Version 6 TEST
EQAZIVP5[1]	Enterprise COBOL for z/OS Version 3 Release 4 or Version 4 NOTEST

Figure 33 (Page 2 of 2). z/OS Debugger IVPs for Remote debug in standard mode	
IVP Name	**Description**
EQAZIVP6[1]	z/OS XL C DEBUG(FORMAT(DWARF),NOHOOK)

Note:

1. See chapter "Installing the z/OS Debugger SVCs" **and** chapter "Debug Manager (DBGMGR)" in the *IBM z/OS Debugger Customization Guide* (SC27-9583) before running these IVPs.

Figure 34 lists the IVPs for full-screen mode using the Terminal Interface Manager.

Figure 34. z/OS Debugger IVPs for Full-screen Mode using the Terminal Interface Manager	
IVP Name	**Description**
EQAWIVPB[2]	Dynamic Debug facility and Language Environment Assembler
EQAWIVPD[2]	Dynamic Debug facility and Non-Language Environment Assembler
EQAWIVPK[2]	Dynamic Debug facility and LangX Language Environment COBOL IVP
EQAWIVPU[2]	Dynamic Debug facility and COBOL TEST Enterprise COBOL for z/OS V5 and V6
EQAWIVPW[2]	Dynamic Debug facility and OS/VS COBOL
EQAWIVPY[2]	Dynamic Debug facility and Non-Language Environment VS COBOL II
EQAWIVP5[1]	COBOL TEST(ALL) COBOL for MVS and VM COBOL for OS/390 and VM Enterprise COBOL for z/OS and OS/390 V3 Enterprise COBOL for z/OS V3 and V4
EQAWIVP6[1]	C TEST(ALL)
EQAWIVP7[1]	PL/I for MVS and VM TEST(ALL)
EQAWIVP9[1]	VisualAge or Enterprise PL/I TEST(ALL)

Note:

1. See chapter "Enabling debugging in full-screen mode using the Terminal Interface Manager" in the *IBM z/OS Debugger Customization Guide* (SC27-9583) before running these IVPs.
2. See chapter "Enabling debugging in full-screen mode using the Terminal Interface Manager" **and** chapter "Installing the z/OS Debugger SVCs" before running these IVPs. Both of these chapters are in the *IBM z/OS Debugger Customization Guide* (SC27-9583).

Figure 35 lists the IVPs for running under CICS in full-screen mode.

Figure 35 (Page 1 of 2). z/OS Debugger IVPs for Running under CICS in Full-screen Mode	
IVP Name	**Description**
EQAWIVCC[2]	Dynamic Debug facility and Non-Language Environment Assembler

Figure 35 (Page 2 of 2). z/OS Debugger IVPs for Running under CICS in Full-screen Mode

IVP Name	Description
EQAWIVCG[1]	z/OS C DEBUG(FORMAT(DWARF),HOOK(LINE,NOBLOCK,PATH),SYMBOL)
EQAWIVCI[2]	Dynamic Debug facility and Enterprise PL/I TEST(ALL,SYM,NOHOOK,SEPARATE)
EQAWIVCJ[2]	Dynamic Debug facility and LangX Language Environment COBOL CICS IVP
EQAWIVCP[2]	Dynamic Debug facility and COBOL TEST(NONE,SYM,SEPARATE) COBOL for OS/390 and VM Enterprise COBOL for z/OS and OS/390 V3 Enterprise COBOL for z/OS V3 and V4
EQAWIVCT[2]	Dynamic Debug facility and COBOL TEST Enterprise COBOL for z/OS V5 and V6
EQAWIVC2[1]	C TEST(ALL)
EQAWIVC8[1]	VisualAge or Enterprise PL/I TEST(ALL)

Note:

1. See chapter "Adding support for debugging under CICS" in the *IBM z/OS Debugger Customization Guide* (SC27-9583) before running these IVPs.
2. See chapter "Adding support for debugging under CICS" **and** chapter "Installing the z/OS Debugger SVCs" before running these IVPs. Both of these chapters are in the *IBM z/OS Debugger Customization Guide* (SC27-9583).

Figure 36 lists the IVPs for running under CICS in standard mode.

Figure 36. z/OS Debugger IVPs for Running under CICS in standard mode

IVP Name	Description
EQAZIZCG	z/OS XL C DEBUG(FORMAT(DWARF),NOHOOK)
EQAZIZCI	Enterprise PL/I for z/OS Version 4 or Version 5 31-bit TEST(NOHOOK,SEPARATE)
EQAZIZCT	Enterprise COBOL for z/OS Version 5 or Version 6 TEST
EQAZIZC4	Enterprise COBOL for z/OS Version 3 Release 4 or Version 4 NOTEST

Figure 37 lists the IVPs for Load Module Analyzer.

Figure 37. z/OS Debugger IVPs for Load Module Analyzer

IVP Name	Description
EQAWLMA	Load Module Analyzer

Figure 38 lists the IVPs for z/OS Debugger Code Coverage.

Figure 38. z/OS Debugger IVPs for z/OS Debugger Code Coverage	
IVP Name	**Description**
EQACC1VZ	Enterprise COBOL for z/OS and OS/390 V3 Enterprise COBOL for z/OS V3 and V4
EQACC2VZ	Enterprise PL/I for z/OS V4.2 through V4.5 and V5
EQACC3VZ	z/OS XL C
EQACC4VZ	Enterprise COBOL for z/OS V5 and V6

6.1.19 Verify the z/OS Debugger Utilities Setup Utility Function

You can verify the installation and customization of the z/OS Debugger Utilities Setup Utility function by following one (or more) of the samples described in "Appendix D. Examples: Preparing programs and modifying setup files with z/OS Debugger Utilities" in the *IBM z/OS Debugger User's Guide* (SC27-9580).

See "Customizing z/OS Debugger Utilities" in the *IBM z/OS Debugger Customization Guide* (SC27-9583) first for customization information.

6.1.20 Perform SMP/E ACCEPT

Edit and submit sample job EQAWACPT to perform an SMP/E ACCEPT CHECK for z/OS Debugger. Consult the instructions in the sample job for more information.

If your order contains the Japanese feature, edit and submit sample job EQAWACPJ to perform an SMP/E ACCEPT CHECK for that feature. Consult the instructions in the sample job for more information.

If your order contains the Korean feature, edit and submit sample job EQAWACPK to perform an SMP/E ACCEPT CHECK for that feature. Consult the instructions in the sample job for more information.

To receive the full benefit of the SMP/E Causer SYSMOD Summary Report, do *not* bypass the PRE, ID, REQ, and IFREQ on the ACCEPT CHECK. The SMP/E root cause analysis identifies the cause of *errors* but not *warnings* (SMP/E treats bypassed PRE, ID, REQ, and IFREQ conditions as warnings rather than errors).

Before you use SMP/E to load new distribution libraries, it is recommended that you set the ACCJCLIN indicator in the distribution zone. In this way, you can save the entries that are produced from JCLIN in the distribution zone whenever a SYSMOD that contains inline JCLIN is accepted. For more information about the ACCJCLIN indicator, see the description of inline JCLIN in the SMP/E Commands book for details.

After you take actions that are indicated by the ACCEPT CHECK, remove the CHECK operand and run the job again to perform the ACCEPT.

Note: The GROUPEXTEND operand indicates that SMP/E accepts all requisite SYSMODs. The requisite SYSMODS might be applicable to other functions.

Expected Return Codes and Messages from ACCEPT CHECK: You will receive a return code of 0 if this job runs correctly.

If PTFs that contain replacement modules are accepted, SMP/E ACCEPT processing will link-edit or bind the modules into the distribution libraries. During this processing, the Linkage Editor or Binder might issue messages that indicate unresolved external references, which will result in a return code of 4 during the ACCEPT phase. You can ignore these messages, because the distribution libraries are not executable and the unresolved external references do not affect the executable system libraries.

Expected Return Codes and Messages from ACCEPT: You will receive a return code of 0 if this job runs correctly.

6.1.21 Run REPORT CROSSZONE

The SMP/E REPORT CROSSZONE command identifies requisites for products that are installed in separate zones. This command also creates APPLY and ACCEPT commands in the SMPPUNCH data set. You can use the APPLY and ACCEPT commands to install those cross-zone requisites that the SMP/E REPORT CROSSZONE command identifies.

After you install z/OS Debugger, it is recommended that you run REPORT CROSSZONE against the new or updated target and distribution zones. REPORT CROSSZONE requires a global zone with ZONEINDEX entries that describe all the target and distribution libraries to be reported on.

For more information about REPORT CROSSZONE, see the SMP/E manuals.

6.1.22 Cleaning Up Obsolete Data Sets, Paths, and DDDEFs

The web download data sets listed in Figure 24 on page 26 (section 5.2.3, “DASD Storage Requirements” on page 20) are temporary data sets. You can delete these data sets after you complete the SMP/E install.

The following data sets, which were allocated and used by previous releases of this product, are no longer used in this release. You can delete these obsolete data sets after you delete the previous release from your system.

- AEQABIN: obsolete as of Debug Tool V13.1.0.
- SEQABIN: obsolete as of Debug Tool V13.1.0.
- LPALIB is no longer required as of Debug Tool V4.1. However, if you install z/OS Debugger in a zone that currently contains an old release that has an LPALIB, ensure that you leave the old LPALIB allocated so that SMP/E can perform the proper cleanup. After you complete the installation of z/OS Debugger V14.2.0, do **not** delete this data setas it may contain other products' modules.
- SEQALPA, which was obsolete as of Debug Tool V1.3, is now required as of Debug Tool V4.1.
- AEQAMLIB, AEQAPLIB, AEQASLIB: obsolete as of Debug Tool V3.1.0.
- SEQAMLIB, SEQAPLIB, SEQASLIB: obsolete as of Debug Tool V3.1.0.

- SEQACLIS, SEQADUM, SEQAIENU, SEQALPA, SEQAOS2, SEQAPROC, SEQA2ENU: obsolete as of Debug Tool V1.3.0.
- AEQACLIS, AEQAIENU, AEQAMOD2, AEQAOS2, AEQASRC2, AEQA2ENU: obsolete as of Debug Tool V1.3.0.

The following DDDEF entries, which were created and used by previous releases of this product, are no longer used in this release. You can delete these obsolete DDDEF entries after you delete the previous release from your system.

- AEQABIN: obsolete as of Debug Tool V13.1.0.
- SEQABIN: obsolete as of Debug Tool V13.1.0.
- LPALIB is no longer required as of Debug Tool V4.1. However, if you install z/OS Debugger in a zone that currently contains an old release that has an LPALIB, ensure that you leave the old LPALIB allocated so that SMP/E can perform the proper cleanup. After you complete the installation of z/OS Debugger V14.2.0, do **not** delete this data setas it may contain other products' modules.
- SEQALPA, which was obsolete as of Debug Tool V1.3, is now required as of Debug Tool V4.1.
- AEQAMLIB, AEQAPLIB, AEQASLIB: obsolete as of Debug Tool V3.1.0.
- SEQAMLIB, SEQAPLIB, SEQASLIB: obsolete as of Debug Tool V3.1.0.
- SEQACLIS, SEQADUM, SEQAIENU, SEQALPA, SEQAOS2, SEQAPROC, SEQA2ENU: obsolete as of Debug Tool V1.3.0.
- AEQACLIS, AEQAIENU, AEQAMOD2, AEQAOS2, AEQASRC2, AEQA2ENU: obsolete as of Debug Tool V1.3.0.

6.2 Product Customization

The publication *IBM z/OS Debugger Customization Guide* (SC27-9583) contains the necessary information to customize and use z/OS Debugger.

The publication *IBM z/OS Debugger User's Guide* (SC27-9580) contains information on possible compilation and link edit changes you may need to make to your build processes in order to debug your programs with z/OS Debugger.

A workstation client is required for some of the products that include z/OS Debugger. The workstation client is optional otherwise. The client is part of the product your aquired. A trial version of the client can be downloaded from the Mainframe DEV download page, https://developer.ibm.com/mainframe/products/downloads/.

- IBM Developer for z/OS Enterprise Edition V14.2.0, program number 5655-AC5, client is optional
- IBM Developer for z/OS V14.2.0, program number 5724-T07, client is required
- IBM Z Open Development V2.0.0, program number 5737-I22, client is required
- IBM Debug for z/OS V14.2.0, program number 5655-Q50, client is optional

7.0 Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

APAR numbers are provided in this document to assist in locating PTFs that may be required. Ongoing problem reporting may result in additional APARs being created. Therefore, the APAR lists in this document may not be complete. To obtain current service recommendations and to identify current product service requirements, always contact the IBM Customer Support Center or use S/390 SoftwareXcel to obtain the current "PSP Bucket".

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

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, New York 10504-1785
USA

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

7.1 Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Reader's Comments

Program Directory for IBM z/OS Debugger, September 2019

We appreciate your input on this publication. Feel free to comment on the clarity, accuracy, and completeness of the information or give us any other feedback that you might have.

Send your comments by emailing us at ibmkc@us.ibm.com, and include the following information:

1. Your name and address
2. Your email address
3. Your telephone or fax number
4. The publication title and order number
5. The topic and page number related to your comment
6. The text of your comment

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you submit.

Thank you for your participation.

Printed in USA

GI13-4540-03