IBM z/VSE

System Upgrade and Service

Version 5

IBM z/VSE

System Upgrade and Service

Version 5

Contents

Figures

Tables

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. 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 of the intellectual property rights of IBM may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

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, NY 10504-1785, U.S.A.

Any pointers in this publication to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement. IBM accepts no responsibility for the content or use of non-IBM websites specifically mentioned in this publication or accessed through an IBM website that is mentioned in this publication.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

```
IBM Deutschland GmbH
Dept. M358
IBM-Allee 1
71139 Ehningen
Germany
```

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

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.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

IPv6/VSE is a registered trademark of Barnard Software, Inc.

Accessibility

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features in z/VSE enable users to:

- Use assistive technologies such as screen readers and screen magnifier software
- Operate specific or equivalent features using only the keyboard
- Customize display attributes such as color, contrast, and font size

Using Assistive Technologies

Assistive technology products, such as screen readers, function with the user interfaces found in z/VSE. Consult the assistive technology documentation for specific information when using such products to access z/VSE interfaces.

Documentation Format

The publications for this product are in Adobe Portable Document Format (PDF) and should be compliant with accessibility standards. If you experience difficulties when you use the PDF files and want to request a web-based format for a publication, you can either write an email to s390id@de.ibm.com, or use the Reader Comment Form in the back of this publication or direct your mail to the following address:

```
IBM Deutschland Research & Development GmbH
Department 3282
Schoenaicher Strasse 220
D-71032 Boeblingen
Federal Republic of Germany
```

In the request, be sure to include the publication number and title.

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.

About This Publication

This publication describes how to apply service to z/VSE.

Applying service means, for example, using the Fast Service Upgrade (FSU) process to install a new service refresh, or installing Program Temporary Fixes (PTFs) and Authorized Program Analysis Reports (APARs) to a z/VSE system.

Who Should Use This Publication

This publication is intended for people who:
- Apply service to the z/VSE system.
- Install refreshes via FSU for z/VSE 5.2.
- Upgrade a z/VSE 4.3 system or z/VSE 5.1 to z/VSE 5.2.

A knowledge of basic VSE functions and some hardware operations is required.

How to Use This Publication

Please follow the instructions given in this publication to apply service to your system. z/VSE dialogs help you apply service effectively and efficiently. This publication references dialogs available with the Interactive Interface.

Accessing Dialogs

To help you access a dialog, this publication uses *sequences* which show the selection panel name and selection number you must enter to reach the Interactive Interface dialog. If you want to access IBM® service, for example, start with the *z/VSE Function Selection* panel and select:

1 (Installation)

4 (IBM Service)

Administrator Fast Path	Synonym
14	Default: **service** Yours:

Using Fast Path

As shown above, the left box under a dialog sequence contains the *Fast Path* for a selection. This facility allows you to go directly to a dialog without working through the entire panel hierarchy. To use Fast Path, you enter all the numbers you would enter on the individual panels in the hierarchy on a single selection panel.

Using Synonyms

In addition to Fast Path, you can use a synonym for a certain dialog selection. The right box, as shown in the above example, contains the synonym for a selection.

Instead of entering the Fast Path 14, you could enter *service*. This invokes the dialog *IBM Service*. Instead of using the default synonyms, you also can use your own synonyms which you remember better. To create your own synonyms, use the *Maintain Synonym* dialog.

For more information on the Fast Path and synonym functions, refer to *z/VSE Administration*.

Where to Find More Information

Details about z/VSE 5.2, its programs and functions are provided in *z/VSE Release Guide* and *z/VSE Planning*.

z/VSE Home Page

z/VSE has a home page on the World Wide Web, which offers up-to-date information about VSE-related products and services, new z/VSE functions, and other items of interest to VSE users.

You can find the z/VSE home page at

http://www.ibm.com/systems/z/os/zvse/

You can also find VSE User Examples (in zipped format) at

http://www.ibm.com/systems/z/os/zvse/downloads/samples.html

Summary of Changes

This publication has been updated to reflect enhancements and changes that are implemented with z/VSE® Version 5 Release 2. It also includes terminology, maintenance, and editorial changes.

- The name VSE/ESA has changed to z/VSE. However, the names of many features and programs related to z/VSE remain unchanged (such as, for example, IBM COBOL for VSE/ESA, or TCP/IP for VSE/ESA).

- You can use the FSU (Fast Service Upgrade) process to upgrade z/VSE 4.3. or z/VSE 5.1. to z/VSE 5.2, but not from earlier releases. This is reflected in the corresponding FSU sections of this publication.

- z/VSE is distributed on 3590 or 3592 tapes, DVD or Internet (e-delivery).

- The Extended Base Tape now contains IPv6/VSE™.

- The maximum number of SCSI disks that can be defined with IPL DEF SCSI statement has been limited to 100. Refer to "Before Starting the FSU" on page 40 for details.

Chapter 1. General Service Concept for z/VSE

Applying service means that you apply a fix for an actual problem in your z/VSE system, or you install a preventive update on your z/VSE system to stay on a supported release and service level.

IBM offers you the following options to apply corrective and alternatively preventive service to your z/VSE system:

Program Temporary Fix (PTF)
> With PTFs you can fix actual problems that you have encountered in your system.

Preventive Service Planning Buckets (PSPs)
> PSP buckets for VSE contain all HIPER PTFs (and other recommended service) for a specific set of products or components on a given VSE refresh level.

Recommended Service Levels (RSLs)
> An RSL consists of a list of ALL APAR/PTF numbers, which are available at specific cutoff dates. Not only HIPER PTFs like the PSPs.

Service Refresh
> IBM periodically upgrades the VSE system libraries with the latest level of maintenance. This is known as a service refresh and contains all PTFs available at build time.

Release Upgrade
> If you are a z/VSE 4.3 or z/VSE 5.1 user and your system meets specific requirements (as described in *z/VSE Planning*), consider installing z/VSE 5.2 using the Fast Service Upgrade (FSU) process.

In the following sections these procedures are explained in more detail.

Note: For the latest service information see the *z/VSE Program Directory* and the z/VSE home page at: http://www.ibm.com/systems/z/os/zvse/

Program Temporary Fixes (PTFs)

If you detect a problem with your installed IBM software, contact your IBM Support Center. If you have a known problem, the Support Center can order a Program Temporary Fix (PTF) for you. The PTF is shipped to you on a distribution tape. You can also order a PTF via an online database directly from your IBM Support Center, or you can use ShopzSeries to order a PTF over the Internet.

When ordering a PTF, you get a fix for your problem without upgrading your total system to the most current z/VSE maintenance level. Usually, you will receive the prerequisite and corequisite PTFs shipped with the PTF you have ordered to ensure proper installability. To upgrade your z/VSE system to the latest level of maintenance, it is recommended to install the latest service refresh rather than successively install the PTFs.

"Applying Service through PTFs" on page 7 has more information on applying PTFs.

Preventive Service Planning Bucket (PSP)

Preventive Service Planning (PSP) buckets for VSE contain all HIPER PTFs (and other recommended service) for a specific set of products or components on a given VSE refresh level. A PSP bucket has a name (upgrade ID) and contains product information ordered into different 'subsets'. The upgrade ID can be used to order a complete PSP bucket from IBM support. To limit an order to specific products, subset IDs can be specified in addition. A subset contains different 'sections'. When ordering a PSP bucket, you get one or more PTF tapes containing all PTFs from the 'PTF Include List' of each ordered subset. PSPs are available on the Internet via the z/VSE Service and Support page where you can check individual fixes and/or order selected PTFs.

Recommended Service Level (RSL)

Another preventive service offering for VSE is the Recommended Service Level (RSL). This service offering fills the gap between VSE Refresh levels and the "High Impact or Pervasive APAR" (HIPER) service provided via "Preventive Service Planning" (PSP) buckets. An RSL consists of a list of ALL APAR/PTF numbers, which are available at specific cutoff dates. RSLs are updated more frequently than refreshes and contain ALL available service, not only the HIPER service. RSLs are published via special RLS PSP buckets and on the Internet. RSL PSPs are ordered on tape like HIPER PSPs and the Internet RSLs can be ordered electronically. This service offering helps customers keep their VSE system on a current and reliable service level. An RSL should be installed on top of the latest refresh, followed by the most current HIPER PSP bucket.

Service Refreshes

IBM periodically upgrades the z/VSE system libraries with the latest level of maintenance. This is known as a *service refresh* and contains all PTFs available at build time. Refreshes are primarily intended as service mechanisms for z/VSE systems.

When you initially order z/VSE, you automatically receive distribution tapes (cartridges) which reflect the latest service refresh. The content of the z/VSE distribution tapes is shown in Table 2 on page 4 and Table 3 on page 5. To ensure that your new system has the most recent level of complete system service, you should order and install a refresh as soon as possible after it becomes available. This refreshed system has been thoroughly tested. In general, a service refresh contains only corrections to the existing z/VSE system. In rare cases, it also provides additional functions. These functions do not influence your system if you do not use or need them.

As required, IBM will provide refreshes at identical service levels for the z/VSE base programs, extended base products and VSE optional programs.

As indicated previously, you can install a refresh to avoid possible problems. Thus a service refresh is an option for you, even if your system and its environment are stable. In addition, a refresh is advisable when you want to make major changes to your system (add hardware devices or IBM licensed programs, for example). The maintenance requirements for these changes make installing a service refresh the most efficient way to meet them.

How to install a service refresh is described in "Selection 3 – Fast Service Upgrade" on page 41.

Release Upgrade

If you are a z/VSE 4.3 or a z/VSE 5.1 user and your system meets specific requirements (as described in *z/VSE Planning*), consider installing z/VSE 5.2 using the Fast Service Upgrade (FSU) process.

Note: There are different requirements for file sizing depending on the source version. Such differences will be described in a note when they apply.

Unlike an initial installation, the FSU does not replace user data and hardware configuration tables. It only replaces system data. This results in a faster migration to z/VSE 5.2. See Chapter 5, "Performing a Fast Service Upgrade," on page 37 for more information.

Content of the z/VSE Distribution Media

z/VSE is distributed on the following media:
- three IBM 3590 or 3592 tapes
- 2 DVDs
- electronic delivery over the Internet (ShopzSeries).

Table 1. Media and Volumes

Distribution Medium	External Label Identifcation	Content	Filenames
IBM 3590, or 3592 tapes, Volume 1	z/VSE5.2.x-YY	z/VSE Base	n/a
IBM 3590, or 3592 tapes, Volume 2	z/VSE5.2.xXBASE	z/VSE Extended Base	n/a
IBM 3590, or 3592 tapes, Volume 3	z/VSE DB2® Help	DB2 Server for VSE Help	n/a
DVD, Volume 1 (in virtual tape format)	VSE_BASE_DVD_YYY	z/VSE Base, z/VSE Extended Base, DB2 Server for VSE Help, Readme	VSEvrmYY.aws, VSEvrmXB.aws, vrm = Version, Release, Modification Level,DB2SERVE.aws, READM5.2x.txt
DVD, Volume 2 (in virtual tape format)	VSEOPT5.2_YYY_DVD	Optional Products, Readme	VSEvrmOP.aws, READM5.2x.txt
Electronic delivery via ShopzSeries (in virtual tape format)	-	z/VSE Base, z/VSE Extended Base, DB2 Server for VSE Help, Readme	See ShopzSeries website.

The third distribution tape containing DB2 Server for VSE Help is also shipped, but is not used during FSU.

The contents of the distribution tapes and CD-ROMs are listed in Table 2 on page 4 and Table 3 on page 5. For more details and the latest information, refer to the *Program Directory for z/VSE 5.2*.

General Service Concept for z/VSE

Note that you receive additional tapes or CDs, if you have ordered z/VSE optional programs.

Note: The "x" in the tape ID or file name shows the latest modification level of the current release. IBM provides Multicultural Support for the Kanji version of z/VSE. "YY" is the language indicator for the language used. "YY" can be:

EN or ENU
   for English

KA   for Kanji (Japanese)

Table 2. Content of z/VSE Base Tape

File	z/VSE Base Tape
1	Header and SA Supervisor Facilities
2	VSE Standalone Utilities
3	MSHP History File - SYSRES Products
4	SYSRES Library Backup
5	Null File
6	End of Backup Record
7	DTSFILE Header
8	DTSFILE Non-NLS
9	DTSFILE Trailer
10	Header File
11	MSHP History File - PRD1.MACLIB Products
12	PRD1.MACLIB Sublibrary
13	Header File
14	MSHP History File - PRD2.SCEEBASE Products
15	PRD2.SCEEBASE Sublibrary
16	Header File
17	MSHP History File - MCS Products
18	MCS Library
19	Null File
20	End of Backup Record
21	DTSFILE - Header
22	DTSFILE - MCS
23	DTSFILE Trailer
24	Header File
25	MSHP History File - Generation Features
26	Generation Library
27	Null File
28	End of Tape Record
29	Header File
30	MSHP History File - PRD1.BASE Products
31	PRD1.BASE Sublibrary
32	Header File

Table 2. Content of z/VSE Base Tape (continued)

File	z/VSE Base Tape
33	MSHP History File PRD2.TCPIPC
34	PRD2.TCPIPC Sublibrary
35	Null File
36	End of Backup Record
37	Online Message File
38	Null File
39	End of Tape Record

Table 3. Content of z/VSE Extended Base Tape

File	z/VSE Extended Base Tape
1	Stacked Header
2	Null File
3	Header File
4	MSHP History File OS/390® API
5	OS/390 API Sublibrary
6	Null File
7	End of Backup Record
8	Null File
9	Header File
10	MSHP History File LE/VSE DBCS Locales
11	LE/VSE DBCS Locales Sublibrary
12	Null File
13	End of Backup Record
14	Null File
15	Header File
16	MSHP History File DB2/VSE Server
17	DB2/VSE Server Sublibrary
18	Null File
19	End of Backup Record
20	Null File
21	Header File
22	MSHP History File ASN/VSE Data Propagator
23	ASN/VSE Data Propagator Sublibrary
24	Null File
25	End of Backup Record
26	Null File
27	Header File
28	MSHP History File DB2/VSE Server Client Edition
29	DB2/VSE Server Client Edition Sublibrary
30	Null File

Table 3. Content of z/VSE Extended Base Tape (continued)

File	z/VSE Extended Base Tape
31	End of BACKUP Record
32	Null File
33	Header File
34	MSHP History File VSE Connectors Workstation Code
35	VSE Connectors Workstation Code Sublibrary
36	Null File
37	End of Backup Record
38	Null File
39	Header File
40	MSHP History File IPv6/VSE
41	IPv6/VSE
42	Null File
43	End of Backup Record
44	Null File
45	Null File
46	Stacked Trailer

The extended base tape for z/VSE 5.2 includes the following base products:
- OS/390 API
- LE/VSE DBCS Locals
- DB2 Server for VSE
- DB2 DataPropagator Relational Capture
- DB2 Server for VSE Client Edition
- LE/VSE DBCS Locales
- VSE Connectors Workstation Code
- IPv6/VSE

Performing Problem Analysis

If you detect a problem in your z/VSE system, you have several possibilities to analyze and solve error situations:
- Your IBM Support Center assists you in all phases of problem handling.
- Go to the z/VSE home page http://www.ibm.com/systems/z/os/zvse/
- *z/VSE Guide for Solving Problems* provides information about analyzing error situations and recovering from them.
- *z/VSE Planning* provides a list with skeletons for debugging.
- *Hints and Tips for z/VSE* discusses the different types of software errors, which might occur in a z/VSE system, and it describes the procedures available for error detection, error recording and error resolution. You can download this publication from the z/VSE website.
- The Online Problem Determination (OLPD) file contains information about CICS® transaction abends.

  * The *CICS TS Problem Determination Guide* provides information about diagnosing and resolving problems in a running CICS TS system.

Reporting a Problem to IBM

Before you call your IBM Support Center, find out:
  * if the problem is hardware- or software-related,
  * if the problem occurred in IBM shipped code or in other code. To identify this, use all the appropriate tools and resources such as traces and dumps.
  * which component is failing, if the problem is a z/VSE problem.

Receiving a PTF from IBM

Corrective service for VSE can either be received over the Internet or on physical media, depending on the size of the service package. If corrective service is received on physical media, the PTFs can be installed using the VSE Interactive Interface dialogs.

If you want to request and receive service from the Internet, refer to Appendix C, "Downloading PTFs from the Internet," on page 87 for details about preparing your VSE system.

Your IBM Support Center will advise you on when to install which type of service. When a problem is detected, contact the IBM Support Center with a description and the symptoms of the problem.

The IBM Support Center will use the information to perform a search of the IBM databases. If the problem is known to IBM and a PTF is available, IBM sends the PTF to you. It contains fixes for the problem in your system libraries. If the problem is known to IBM but no PTF is available yet, you can request to be notified as soon as the PTF is available. If the problem is not known, IBM creates an APAR and sends it to the IBM Change Team. The Change Team writes an APAR fix and creates a PTF. In some cases an APAR fix might also be created by the Change Team as a temporary solution for your problem before the PTF is generally available.

Applying Service through PTFs

Note that there are several aspects of service application via PTF:
  * Direct service application.
  * Indirect service application.
  * Service for VSE/ICCF members.
  * Service for the Generation Feature.
  * Special considerations for SVA phases.

Deciding Whether to Apply Direct or Indirect Service

Some of the *Service* dialogs described in the next chapter allow you to specify **direct** or **indirect** service. Direct service is the most common way of PTF application. It means that PTFs are merged directly into the *running system*.

"Job Sequence for PTF Application (Direct Service)" on page 28 describes in detail the job sequence used when you apply service through dialogs of the Interactive Interface.

Service will be applied indirectly if you decide to do so.

If you decide for direct application, any PTF for products residing in the system library (IJSYSRS.SYSLIB) that is flagged for indirect application cause **all** PTFs to be applied indirectly. This is done by MSHP control statements without your intervention.

However, you also can force indirect service application for system libraries, even if the PTF(s) do not require indirect service. This gives you the possibility to test the applied service before you finally merge it into your system.

When deciding whether or not to force indirect service, you should remember that indirect service application:

- Creates a "work" system library (IJSYSR**1**.SYSLIB) and copies both members from the service tape and unserviced members from the current system library to it.

  For all other system sublibraries the service is applied to the delta sublibraries. See "Job Sequence for PTF Application (Indirect Service)" on page 30 for more information.
- Requires you to shutdown your system and then IPL from the second system volume (SYSWK1).
- Requires you to run a procedure (DTRMRG) to move the contents of the work system library and the delta sublibraries to the system library, and IPL from the system library.

Thus use indirect service when a PTF requires it or when you first want to test the service that you apply to your system.

If you do not want the applied service to be activated in your system due to bad test results, you can remove the preliminary installed service by running the procedure DTRNOMRG.

Note: Do not use BASIC startup on either DOSRES (or SYSWK1) until service application is finished by running the DTRMRG or DTRNOMRG procedure.

Applying Service for VSE/ICCF Members

When service affects a VSE/ICCF member provided by z/VSE, the member is replaced. If you modify any system-provided members (such as the SUBMIT procedure), you should rename them or copy them to another VSE/ICCF library. For servicing VSE/ICCF members, do **not** delete the internally defined VSE/ICCF user "AAAA".

If you plan to modify a z/VSE skeleton, you should copy it to another library before you make your changes. Otherwise, the member will be replaced if it is affected by service.

Table 4 shows VSE/ICCF libraries and their use. Note the libraries that are reserved for z/VSE.

Table 4. Contents of VSE/ICCF Libraries

Library	Type	Contents	Usage
1	Private	VSE/ICCF administrative library. Contents shipped with VSE/ICCF.	system

Table 4. Contents of VSE/ICCF Libraries (continued)

Library	Type	Contents	Usage
2	Common	Common library. Macros and procedures. VSE/ICCF and z/VSE code members.	system
3 - 6	Public	Empty	user
7	Private	Empty	user
8	Private	Default primary library for operator profile.	user
9	Private	Default primary library for programmer profile.	user
10	Private	Default primary library for administrator profile.	user
11 - 49	Private	Empty	user
50 - 58	Public	Reserved for z/VSE.	system
59	Public	z/VSE job streams, skeletons, CICS tables, and sample programs.	system
60 - 67	Public	Reserved for z/VSE.	system
68	Public	z/VSE members for Personal Computer tasks.	system
69	Public	Reserved for z/VSE.	system
70 - 199	Private	Empty	user

Applying Service that Affects the z/VSE Generation Feature

Installation of the z/VSE Generation Feature is optional and only needed if you want a listing of the supervisor.

In general, service to supervisor generation macros consists of two PTFs. If you have the Generation Feature installed, both PTFs for the pregenerated supervisor and the Generation Feature are applied automatically.

- PTF1 applies service to the pregenerated supervisor.
- PTF2 applies service to the Generation Feature.

If you installed the Generation Feature but you keep it offline, you must restore it before you apply service which affects supervisor generation macros. Doing this ensures that all required service is applied to your system.

If you **do not** have the Generation Feature installed, only service for the pregenerated supervisor is installed.

Attention: Your generated supervisor and the Generation Feature must be at the same service level.

This means that you have to do one of the following:

- If a service refresh of z/VSE is available that is at a service level higher than your present system:
  1. Order the refresh. The Generation Feature is automatically included.
  2. Use the *Fast Service Upgrade (FSU)* dialogs to install the refresh, and select *Upgrade of Generation Feature* in the FSU dialog.
  3. In case you do not use FSU, use the *Install Generation Feature* dialog to install the Generation Feature from the refresh tapes or cartridge(s) after installing the refresh.

- If no refresh at a higher service level is available:
    1. Use the *Install Generation Feature* dialog to install the Generation Feature with your system.
    2. Reinstall any service applied to the standard supervisor. By specifying the reapplication of these PTFs, you force MSHP to apply also the related Generation Feature PTFs. It is recommended to reinstall the missing service for the Generation Feature using the PSP bucket for the current refresh.

Special Considerations for SVA Phases

When service affects VTAM, LE/VSE, or REXX/VSE, you must be aware of the following:

In z/VSE, some phases of products residing in PRD1.BASE are loaded into the Shared Virtual Area (SVA), for example, VTAM or REXX/VSE. If a PTF for such a phase is installed, this phase is **not** automatically reloaded into the SVA, because the product does not reside in the IJSYSRS.SYSLIB system library. In this case, you have to IPL your system to activate the PTF. This applies to direct application; in case of indirect application, no further action is necessary.

Chapter 2. Overview of the Service Dialogs

The following section gives you an overview of the service dialogs available with the Interactive Interface dialogs of z/VSE 5.2. The dialogs are explained in the same sequence as the selections of the main panel (Figure 1).

Use the dialogs provided by the Interactive Interface to apply service. If for any reason you do not want to use the dialogs, refer to Chapter 7, "Installing Service Changes (without Dialogs)," on page 67.

The Maintain System History Program (MSHP) refers to every program that it maintains as a *product*. Thus, both terms **program** and **product** appear in the following chapters.

To access the service dialogs, start with the *Function Selection* panel and select:

1 (Installation)

4 (Service)

Administrator Fast Path	Synonym
14	Default: **service** Yours:

The following panel is displayed:

```
 IESADMSL.IESESERV        IBM SERVICE
                                                   APPLID: DBDCCICS


   Enter the number of your selection and press the ENTER key:

        1  Verify Location of Involved Serviced Files
        2  PTF Handling
        3  Fast Service Upgrade
        4  Retrace History File
        5  Personalize History File
        6  Change Nicknames
        7  Defragmentation of History File



 PF1=HELP           3=END        4=RETURN                6=ESCAPE(U)
                    9=ESCAPE(m)

 ==> _                                            PATH: 14
```

Figure 1. Panel - IBM Service

Selections of the IBM Service Dialog

To continue with service application, select the task that you want to perform and continue as described in the applicable chapter.

1. *Verify Location of Involved Serviced Files*

   This selection allows you to change the values of some EXTENT parameters and volume serial numbers of system files involved during the service application (FSU, PTF). The dialog allows you to change the location

information of the files affected by service application. Using the dialog, you **cannot change the location itself.** Refer to "Selection 1 – Verify Location of Involved Serviced Files" on page 13 for details.

2. *PTF Handling*

   This applies PTFs from one or more service tapes or from the disk file. You can also print different types of service documentation from a service tape. In addition, you can remove a PTF using this dialog. Refer to "Selection 2 – PTF Handling" on page 17 for details.

3. *Fast Service Upgrade (FSU)*

   Traditionally, an FSU is used to install a service refresh. If you are a z/VSE 4.3.x or z/VSE 5.1.x user, you can use the FSU to install z/VSE 5.2. Refer to Chapter 5, "Performing a Fast Service Upgrade," on page 37 for details.

4. *Retrace History File*

   This prints information from the system history file. Refer to "Selection 4 – Retrace History File" on page 59 for details.

5. *Personalize History File*

   This allows you to update personalized information in the system history file. Refer to "Selection 5 – Personalize History File" on page 62 for details.

6. *Change Nicknames*

   This allows you to change nicknames of products and components. Refer to "Selection 6 – Change Nicknames" on page 63 for details.

7. *Defragmentation of History File*

   This copies the history file to the alternate history file and merges it back into an empty system history file. This merge performs a defragmentation of the file. Refer to "Selection 7 – Defragmentation of History File" on page 64 for details.

MSHP Requirements

Most of the dialogs create jobs which use MSHP to maintain service on your system. For more information about using MSHP functions, refer to Chapter 7, "Installing Service Changes (without Dialogs)," on page 67, or to *z/VSE System Control Statements*.

Note that the **minimum** partition size to run MSHP is 1024 KB, plus a 256 KB partition GETVIS area. In certain cases, this minimum size may not be enough. The size of the partition needed for MSHP depends on the number of records in the system history file, and on storage needed by involved subfunctions like LIBR or LNKEDT. The allocation of the partition should be at least 2 MB, the size should be at least 1280 K.

For information on the default partition sizes specified for predefined environments, refer to *z/VSE Planning*.

Chapter 3. Changing the Location Information of Serviced Files

This section describes the Interactive Interface dialog *Verify Location of Involved Serviced Files*. You only have to use this dialog, if you have changed the location of any system files. **Do not use it if you have not changed the location or size of the files**.

Selection 1 – Verify Location of Involved Serviced Files

This dialog allows you to change the location **information** of the files affected by service application. Using this dialog, **you cannot change the location itself**. Note, however, that the system needs accurate location information for all other IBM service dialogs. **Therefore, you must use this dialog if you have changed the location or size of any of the following files**.

Descriptive File Name	File ID	Technical File Name	VSAM catalog
System residence file	VSE.SYSRES.LIBRARY	IJSYSRS[1]	
System residence work file	SYS.NEW.RES	IJSYSR1[1]	
System history file	VSE.SYSTEM.HISTORY.FILE	IJSYSHF	
System work history file	WORK.HIST.FILE	IJSYSHF	
Job Manager file	VSESP.JOB.MANAGER.FILE	VSEJMGR	
Text repository file	VSE.TEXT.REPSTORY.FILE	IESTRFL	VSESPUC
Text repository work file	TEXT.REPSTORY.WORKFILE	IESTRWF	VSESPUC
Service File on Disk	PTF.FILE	IJSYSPF	VSESPUC

[1] IJSYSRS and IJSYSR1 must start at the beginning of the disk device.

Note: Do not put these files on virtual disks.

To access the dialog, start with the *Function Selection* panel and select:

> **1** (Installation)
>
> **4** (IBM Service)
>
> **1** (Verify Location of Involved Serviced Files)

Administrator Fast Path	Synonym
141	Default: Yours:

The system then displays the panel shown in Figure 2 on page 14. The example shown in this figure refers to a disk device of type 3390. The layout on your system may look different, depending on the type of disk device you use.

Verifying Location of Files

```
SRV$LOC1               FILE LOCATION FOR SERVICE DIALOGS

 LIST OF AFFECTED FILES
                                                        PAGE 1 OF 2


   FILENAME          VOLUME SERIAL      START        NUMBER OF
                       NUMBER          TRK/BLK        TRK/BLK

   IJSYSRS           DOSRES           _____1        __899___
   IJSYSR1           SYSWK1           _____1        __899___
   IJSYSHF (SYSTEM)  DOSRES           ___5895       ___75___
   IJSYSHF (WORK)    SYSWK1           ___900        ___75___
   VSEJMGR           SYSWK1           ___8340       ___15___




 PF1=HELP      2=REDISPLAY  3=END                5=PROCESS
               8=FORWARD
```

Figure 2. Panel 1 - Verify Location of Involved Serviced Files

The above panel applies to the following files:
- System residence file
- System residence work file
- System history file
- System work history file
- Job Manager file.

In the panel, you can change the
- Volume serial number,
- Start track/block, and
- Number of tracks/blocks

for these files.

The system issues a message telling you to do this with caution.

Note:

1. IJSYSHF (system history file) and VSEJMGR (Job Manager file) are used by other programs. This means that if you change their locations, you also should change the standard labels.

2. When you move VSEJMGR to another location within SYSWK1, you must initialize VSEJMGR at the new location. Change the standard labels and then run the following job (skeleton SKJMGRIN in VSE/ICCF library 59):

```
// JOB INITJMGR
// EXEC DTRIJMGR,PARM='/.FORMAT FORCE'
/*
/&
```

On a subsequent panel (press **PF8**), you can change the location of the text repository file and the service file on disk (see the panel shown in Figure 3 on page 15). Type the desired parameters over the existing ones and then press **PF5** to process the changes.

You can change the catalog name and catalog ID for the text repository file. To do this, type the device name over the one shown. For the service file, you can change the catalog name.

Note that if you have already pressed PF5 on the first page, you will not get the subsequent page. Press **PF5** only after you have changed the information you want on both pages. This will store your changes in internally-used tables.

If you have typed in wrong information or if you want to quit the dialog for whatever reason, press **PF3**. None of the information you have entered so far will be saved.

```
 SRV$LOC2               FILE LOCATION FOR SERVICE DIALOGS

 LIST OF AFFECTED FILES

                                                     PAGE 2 OF 2
 FILENAME: IESTRFL     FILE-ID: VSE.TEXT.REPSTORY.FILE

       CATALOG NAME:            VSESPUC

 FILENAME: IESTRWF     FILE-ID: VSE.TEXT.REPSTORY.WORKFILE

       CATALOG ID:              VSESP.USER.CATALOG
       CATALOG NAME:            VSESPUC
       VOLUME SERIAL NUMBER:    DOSRES      SYSWK1

 FILENAME: IJSYSPF    FILE-ID: PTF.FILE

       CATALOG NAME:            VSESPUC


 PF1=HELP      2=REDISPLAY  3=END                  5=PROCESS
 PF7=BACKWARD
```

Figure 3. Panel 2 - File Location for Service Dialogs

Note: Keep in mind that the dialog *Verify Location of Involved Serviced Files* does not move the files. Instead, the information gathered during this dialog is used later during the service application. If you have changed the position of any of the files affected, the service dialogs need to know the new position.

Only the Text Repository workfile is newly defined and need not be moved.

Verifying Location of Files

Chapter 4. Applying PTFs

This chapter describes the *PTF Handling* dialog available for applying PTFs to your current z/VSE system. PTFs (Program Temporary Fixes) are supplied by IBM on one or more service tapes (or cartridges), or electronically via the Internet.

Selection 2 – PTF Handling

The *PTF Handling* dialog applies PTFs from a service file residing on one or more distribution tapes or cartridges, or from the service file on disk. You will also find the term **service unit** in the dialog and in the following text. A service unit consists of one or more PTFs, either on disk or on tape (cartridge).Chapter 7, "Installing Service Changes (without Dialogs)," on page 67 provides details about the tape layout. Refer also to "Selection 3: Apply PTFs" on page 26.

You can apply PTFs in two ways:

- Apply **all** PTFs from a service file.
- Apply **selected** PTFs from a service file. If you choose this method, product and component are offered as further selection criteria.

z/VSE provides a dialog to apply PTFs for your z/VSE base system (and for a number of optional programs) which might be on **different** PTF tapes, or which reside in the service file on disk.

To access the dialog, start with the *Function Selection* panel and select:

> **1** (Installation)
>
> **4** (IBM Service)
>
> **2** (PTF Handling)

Administrator Fast Path	Synonym
142	Default: Yours:

Note: For the 'PTF.FILE' the blocksize has been changed from 80 to 10320.

The following panel is displayed:

```
 IESADMSL.IESEPTF                PTF HANDLING
                                                         APPLID: DBDCCICS
    Enter the number of your selection and press the ENTER key:


         1   Print Service Document
         2   Analyze and Apply PTFs
         3   Apply PTFs
         4   Remove PTF Records from History File




    PF1=HELP               3=END       4=RETURN                6=ESCAPE(U)
                           9=Escape(m)                    PATH: 142
 ==>
```

Figure 4. Panel - Selection for PTF Handling

The following sections describe the four selections in detail.

Selection 1: Print Service Document

This dialog allows you to print service documentation on SYSLST from a service tape, or from a file. Such information can be service tape documents, cross reference lists, and PTF cover letters. It might also contain actions that are required for PTF application.

```
 SRV$LST1                  PRINT SERVICE DOCUMENTS

  Enter the required data and press ENTER.

 SERVICE MEDIUM.............. 1           Is the service file on tape?
                                           (Enter 2 if on disk).
 VIRTUAL TAPE................ 2           Enter 1 if service tape is a
                                         remote virtual tape.
 TAPE UNIT ADDRESS........... ___         For a list of valid addresses see
                                         HELP

 Enter 2 for NO and 1 for YES to the following questions.

 PRINT DOCUMENT.............. 1           Do you want to print the service tape
                                         document?
 PRINT CROSS-REF LIST........ 1           Do you want to print the PTF cross-
                                         reference list?
 PRINT COVER LETTERS......... 1           Do you want to print PTF cover let-
                                         ters?


  PF1=HELP      2=REDISPLAY  3=END
```

Figure 5. Panel - Print Service Documents

The dialog requests the following information:

SERVICE MEDIUM

> As the service file including the PTFs can reside on tape or disk, specify the service medium. Enter 1 for tape, and 2 for disk.

VIRTUAL TAPE

> Specify if the PTF is on a virtual service tape or on a real tape.

TAPE UNIT ADDRESS

> If the service file is on tape, specify the address (cuu) of the tape unit used for the service tape.

In addition, you must specify whether or not you want the following three types of documentation printed. For each type, specify:

1 - YES (print documentation)

2 - NO (do not print documentation)

PRINT DOCUMENT
Print information about installing service from the tape. Most PTF tapes do not include information for this selection. This option is not valid for the PTF file.

PRINT CROSS-REF LIST
Print a cross-reference list of all PTFs on the tape or file.

PRINT COVER LETTERS
Print all or selected PTF cover letters with information about the errors or comments and hints.

You can start the printing of each cover letter on a new page, or you can print the ACTION info only. You need the following information:

- **ALL PTFs** Specify one of the following:

  1 Print all cover letters

  If you decide to print the cover letters of all PTFs on the tape or file, the *Job Disposition* panel is shown after you have pressed **ENTER**.

  2 Select the cover letters you want printed

  If you decide to print selected cover letters, another panel is shown after you have pressed **ENTER**. Type in the numbers of the PTFs whose cover letters you want to print. The dialog redisplays the panel until the last field on the panel is left blank.

- **CONTINUOUS** Specify one of the following:

  1 Do not start at a new page for each cover letter

  2 Start at a new page for each cover letter

- **ACTION** Specify one of the following:

  1 Print only the action comments listed in the cover letter.

  2 Print all information.

The dialog creates a job with the default name DOCPRINT. From the *Job Disposition* panel, you can submit the job to batch, file it in your default primary library, or both.

If you use the service file from tape, mount the service tape before you submit the job. Use the **same** tape address as specified in the dialog.

If the PTFs have been distributed electronically, application from disk is recommended.

Selection 2: Analyze and Apply PTFs

Use selection 2 if you want to apply one or several PTFs.

The dialog displays a list of the PTFs stored on the service tape or file. From the list, you can choose the required PTF(s) by marking each with an **X**. The dialog allows you to choose PTFs from a hierarchical order of products, components and PTFs.

Before job submission, the dialog shows you:
• Whether the PTF you need is on the service tape or file.
• Which pre-requisites are necessary for the application of a PTF.

When you select *Analyze and Apply PTFs*, the system displays a list of all the service tapes known to it, or it lists the service file on disk. An example is shown in Figure 6.

If you have not used the dialog before and no service tapes are known to your system, you first have to **add** a tape or the file to the list. Press **PF6** (Add Info) to do so. The same applies if you want to add another tape to the existing list.

If the service file is on tape, specify if it is a virtual tape, the address of the tape drive, (that is, where the tape is mounted), and the service unit name. The system then creates a job that analyzes the contents of the service unit and stores this information in internal tables.

Note: If you want to use this dialog, you must enter this information for each tape that you received from IBM.

```
 SRV$PH01               ANALYZE AND APPLY PTF
                                                          PAGE 1 of 1
 LIST OF ALL PROCESSED SERVICE UNITS

 OPTIONS: 5 = DELETE  7 = LIST PROD/COMP/PTF   8 = LIST PTFS

        OPT        UNIT-NAME        DATE-ADDED

         _         PTFCICS          10/06/04
         _         SZENST11         11/04/04
         _         SZENALL          11/04/04
         _
         _
         _
         _
         _
         _
         _

 PF1=HELP      2=REDISPLAY    3=END                   6=ADD INFO
                              9=FORGET   10=ALL PTFS
```

Figure 6. Panel - List of All Processed Service Units

The options offered in Figure 6 are discussed in detail on the following pages.

Description of PF Keys

PF3 (End)
> If you press this PF key, the system goes one level back in the hierarchy, and the action you want to perform is completed.

PF6 (Add info)
> On the *List of All Processed Service Units* panel, press **PF6** if you want to add a new service unit for service processing. The panel SRV$PH09 is displayed and you are asked for the service medium, for a unique service

unit name and, if it is a tape, for the tape unit address on which you have mounted the tape. Enter the required information and press ENTER. The *Job Submission* panel appears.

If you press PF3, the information you have entered is lost and no job will be submitted. No additional service unit is shown on panel SRV$PH01.

If you have built a job and have pressed ENTER, an additional service unit is shown (with the name you have specified) on panel SRV$PH01, showing eight asterisks in the DATE ADDED field. After you have submitted this job, you have to leave panel SRV$PH01 (by pressing PF3). Enter the panel again to get the information about the new service unit, which then shows the correct information.

Then the system scans the tape or file and stores all relevant information in a table for further processing. Note that you must perform this step to have all information about the PTF(s) on tape available.

PF9 (Forget)

If you have deleted a service unit by mistake, you can press **PF9**. You will go one level back in the hierarchy. When you enter the panel again, the PTF tape will be shown again on the list.

PF10 (All PTFs)

If you press **PF10** on the *List Service Units* panel, all PTFs from all service units are displayed in one single fulist. The panels displayed provide the same functions as *List Prod/Comp/PTF (option 7)* described below, or *List PTFs (option 8)* described in"Description of Options." The only difference is that the three possibilities start from a different level.

Description of Options

Delete a Service Unit (Option 5)

All internal tables which keep the information related to the specified service unit are deleted. Type **5** in the option column next to the service unit you want to delete and then press **ENTER**. The service unit name disappears from the list, but information is not deleted until you press **PF3=END**. If you have deleted a unit by mistake, press **PF9=FORGET**. This keeps the original tables. In the next session, the service unit name again appears on the list.

List Prod/Comp/PTF (Option 7)

This option allows you to choose the PTFs you want to apply from the hierarchical order of:

1. Products
2. Components
3. PTFs

You can decide whether you want to apply all or selected PTFs of a product/component. If you decide to apply service on the product level, the lower levels (components, PTFs) will be automatically included in the final job stream.

The dialog does not allow you to select a product that is not installed in the system.

First, the system displays a list of the products among which you can choose. An example of this list is shown in Figure 7 on page 22.

```
SRV$PH02                    ANALYZE AND APPLY PTF

LIST OF PRODUCTS

SERVICE UNIT: SZENST11                                    PAGE 1 OF 1
OPTIONS:   1 = INCLUDE PRODUCT  2 = COMPONENTS/PTF 3 = SUBLIBRARIES
           4 = AFFECTED MEMBERS 5 = REQUIREMENTS   6 = APARS    7 = EXCLUDE

OPT       NICKNAME  INCLUDED    PRODUCT-ID    RESIDENCE-LIBRARY
                                              FIRST          ADDITIONAL
 _        DSF                   DS21NM        IJSYSRS.SYSLIB
 _
 _
 _
 _
 _
 _
 _
 _
 _

PF1=HELP      2=REDISPLAY  3=END              5=PROCESS    6=NOCHECK
```

Figure 7. Panel - Analyze and Apply PTF (List of Products)

The options for the list that is created are:

Include Product (Option 1)
> This option allows you to select from the listed products those which you want to include for processing. After you type **1** in the option column and press **ENTER**, the system displays an "X" in the INCLUDED column. At the end of the session, a job stream is built which causes MSHP to apply the corresponding PTFs for these products.

Components/PTF (Option 2)
> This option leads you to the next level, which shows a list of all components for a selected product. **This level is skipped** if the product has only one component or if only one of the components receives service. The third level follows the list of components or, if the product is composed by only one component, the list of products. It lists all PTFs for a selected component/product. Select option 2 to reach both the second and the third level.

Sublibraries (Option 3)
> This option appears for both the first level (list of products) and the second level (list of components). If you select this option, the system displays the sublibrary or sublibraries affected by the PTF(s). For example, the affected sublibrary for component ID '5658-992-01' of program ICKDSF/VSE is IJSYSRS.SYSLIB.

Affected Members (Option 4)
> This option appears for all three levels (products, components and PTFs). Depending on the level from which you selected this option (which is option 4 on all three levels), the system displays the affected members by product, component or PTF. For example, if you selected option 4 from the list of PTFs, the system displays $LNKEDT.PHASE as the affected member for PTF *AB11111* (where AB11111 is a dummy name).

Requirements (Option 5)
> This option appears for all three levels (products, components and PTFs). Depending on the level from which you selected this option (which is option 5 on all three levels), the system displays the requirements for a product, component or PTF. These requirements can be prerequisites

and/or corequisites. A prerequisite, for example, is that you have to apply PTF *AB11111* before you can apply PTF *BB22222*.

> **PF10 (Status)**The panel that you get when you have selected option 5 (Requirements), lists all requirements for the selected PTF. On this panel, PF10 (Status) shows the actual status of selected PTF(s).

APARs (Option 6)

This option appears for all three levels (products, components and PTFs). Depending on the level from which you selected this option (which is option 6 on all three levels), the system displays the APARs fixed by product, component or PTF.

Exclude (Option 7)

This option appears for all three levels (products, components and PTFs). By using the EXCLUDE option, you reset the INCLUDE option that you have used before. After you type in **7** in the option column and then press **ENTER**, all PTFs that were flagged to be included are excluded again. The "X" in the INCLUDE column disappears. You can use this option if:

- You selected option 1 (include product/component/PTF) accidentally.
- You want to apply all PTFs except for the one(s) you specified.

  Example: Assume you have one product with 6 PTFs. You want to include all PTFs except for PTF number 5. Proceed as follows:

  1. On the list of products shown (Figure 7 on page 22), type in **1** (include product) next to the desired product. Press **ENTER**.
  2. On the same panel, type in **2** (component/PTF) next to the desired product. Press **ENTER**.
  3. The 6 PTFs will be listed. Type in **7** (exclude) next to PTF number 5. Press **ENTER**.

List PTFs (Option 8 on panel SRV$PH01)

This selection, from the initial *List of All Processed Service Units* panel (see Figure 6 on page 20) lists all PTFs that are stored on a service unit.

You do not have to go through the hierarchy of products, components and PTFs if you want to apply single PTFs. You can also locate a PTF by typing in the PTF number in the Locate PTF line, as shown in Figure 8 on page 24.

You can choose from the following panel options:

```
SRV$PH08                    ANALYZE AND APPLY PTF

LIST PTFS ON PROCESSED SERVICE UNIT

SERVICE UNIT: OGNLOCK                                     PAGE 1 OF 2
OPTIONS:   1 = INCLUDE PTF     3 = COMPONENT/LIBRARY   4 = AFFECTED MEMBERS
           5 = REQUIREMENTS    6 = APARS FIXED         7 = EXCLUDE PTF
OPT          PTF-NUMBER     INCLUDED

_              UN40067          X
*              UN40068          X
_              UD48399
_              UD48398
_              UD48403
_              UD48402
_              UD48395
_              UD48559

LOCATE PTF.....  _____

PF1=HELP       2=REDISPLAY  3=END                 5=PROCESS    6=NOCHECK
               8=FORWARD
```

Figure 8. List PTFs on Processed Service Tape

Include PTF (Option 1)
> This option allows you to select from the listed PTFs the ones that you want to include for processing. After you have typed in **1** in the option column and then pressed **ENTER**, the system displays an "X" in the INCLUDED column. At the end of the session, a job stream is built which causes MSHP to apply the PTF(s).

Component/Library (Option 3)
> This option lists the affected sublibrary/sublibraries for a component (for example, PRD1.BASE).

Affected Members (Option 4)
> This option lists the affected members for a specified PTF (for example, $LNKEDT.PHASE).

Requirements (Option 5)
> This option shows you a list of all requirements which must be satisfied to apply this PTF. You can query the actual status of the requirements by pressing PF10. The system then tells you whether the PTF already is applied. See the note for PF6 (NOCHECK).

> **PF10 (Status)**
>> On the panel that you get when you have selected option 5 (Requirements), all requirements for the PTF selected are listed. On this panel, PF10 (Status) shows the actual status of the PTF (whether all prerequisite and corequisite PTFs are installed).

APARs Fixed (Option 6)
> This option lists the APARs fixed by the PTF that you specified.

Exclude PTF (Option 7)
> You can use this option if you selected option 1 (include PTF) accidentally for a PTF.

Description of PF Keys

The following additional **PF keys** are used in the panel shown in Figure 7 on page 22 and in subsequent panels:

PF3 (End)
> If you press this PF key, all data that you have entered on the panel up to

now is saved. The system goes one level back in the hierarchy. If you are on the highest level in the hierarchy and you press **PF3**, a message informs you that all your input will be lost if you press PF3 again. If you do not want to loose your data, press **PF5=PROCESS**. This creates a job stream.

PF5 (Process)

This PF key is only offered on the highest level in the hierarchy, when all necessary data is available for processing.

Use **PF5** to check whether the PTF(s) selected for application are already applied. (This is similar to the PF10=STATUS key in the requirements option.)

This process does **not** check all requirements defining whether a PTF can be applied or not. When you use PF5, only the current status of the selected PTF(s) is checked (applied or not applied). Depending on the amount of service on your tape(s) this takes a while.

PF6 (NOCHECK)

When lists of products, components, or PTFs are displayed, you can use **PF6** if you do **not want to check** whether the PTF(s) are already applied.

The following is valid for PF5 (PROCESS) and PF6 (NOCHECK): The system creates the *PTF Apply* job stream. The following information panels are displayed:

- List of PTFs already applied (if you used PF5=PROCESS).

  The panel shows all selected PTFs that are already installed in the system, if there are any.

- List of PTFs to be applied (if you used either PF5 or PF6).

  The panel shows the final list of PTFs that are applied when the job stream is run.

Attention: The reapply option lets you force a new installation of these PTFs. Be careful when you use it, because you can downlevel your code with it.

In the final step in creating the job stream, the system asks you for:
- The tape unit (if the service file is on tape), you want to use for mounting your PTF tapes.
- The number of service tapes you want to have processed by this PTF application.
- Whether you want to have a backup of all libraries used for the base and optional programs listed in the system history file.
- The type of application for each PTF on a service unit is predefined. PTFs for programs **not** residing in the system sublibraries IJSYSRS.SYSLIB, PRD1.BASE, PRD1.MACLIB, PRD2.SCEEBASE or PRD2.GEN1 (if available) are always applied directly. PTFs for the programs in the system libraries can specify direct or indirect application.

  You should accept the predefined application by entering **2 (NO)**. You can, however, force indirect application of all PTFs for programs residing in the above listed system sublibraries by entering **1 (YES)**.

The dialog creates a job sequence that is managed by the Job Manager with the default VSE/ICCF member name APPLYSRV. The included VSE/POWER jobs start with prefix DTRPTF. On the *Job Disposition* panel, you can submit the job to batch, file it in your default library, or both.

"Job Sequence for PTF Application" on page 28 describes how to proceed, when you want to apply PTFs.

Selection 3: Apply PTFs

This dialog creates a job sequence that is managed by the Job Manager. "Job Sequence for PTF Application" on page 28 has information about Job Manager processing and other considerations for this dialog.

If you chose selection 3 from the *PTF Handling* panel, a service unit (on tape or file) is used to correct one or more **specific** problems. In this case, the service unit is not analyzed as in selection 2. You can decide whether you want one or more tapes to be processed.

The dialog asks you for the following:

SERVICE MEDIUM
> As the service file including the PTFs can reside on tape (or cartridge), or on disk, specify the service medium. If the PTFs have been distributed electronically, application from disk is recommended. Enter 1 for tape, and 2 for disk.

VIRTUAL TAPE
> Enter 1 if the PTF is on a virtual service tape. If the PTF is on a real tape or on the service file, enter 2.

TAPE UNIT ADDRESS
> If the service file is on tape, specify the address (**cuu**) of the tape drive.

TAPE QUANTITY
> If the service file is on tape, specify the number of service tapes you have for the PTFs. If all requirements are met for the PTFs you wish to apply, you are not asked for the number of tape mounts during job execution. Perform the following steps for mounting the tapes:
>
> 1. Mount **all** tapes at the beginning, as the system first scans all mounted tapes.
> 2. Mount the tapes again in sequence for the PTF application.
> 3. If you have mounted two tapes and are prompted the third time, mount tape 1 again. At the fourth prompt mount tape 2. If you have to process three or more PTF tapes, refer to the *z/VSE System Control Statements* for instructions on how to do so.

TYPE Specify whether you want to apply all or selected PTFs of a particular tape. You can choose among the following:

> **1=ALL**
> > All PTFs of the corresponding tape are applied.
>
> **2=INCLUDE**
> > Another panel will be displayed on which you can specify all the PTFs you want to apply.
>
> **3=EXCLUDE**
> > A panel is displayed on which you can specify all the PTFs you want to exclude. All remaining PTFs are applied.

BACKUP
> Specify whether you want to backup **all** the libraries used for base and optional programs listed in the system history file. If you specify **1** (YES,

backup) **all** sublibraries which were installed by installation dialogs are
backed up. Enter one of the following:

1   - YES (Backup taken)

2   - NO (Backup is not taken)

FORCE INDIRECT
> With this option, you can force indirect application of all PTFs for
> programs residing in the system libraries IJSYSRS.SYSLIB, PRD1.BASE,
> PRD1.MACLIB, PRD2.SCEEBASE, and PRD2.GEN1 (if available).
>
> With indirect application you can first test the applied service and then
> accept executing the merge procedure.
>
> As the direct method is faster, it is recommended to enter **2 (NO)** to accept
> predefined defaults.

ADD INFO TO THE LIST
> Note that this option is only available for one tape, that is, if you specified
> TAPE QUANTITY = 1. If you specify **1**=YES, the system asks you for a
> unique service unit name. If it is a tape, specify if it is a virtual tape and
> also the tape drive on which you have mounted the tape. Then the system
> scans the service unit and stores all relevant information in a table for
> further processing. Note that for the dialog *Analyze and Apply PTFs*, you
> must perform this step to have all information about the PTF(s) on tape
> available. Specify one of the following:

1   - YES (Add new service unit for service processing)

2   - NO (Do not add new service unit)

The dialog creates a job with the default name APPLYALL, consisting of
VSE/POWER jobs starting with the prefix DTRPTF. From the *Job Disposition* panel,
you can submit the job to batch, file it in your default library, or both.

"Job Sequence for PTF Application" on page 28 describes how to proceed, when
you want to apply PTFs.

Selection 4: Remove PTF Records from History File

This dialog allows you to remove PTF records for one or more components from
the system history file. **The fix itself, however, remains in the system** after the job
stream is executed.

Note: This is an exceptional case and only should be done on IBM's advice. You
have to use this function, for example, if a PTF is in error and you need to reapply
the (correct) PTF with the **same** PTF number.

This selection of the *PTF Handling* panel displays a list of all components installed
in the system. You now specify the components that are related to the PTFs which
you want to remove from the system history file. Type **1** in the option column next
to these component(s). The next panel allows you to specify the PTF you want to
remove.

Optionally, you can decide whether you want to verify immediately that the
specified PTF(s) actually is/are applied. If you want the system to check in the
system history file before generating a job which removes the PTF(s), enter **1** in the
Verify PTF field. Then press **ENTER**. If you are sure that the PTF number is correct

and you want to bypass the verification, enter **2** in the same field. The job which removes the PTF(s) is then generated without verification.

Once you have specified all PTFs you want to remove, press **PF5** to create a job which removes the history record(s) of the specified PTF(s) from the system history file.

Job Sequence for PTF Application

The following information refers to the *PTF Handling* dialog and its selections:
- Analyze and apply PTFs
- Apply PTFs

1. The dialogs create a job sequence which runs under the control of Job Manager. If a problem occurs and you **cannot** resume or reset, delete the following jobs in the VSE/POWER reader queue:
   - DTRCLPTF
   - All jobs beginning with the prefix DTRPTF (DTRPTF01, DTRPTF02 and so on).

   If you do not do this, the Job Manager might not work correctly at a later time. Refer to *z/VSE Installation* for more information on the Job Manager.

2. The job sequence runs in the background partition (BG). This is the default. You should start BG with **CLASS=0**. If you prefer to use a different class, you can specify this in the *Job Disposition* panel, as long as you select a class that was started in VSE/POWER. The partition size should be at least 1280 K.

3. Only one job sequence from the dialogs must be in the reader queue at one time.

4. If you use the automatic startup facility, do **not request a BASIC start** during IPL from IJSYSR1 (part of indirect PTF application). This is not possible because the VSE/POWER queue files and the VSE/POWER data file reside on the same location as IJSYSR**1**.SYSLIB. For a detailed description of the different startup modes, refer to *z/VSE Operation*.

5. If service affects a Generation library, the library must be online. If it is not online, the job fails. In this case, do the following to stop the job sequence temporarily and exit from Job Manager:
   a. Enter **x EXIT** (where *x* is the reply ID).
   b. Restore the Generation library using the *Install Generation Feature* dialog (Fast Path **13**).
   c. Enter **R RDR,DTRPTFAB** to restart apply service and to proceed with service application:
   d. Enter **x RESUME** (where *x* is the reply ID).

6. You should have a current backup of your system available, in case service application is not successful. If you do not have one, specify **1** (YES) in the BACKUP field of the *Apply PTFs from Service Unit* panel.

Job Sequence for PTF Application (Direct Service)

This is the method for applying service for products not residing in the system libraries IJSYSRS.SYSLIB, PRD1.BASE, PRD1.MACLIB, PRD2.SCEEBASE and PRD2.GEN1. You also can use direct service application for products residing in the system libraries, if you want to merge the service directly into the running system. You cannot test or verify the service before final application, when you apply it directly.

When you use the dialogs, the following jobs are created:

DTRPTF01

This job ensures that any previous application of indirect service was completed. It is especially important for users who applied indirect service and did not merge this service. Refer to "Job DTRPTF01 – Dialog Problem Checking" on page 34 for details.

DTRPTF02

If requested, this job performs a backup of all libraries. The system library IJSYSRS is backed up with the option RESTORE=STANDALONE and is the first one on the backup tape.

DTRPTF03

This job copies the system history file to the work history file.

IJSYSR**1**.SYSLIB is created to prepare special treatment of service for products residing in IJSYSRS.SYSLIB. This is done to avoid possible space problems in the system library.

The MSHP residence information of the work history file for products residing in the following libraries is changed to a new name:

    IJSYSRS.SYSLIB – IJSYSR**1**.SYSLIB

    PRD1.BASE – PRD1.BASE**D**

    PRD2.GEN1 – PRD2.GEN1**D** (if available)

    PRD1.MACLIB – PRD1.MACLIB**D**

    PRD2.SCEEBASE – PRD2.SCEEBAS**D**

These libraries are later on referred to as **delta sublibraries**.

MSHP is executed to test for PTF requirements and to have the PTFs integrated into the appropriate product sublibraries. Due to the changed residence information, service applied to one of the four system sublibraries mentioned above is cataloged into the corresponding delta sublibraries.

All changes are recorded by updating the work history file.

DTRPTF04

After MSHP has successfully applied all PTFs, all members in the delta sublibraries are moved into their corresponding base sublibraries. Existing members are replaced. This means direct update of the system sublibraries of a running system.

Service is merged directly into the system. The MSHP residence information of the work history file is reset.

DTRPTF05

For VSE/ICCF members the job will disconnect the DTSFILE. There is a PAUSE statement to check if disconnecting was successful.

Create a backup of your VSE/ICCF DTSFILE. For servicing VSE/ICCF members, do not delete the internally defined VSE/ICCF user "AAAA".

DTRPTF06 and DTRPTF07

Further processing for the text repository file.

DTRPTF08

The work history file is copied back to the system history file.

DTRPTF09 and DTRCLPTF

These jobs are cleanup jobs for PTF application.

Direct Service application is complete.

Note:

- PTFs applied directly to z/VSE components (such as CICS, VTAM, or VSE/POWER) might require a restart of that component or – in the case of VSE/POWER – an IPL.
- If the PTF requires a VSE/POWER COLD start, you have to POFFLOAD your VSE/POWER queues prior to IPL (POFFLOAD BACKUP,ALL,cuu). Generally, such PTFs are flagged for indirect service application.

Further Processing (Direct Service)

1. After applying service to user-modified or CICS generated programs and tables, regenerate startup procedures and VTAM® books.

   To access the dialog, start with the *Function Selection* panel and select:

   2 (Resource Definition)
   4 (Hardware Configuration and IPL)
   1 (Configure Hardware))

Administrator Fast Path	Synonym
241	Default: Yours:

   On the *Configure Hardware* panel, press **PF5**. Select the startup procedures you want to regenerate. The dialog then generates a job which assembles the required books. The dialog is described in *z/VSE SNA Networking Support*.

   Review the output of the job to make sure everything is correct. Error messages in the job output might indicate difficulties for a subsequent IPL.

2. After PTF application, the updated text repository file must be activated. See the instructions for updating default selection panels and application profiles under "Post-Stage 2 Processing" on page 50.

3. If the output of the *Print Service Documents* dialog indicates any action items, you should perform these action items now. For example:
   - updating the CSD file
   - updating the transaction security definition
   - updating the application profiles and selection panels.

Job Sequence for PTF Application (Indirect Service)

This is the method for applying critical service for products residing in the system sublibraries IJSYSRS.SYSLIB, PRD1.BASE, PRD2.SCEEBASE, PRD2.GEN1 (if available), and PRD1.MACLIB. It allows to test the serviced system and to remove this service in case of failure.

Depending on the type of service you should backup the DTSFILE and the text repository file, in order to make sure you can switch back to the level before the service application. See also job DTRPTF05 and DTRPTF06.

This method is used automatically if at least one PTF for IJSYSRS.SYSLIB is flagged for indirect service application.

When you use the dialogs, the following jobs are created:

DTRPTF01
    This job is important for users who applied indirect service before and did

not merge this service. If this applies to you, please refer to "Job
DTRPTF01 – Dialog Problem Checking" on page 34 for more information.

DTRPTF02
If requested, this job performs a backup of all libraries. The system library
IJSYSRS is backed up with the option RESTORE=STANDALONE and is
the first one on the backup tape.

DTRPTF03
This job copies the system history file to the work history file.

IJSYSR**1**.SYSLIB is created to prepare special treatment of service for
products residing in IJSYSRS.SYSLIB. This is done to avoid possible space
problems in the system library. It is also necessary to prepare for indirect
service application.

The MSHP residence information of the work history file for products
residing in the following libraries is changed to a new name:

> IJSYSRS.SYSLIB – IJSYSR**1**.SYSLIB
>
> PRD1.BASE – PRD1.BASE**D**
>
> PRD2.GEN1 – PRD2.GEN1**D** (if available)
>
> PRD1.MACLIB – PRD1.MACLIB**D**
>
> PRD2.SCEEBASE – PRD2.SCEEBAS**D**

These libraries are later on referred to as **delta sublibraries**.

MSHP is executed to test for PTF requirements and to have the PTFs
integrated into the appropriate product sublibraries. Due to the changed
residence information, service applied to one of the four system
sublibraries mentioned above is cataloged into the corresponding delta
sublibraries.

All changes are recorded by updating the work history file.

DTRPTF04
After MSHP has successfully applied all PTFs to IJSYSR1.SYSLIB, all
unchanged members are copied from IJSYSRS.SYSLIB to IJSYSR**1**.SYSLIB.
Thus a new system sublibrary is created, usually on volume SYSWK1,
containing the new service. The system sublibraries of the running system
are not changed yet. The MSHP residence information of the work history
file is reset.

DTRPTF05
For VSE/ICCF members the job will disconnect the DTSFILE. There is a
PAUSE statement to check if disconnecting was successful.

Create a backup of your VSE/ICCF DTSFILE. For servicing VSE/ICCF
members, do not delete the internally defined VSE/ICCF user "AAAA".

DTRPTF06 and DTRPTF07
Backup your text repository file so that you can restore it if you do not
merge service later.

This job further processes the text repository file.

DTRPTF08
This job catalogs a procedure to merge service, and it instructs you to shut
down your system. A new set of LIBDEF and LIBSDL procedures is used
for the startup on SYSWK1. The new set is LIBDEF**S** and LIBSDL**S** which
contains the so-called delta sublibraries PRD1.BASED, PRD2.SCEEBASD,
PRD2.GEN1D, and PRD1.MACLIBD.

DTRPTF09 and DTRCLPTF
These jobs are cleanup jobs for PTF application.

Note:

- If the PTF requires a VSE/POWER COLD start, you have to POFFLOAD your VSE/POWER queues prior to IPL (POFFLOAD BACKUP,ALL,cuu).
- If you have the Generation Feature installed and use your own supervisor **and** the supervisor is affected by this service application, reassemble your own supervisor into IJSYSR1.SYSLIB **before** shutting down your system. Insert PRD1.MACLIBD and PRD2.GEN1D in the LIBDEF chain sequence **before** PRD2.GEN1.

Further Processing (Indirect Service)

1. After applying service to user-modified or CICS generated programs and tables, regenerate startup procedures and VTAM books.

   To access the dialog, start with the *Function Selection* panel and select:

   > **2** (Resource Definition)

   > **4** (Hardware Configuration and IPL)

   > **1** (Configure Hardware))

Administrator Fast Path	Synonym
241	Default: Yours:

   On the *Configure Hardware* panel, press **PF5**. Select the startup procedures you want to regenerate. The dialog then generates a job which assembles the required books. The dialog is described in *z/VSE SNA Networking Support*.

   Review the output of the job to make sure everything is correct. Error messages in the job output might indicate difficulties for a subsequent IPL.

2. After PTF application, the updated text repository file must be activated. See the instructions for updating default selection panels and application profiles under "Post-Stage 2 Processing" on page 50.

3. If the output of the *Print Service Documents* dialog indicates any action items, you should perform these action items now. For example:

   - updating the CSD file
   - updating the transaction security definition before you merge
   - updating the application profiles and selection panels.

4. IPL from the disk device on which IJSYSR1 resides (default is SYSWK1).

Testing and Merging Service

You should now test the service application. If your test completes successfully, you can merge the service into your system. This is described in "Merge Service into Your System" on page 33.

If your test is not successful, you do not have to merge the service contained in the delta sublibraries. You then continue as described in "Do not Merge Service into Your System" on page 33, and you can return to your original system.

If the DTSFILE or the text repository file have been updated, you have to restore the previously made backup, as this is not done by the system automatically.

Merge Service into Your System

If testing was successful, do the following to finish PTF application (if started from SYSWK1):

- Run the DTRMRG procedure by entering:

  R RDR, PAUSEBG

  0 EXEC PROC=DTRMRG

  The DTRMRG procedure changes the system sublibraries permanently. The work history file is copied into the system history file.

  All members in delta sublibraries are moved into their corresponding base sublibraries. Existing members are replaced.

  Sublibraries PRD1.BASED, PRD1.MACLIBD, and PRD1.SCEEBASD are excluded from the LIBDEF chains, contained in members LIBDEF.PROC and LIBSDL.PROC of IJSYSRS.SYSLIB.

  The original set of procedures LIBDEF and LIBSDL is established in the startup without the delta sublibraries.

- Shut down your system.
- IPL from DOSRES.

Note: You must not do any kind of service application as long as you are IPLed to IJSYSR**1**.SYSLIB. For example, you should not apply local fixes.

As long as the service is not merged, service application or FSU will delete the changed parts from IJSYSR1 and the delta libraries. Changes from the history file will also be deleted.

Do not Merge Service into Your System

If testing was not successful, you should not merge the service. Do the following instead (if started from SYSWK1):

- Run the DTRNOMRG procedure by entering:

  R RDR, PAUSEBG

  0 EXEC PROC=DTRNOMRG

  Sublibraries PRD1.BASED, PRD1.MACLIBD, PRD1.SCEEBASD and PRD2.GEN1D (if available) are cleared. Products residing in one of the delta sublibraries are removed from the work history file. For non-base programs, the work history file is merged into the system history file.

- Shut down your system.
- IPL from DOSRES (your old system resident volume).
- Restore the VSE/ICCF DTSFILE from the backup if its members were affected by the service application.
- Restore the text repository file from the backup if it was affected by the service application.

Note:

1. You do **not** have to restore the **system history file** which has not yet been changed by service application. The final update for the system history file also is done with the job DTRPTF01.

   As far as the system sublibraries are concerned, your system is at its original service level. This is the level before you started the PTF application.

2. If you do not merge your delta sublibraries, but service applied to any non-base program sublibrary depends on an updated system sublibrary (due to a corequisite or prerequisite situation), you have to restore the related sublibrary, too.

Job DTRPTF01 – Dialog Problem Checking

This job is especially important for users, who:
- Applied indirect service to their previous z/VSE system and
- Did not merge this service into the system.

The job sequence checks that indirect PTFs from a previous PTF application have been merged before it installs new PTFs. You might be asked to interfere.

If you did not apply indirect service before, this job also runs, but you do not have to interfere.

If indirect service has not been merged, the job stops and displays the following message:

```
PREVIOUS INDIRECT SERVICE APPLICATION HAS NOT BEEN MERGED TO THE SYSTEM
```

This message is to:
- Inform you that indirect service application was not finished successfully.
- Ask you to do one of the following:
  - Cancel the job.
  - Continue with the job.

The message **does not** necessarily mean there is a problem. It indicates that the DTRMRG procedure was **not** run after a **previous** PTF application.

You might not have run DTRMRG for two reasons:

1. **Prior PTF testing was not successful, and you did not want to merge the service**.

   In this case, you did not want service applied, so you did not run DTRMRG (nor DTRNOMRG). You can continue with the current (new) PTF application. The system makes sure that the previously applied services are deleted.

   Before you reply to the message, make sure you restored the VSE/ICCF DTSFILE and the text repository file to the status they had before the **previous** PTF application.

   Continue as follows:

   a. VSE/ICCF DTSFILE (if affected) and text repository file (if affected) are restored.

      Type in:

      x

      where *x* is the reply ID

      Then press **ENTER**.

      Sublibraries PRD1.BASED, PRD1.MACLIBD, PRD2.SCEEBASD, and PRD2.GEN1D (if available) are cleared. Products residing in one of the delta sublibraries are removed from the work history file. For non-base programs, the work history file is merged into the system history file.

      The current PTF application continues.

b. If affected VSE/ICCF DTSFILE and text repository file are not restored, do the following:

- Cancel the current PTF job and exit from the Job Manager.
- Restore VSE/ICCF DTSFILE and text repository file.
- Release the job **DTRPTF01** from the reader queue.

   The new PTF application continues.

   Note: If you do not follow the above instructions, you might have mixed service levels in the different libraries on your system.

2. **Prior PTF testing was successful, but you did not run DTRMRG to complete PTF application**.

   Perhaps you overlooked the information about running DTRMRG. Service has been installed, but it has not been merged.

   In this case, you should do the following:

   - Cancel the current PTF job and exit from the Job Manager.
   - Shutdown the system (if you did not IPL from the disk device where IJSYSR1 resides, which normally is SYSWK1).
   - IPL from the disk device with IJSYSR1 (which normally is SYSWK1).
   - Run the DTRMRG procedure:

         R RDR,PAUSEBG
         0 EXEC PROC=DTRMRG

   Previous service is merged into your system.

   - Shutdown the system.
   - IPL from DOSRES.
   - Release the job **DTRPTF01** from the reader queue.

   The new PTF application continues.

   Note: If you omit doing this, you might have mixed service levels in the different libraries on the system.

Dialog Problem Checking

Chapter 5. Performing a Fast Service Upgrade

The traditional use for the Fast Service Upgrade (FSU) process is to upgrade your z/VSE system to the current maintenance level. This means that you perform an FSU to install a new **modification level** of z/VSE – referred to as **performing a service refresh**.

If you are currently a z/VSE 4.3 or a z/VSE 5.1 user and you want to install z/VSE 5.2, you can use the FSU process for installation as well. Installing a new version/release of z/VSE through FSU is generally referred to as **performing a release upgrade**. Make sure that you have read the topic "Installing z/VSE" in *z/VSE Planning*. It describes planning tasks and prerequisites required for performing a release upgrade through FSU.

The FSU consists of a dialog sequence that you follow. Please proceed as described in the following table, depending on whether you want to perform a release upgrade or do a service refresh.

Table 5. Version/Release Upgrade or Service Refresh

Version or Release Upgrade	Service Refresh
Do not perform a downlevel check	**Perform a downlevel check** Your current system status is checked and compared with the refresh level. This step does not change your system.
Perform the FSU preparation A job stream is created which restores the latest level of the FSU function from tape. In this case, it is the FSU function from the new z/VSE 5.2 system.	**Perform the FSU preparation** A job stream is created which restores the latest level of the FSU function from tape.
Perform the FSU installation Your system is replaced by the new z/VSE version. The installation of a release upgrade is mainly the same as the one for a service refresh. FSU installation consists of two stages. See "What You Must Know Before You Start with the Fast Service Upgrade" and "FSU Processing and Space Requirements" on page 40 for more information.	**Perform the FSU installation** Your system is replaced by the new service refresh. FSU installation consists of two stages. See "What You Must Know Before You Start with the Fast Service Upgrade" and "FSU Processing and Space Requirements" on page 40 for more information.

What You Must Know Before You Start with the Fast Service Upgrade

Before you use the Fast Service Upgrade dialogs, be aware of the following:

Modifying or Adding Members

- You should not add your own members to the VSE/ICCF system libraries. Also, you should not modify system-provided members in the VSE/ICCF system

libraries. These libraries are reserved for general use of z/VSE. For more information about the VSE/ICCF system libraries, refer to *z/VSE Planning*.

If you modify any VSE/ICCF members supplied by IBM, such as the SUBMIT procedure, you should save them in a user library. For servicing VSE/ICCF members, do not delete the internally defined VSE/ICCF user "AAAA".

- IBM supplied members which you modify by using specific IBM supplied skeletons or dialogs can automatically be saved in PRD2.SAVE (for example, skeletons for system startup tailoring). PRD2.SAVE is for FSU only.

The following members are automatically stored in the system sublibrary PRD2.SAVE:

DTSECTXN.PHASE [1]	$8JCL.PROC
DTSECTXS.A[1]	$8JCLBSX.PROC
DTSECTAB.PHASE (in a secured system)	$9JCL.PROC
$AJCL.PROC	$9JCLBSX.PROC
$AJCLBSX.PROC	ALLOC.PROC
$BJCL.PROC	CPUVAR1.PROC
$BJCLBSX.PROC	DTRCICST.PROC
$IPLESA.PROC	DTRICCF.PROC
$0JCL.PROC	DTRINFOA.PROC
$1JCL.PROC	DTRPOWR.PROC
$1JCLBSX.PROC	LIBDEF.PROC
$2JCL.PROC	LIBSDL.PROC
$2JCLBSX.PROC	PWBSXL.PROC
$3JCL.PROC	PWSPROF.PROC
$3JCLBSX.PROC	STDLABEL.PROC
$4JCL.PROC	STDPROF.PROC
$4JCLBSX.PROC	USERBG.PROC
$5JCL.PROC	...
$5JCLBSX.PROC$6JCL.PROC	plus all user modifications in startup procedures, for example, VSE/POWER startup, if the skeletons were used.
$6JCLBSX.PROC	...
$7JCL.PROC	plus printer FCBs and UCBs.
$7JCLBSX.PROC	

([1] = in case of a Release Upgrade.)

If you modify other IBM supplied members of IJSYSRS, **do not save** a copy of the original member **or** your changed version in **PRD2.SAVE**. If you want to save copies of the original member and your changed version, you should create another VSE sublibrary and catalog them there.

- You should not add your own members to the system libraries IJSYSRS, PRD1 and PRD2.SCEEBASE. These libraries are completely replaced (except for procedures).
- If for certain reasons installation information (such as user generated phases, private VSE/ICCF phases, your own VSE/POWER phases, own VSE/POWER startup procedures, FCBs, UCBs, ASI procedures, and standard labels) must be cataloged in IJSYSRS, these members must also be cataloged in PRD2.SAVE. They have to be cataloged under a name other than any IBM supplied member name in IJSYSRS and PRD2.SAVE, otherwise your changes are lost during installation of a service refresh or version upgrade.
- If you do not use IBM's default security manager BSM, it is possible that the new system, especially CICS TS, cannot be started unless the security server is upgraded to the latest level. It is recommended to upgrade the security server

before the FSU is started. It might also be helpful if the external security manager is switched off for the first IPL from DOSRES after FSU.

Note: If you generate your own POWER phase, it should not be stored into PRD2.SAVE but into PRD2.CONFIG.

It is **your responsibility** to check whether service has been applied to the IBM supplied members that you have modified. Also, you must ensure that the members work correctly when you have modified them.

Release Upgrade: You should not have any phases in sublibrary PRD2.SAVE (especially VSE/POWER phases).

Having Backups Available

In general, it is recommended to have a backup of the system disks DOSRES and SYSWK1 available. You can create such a backup by using the VSE/Fast Copy program, as described in *z/VSE System Utilities*.

- It is useful to have a backup of the VSE/ICCF DTSFILE and the libraries IJSYSRS and PRD1. If you use the *Fast Service Upgrade* dialog to create these backups, a backup of the system history file is included.
- Furthermore, it is useful to have a backup of the following files (or libraries), they are changed during FSU and might also contain user-specific data:
  - VSE/VSAM file CSD (CICS System Definition file DFHCSD)
  - VSE/VSAM file VSE.TEXT.REPSTORY.FILE
  - PRD2.GEN1 (if the Generation Feature is installed)
  - BSTCNTL file VSE.BSTCNTL.FILE. Use job skeleton SKBSTSAV to save security information in format of BST Admin Commands

  Do not restore any of these after FSU has completed successfully.

  Other files, like the VSE.MESSAGES.ONLINE VSE/VSAM file can be restored (if necessary) from the old installation tape. This is only possible if you do **not** have messages from vendor programs stored in the online message file. If you do, please make a backup of the online message file.
- It is useful to create a LISTLOG of your hardcopy file.

Performing Optional Tasks

- You cannot install the Generation Feature through a service refresh if it was not installed before.

  If the Generation Feature is installed, it must reside in library PRD2.GEN1. You are asked during the FSU whether you want to refresh it. If you select to refresh it, the Generation Feature will be replaced. If you choose not to refresh it, the entry for the Generation Feature is removed from the system history file. All members in the related sublibrary will be deleted.
- An FSU is not intended for optional programs, or for programs residing on the extended base tape. If you order a z/VSE refresh, you should also reorder any updated VSE optional programs or other VSE licensed programs that you have installed. If you also want to refresh your VSE optional programs, you must **reinstall** them after having gone through the FSU process. Reinstalling these programs after refreshing z/VSE ensures that all IBM licensed programs on your system are at the same level.

FSU Processing and Space Requirements

1. The z/VSE library structure must exist. This means that the FSU depends on having IJSYSRS.SYSLIB, PRD1, PRD2.CONFIG, PRD2.SCEEBASE, PRD2.SAVE, and PRD2.TCPIPC available. If you have combined or renamed these libraries and sublibraries, the FSU job stream will not work without modification.

   System volumes must be DOSRES and SYSWK1. Also, the VSE/VSAM user catalog VSESPUC must be available.

2. Do not run FSU stage 1 if you performed a BASIC startup. It will overlay your VSE/POWER files.

3. The FSU runs per default in the background partition (BG). You can, however, run the job stream in any partition as long as the partition leaves enough space for MSHP (at least 1280KB, plus 256KB partition GETVIS area), except partition FB. However, in stage 2 the jobs are forced to run in BG and only the following partitions are started: BG, F1, F2, F3, F4, F5, and FB with the basic security manager.

4. The FSU job sequence is managed by the Job Manager. Only one job sequence from the stage 1 dialogs should be in the reader queue at one time.

5. In stage 2 the FSU defines a work file for the text repository file. The work file is defined in VSAM space. You should make sure that your system has sufficient VSAM space on the disk device you have specified in the dialog *Verify Location of Involved Serviced Files* for the definition of this work file. To check this:

   - Verify if you have changed the location of the VSAM catalog in the dialog *Verify Location of Involved Serviced Files*.

   - Estimate the required space for the work file by looking at the space your current text repository file uses. Access the *File and Catalog Management* catalog, enter VSESPUC (or the catalog name you want to use) for CATALOG NAME, choose selection 1 and use option 1.

   - Check how much VSAM space is free in the VSESPUC catalog you have specified in the *Verify Location of Involved Serviced Files* dialog (Fast Path 141) on the specified disk device. The default is VSESPUC. Choose selection 5 of the *File and Catalog Management* dialog and use option 1. If there is not sufficient space left, use option 4 to define new space.

   For the Text Repository workfile, you do not need to define or move any clusters. FSU uses the data specified for it to do so.

Preparing FSU from a Virtual Tape

If you plan to install a service refresh and you received it on DVD or electronically, you can use the virtual tape that has been loaded into VSAM. For a description about how to load the base tape file into VSAM, see Appendix D, "Preparing for Initial Installation or FSU from Virtual Tape," on page 91. The z/VSE Extended Base Tape and Optional Products can be installed directly from the PC using the virtual tape format on a remote server (workstation). These can also be loaded into VSAM and installed from there.

Before Starting the FSU

Note that a TPA tape drive cannot be added with mode "05" any longer. If you have specified mode "05" in a previous z/VSE release, update your IPL procedure (Fast Path 241) with valid modes before starting an FSU to z/VSE 5.2. For a list of valid mode settings refer to the ADD statement in *z/VSE System Control Statements*.

If you start an FSU without replacing mode "05 " in your IPL procedure, you will receive an error message. For details on how to resolve this error, refer to "Problems During Stage 2" on page 55.

z/VSE 5.2 supports a maximum of approximately 100 SCSI disks during IPL, therefore check how many SCSI disks you have defined in your previous system. The actual number of supported devices might be less depending upon your system configuration. If the number of SCSI disks defined in your IPL procedure is larger than the maximum, you have to remove SCSI disks that are not needed during IPL. They can be defined after IPL has completed using the AR/JCL SYSDEF SCSI statement in the BG procedure. However, **do not** remove SCSI system disks (DOSRES, SYSWK1, PAGEDATASET and lock file) that are used during IPL.

Proceed as follows to change the IPL procedure and remove non-system SCSI disks exceeding the limit:

- Access the *Configure Hardware* panel using Fast Path **241**.
- Select a FBA-SCSI device from the list.
- Select option **3** = SELECT FOR FURTHER PROCESSING
- Select option **8** = Specify DEF SCSI command
- Select option **5** = DELETE (To delete the DEF SCSI definition of the non-system SCSI disks from the IPL procedure.)

To change the BG procedure update the SYSDEF SCSI commands (SKJCL0 in ICCF 59) in the $0JCL procedure.

Selection 3 – Fast Service Upgrade

To perform a service refresh, use the *Fast Service Upgrade* dialogs. These dialogs create job sequences to install the service refresh. A job sequence is managed by the Job Manager.

To access the dialog, start with the *Function Selection* panel and select:

    1 (Installation)

    4 (IBM Service)

    3 (Fast Service Upgrade)

Administrator Fast Path	Synonym
143	Default: Yours:

The following panel is displayed:

```
IESADMSL.IESEFSU          FAST SERVICE UPGRADE
                                                       APPLID: DBDCCICS
   Enter the number of your selection and press the ENTER key:


       1  Down-Level Check

       2  FSU Preparation

       3  FSU Installation



PF1=HELP              3=END        4=RETURN              6=ESCAPE(U)
                      9=ESCAPE(m)

==> _                                             PATH: 143
```

Figure 9. Panel - Selection for Fast Service Upgrade

Option 1 – Down-level Check for a Service Refresh

The selection *Down-Level Check* creates a job stream which performs a down-level check **without installing** the service refresh. **This step does not change your system**. Down-level check can help you decide or plan for an FSU. Note that this check can be done independent of the FSU itself.

Down-level check can also be made from a virtual tape. In this case, you must specify the VSAM file name.

After selecting option 1, you have to specify the **address (cuu) of the tape unit** used for the FSU.

During down-level check, the current system level is checked and compared with the refresh level. Then the system provides one or more list outputs containing all PTFs and APARs that **are not** included in the refresh level but which **are** already installed in your current system.

Note that these PTFs and APARs must be reapplied after FSU installation because the FSU removes the PTFs. Down-level check is performed against the base products residing in the following sublibraries:
- IJSYSRS.SYSLIB
- PRD1.BASE
- PRD1.MACLIB
- PRD2.SCEEBASE
- PRD2.GEN1 (if the Generation Feature is installed).
- PRD2.TCPIPC

You should not perform a down-level check when you do a Release Upgrade.

Option 2 – FSU Preparation

FSU Preparation **is performed on the old system.**

For a release upgrade you must always select *FSU Preparation* (see Figure 9) and run the resulting job immediately before the *FSU Installation* is performed.

On the panel that is displayed, specify the **address (cuu) of the tape unit** used for the FSU. If you use a virtual tape, specify the VSAM file name.

This selection creates a job stream which restores the latest level of the FSU functions from the refresh tapes. Thus you can always be sure that only the latest FSU version is used for a refresh.

The following jobs run (VSE/POWER job names are shown in parenthesis):

Step 01 (DTRSTFSU)
> Builds the Job Manager environment.

Step 02 (DTRFSU02)
> Loads the FSU program into IJSYSRS.SYSLIB.

Step 03 (DTRFSU03)
> Loads FSU skeletons into VSE/ICCF library.

Step 04 (DTRFSU04)
> Does a clean up of the preparation.

Do not run the *FSU Installation dialog* **until these jobs have completed**. FSU preparation must have finished before you can run the FSU installation job.

Option 3 – FSU Installation

Note, that FSU installation (stage 1) must immediately follow the FSU preparation job.

You can run stage 1 of the installation while the system is up and running. There is no need to shut down any partitions.

When you select option 3 *FSU Installation,* the following panel is displayed:

```
 SRV$FS03                      FAST SERVICE UPGRADE

  Enter the required data and press ENTER.

 TAPE UNIT ADDRESS............ ___          For valid addresses see HELP
 Enter 1 if FSU is to be done from virtual tape, else enter 2 for real
 tape.
 VIRTUAL TAPE................. 2            FSU from virtual (VSAM) tape?
 If virtual tape is selected, enter:
 VSAM FILE NAME.............. _____        Unique name of the VSAM file
 Enter 1 for YES and 2 for NO to the following questions.
 GEN-LIB REFRESH............. 1            Do you want to refresh Generation
                                           part?
 REORGANIZE DTSFILE.......... 2            Do you want to reorganize DTSFILE ?
 BACKUP LIBRARIES............ 1            Do you want to backup base libraries?
 BACKUP DTSFILE.............. 1            Do you want to backup the DTSFILE ?
 OWN POWER PARAMETERS........ 2            Do you have your own POWER parameters
                                           (Phase, Partition allocs, Partition
                                           size, or PFIX value) ?


  PF1=HELP      2=REDISPLAY  3=END
```

Figure 10. Panel - FSU Installation

TAPE UNIT ADDRESS
> Specify the address (cuu) of the tape unit used for FSU and for creating the backups.

VIRTUAL TAPE

Enter 1 if you want to do an FSU from virtual tape. If you select virtual tape, you must enter the VSAM file name of the file holding the tape image.

In case you specified a virtual tape, a real tape is used for reorganizing the DTSFILE as well as for creating backups of libraries and the DTSFILE. The real tape is used on the same address as the virtual tape.

GEN-LIB REFRESH

This selection appears only for those users who have installed the Generation Feature.

Specify whether you want to upgrade the generation library during FSU. If you enter **2** (NO), the entry for the generation library is removed from the system history file. The sublibrary (PRD2.GEN1) where the Generation Feature resides is not upgraded. Consequently, no service for the Generation Feature (supervisor generation macros) will be applied in future.

REORGANIZE VSE/ICCF DTSFILE

Specify whether you want to reorganize the *refreshed* VSE/ICCF DTSFILE. Reorganization of the VSE/ICCF DTSFILE helps improve system performance.

BACKUP LIBRARIES

Specify whether you want to back up the VSE system libraries IJSYSRS, PRD1, PRD2.SCEEBASE and PRD2.TCPIPC. If you do not have a backup available, specify **1** (YES) in the BACKUP field for the libraries. (Job DTRFSU12 is created.)

BACKUP VSE/ICCF DTSFILE

Specify whether you want to back up the *existing* VSE/ICCF DTSFILE. If you do not have a backup available, specify **1** (YES) in the BACKUP field for the VSE/ICCF DTSFILE. (Job DTRFSU13 is created.)

In case of an error situation, you should have a job in your VSE/POWER reader queue to restore your original VSE/ICCF DTSFILE.

OWN POWER PARAMETERS

Specify whether you want to use a VSE/POWER phase that you have generated yourself. If you specify "YES", another panel appears and prompts you for the name of the VSE/POWER phase and allocation values, such as the allocation size and the SET PFIX value for the VSE/POWER partition. You also have to enter the name of the library where your VSE/POWER phase is stored. It is recommended to make a copy of your own VSE/POWER phase.

This dialog creates a job with the default name DTRFSU. From the *Job Disposition* panel, you can submit the job to batch, file it in your default primary VSE/ICCF library, or both.

After you have entered all data, the *Job Disposition* panel is displayed, and you can submit the job. You can change the class on the *Job Disposition* panel. Note, that the startup for stage 2 of FSU activates seven partitions only (BG, F1 to F5 and FB). In stage 2 all jobs run automatically in partition BG.

Continue with "FSU Installation – Stage 1" on page 45, independent of whether you install a service refresh or perform a version upgrade.

FSU Installation – Stage 1

Stage 1 has the following job sequence. Note that the step numbers consist of a *1* (from stage *1*) and a job number (from *1* to *A*). The VSE/POWER job names are shown in parentheses:

Step 11 (DTRSTFSU)
> Builds the Job Manager environment.

Step 12 (DTRFSU12) – this step is optional
> Backup is made with the following specifications:
> - BACKUP L=IJSYSRS R=S I=HIST
> - BACKUP L=PRD1 ID=PRDBAS
> - BACKUP S=PRD2.SCEEBASE ID=PRDLE
> - BACKUP S=PRD2.TCPIPC ID=PRDTCPC

Step 13 (DTRFSU13) – this step is optional
> Backs up VSE/ICCF DTSFILE, if specified. The job asks you to disconnect the VSE/ICCF DTSFILE.
>
> Before you continue, check the list output to make sure the backup is complete.
>
> You need a scratch tape for this purpose.

Step 14 (DTRFSU14)
> Copies the system history file to a work history file. In addition, the system informs you if this fast service upgrade is a refresh or a release upgrade.
>
> **Service Refresh**

```
* --------------------------------------------------------------------
*   FSU STAGE I
*       THIS FAST SERVICE UPGRADE IS
*       A REFRESH OF Z/VSE 5.2
*    IF ABOVE UPGRADE IS NOT EXPECTED, YOU MAY HAVE MISSED TO RUN
*    THE PREPARATION STEP.
*    REPLY  "(END/ENTER)"  TO CONTINUE THE UPGRADE, OR CANCEL
*    THE JOB AND RERUN THE PREPARATION STEP.
```

> **Release Upgrade from z/VSE 4.3 or z/VSE 5.1**

```
* --------------------------------------------------------------------
*   FSU STAGE I
*       THIS FAST SERVICE UPGRADE IS
*       FROM Z/VSE 4.3 to Z/VSE 5.2
*       FROM Z/VSE 5.1 to Z/VSE 5.2
*    IF ABOVE UPGRADE IS NOT EXPECTED, YOU MAY HAVE MISSED TO RUN
*    THE PREPARATION STEP.
*    REPLY  "(END/ENTER)"  TO CONTINUE THE UPGRADE, OR CANCEL
*    THE JOB AND RERUN THE PREPARATION STEP.
```

Step 15 (DTRFSU15)

> **Service Refresh**

> 1. Installs the system library of the refresh into a system work library (IJSYSR1).
> 2. Refreshes PRD1.MACLIB.
> 3. Restores the language-dependent parts of z/VSE into IJSYSR1.SYSLIB.

Installing a Service Refresh Through FSU

Depending on your software levels, MSHP lists down-level service information. You must reapply this list of PTFs after you have finished with the FSU.

Release Upgrade from z/VSE 4.3 or z/VSE 5.1

1. Removes the old products residing in IJSYSRS.SYSLIB.
2. Refreshes IJSYSR1 (installs the system library of the refresh into a system work library IJSYSR1).
3. Refreshes PRD1.MACLIB
4. Refreshes NLS (restores the language-dependent parts of z/VSE into IJSYSR1.SYSLIB).

Step 16 (DTRFSU16) – this step is performed if the Generation Feature is installed

Refreshes the Generation Feature, if specified. This is done only if you installed the Generation Feature and answered **YES** when you were asked about a Gen-Lib refresh.

If you have installed the Generation Feature and answered **NO** to the question about a Gen-Lib refresh, the Generation Feature will be removed. This is true both in case of an FSU and of a Release Upgrade.

Note: If you are performing a Release Upgrade from z/VSE 4.3 or z/VSE 5.1 and if you answered **NO**, the old Generation Feature is removed from the History File, and PRD2.GEN1 is cleared.

Step 17 (DTRFSU17)

Copies members from PRD2.SAVE and user procedures from IJSYSRS.SYSLIB to IJSYSR**1**.SYSLIB. In addition, the members listed below are copied from IJSYSRS to IJSYSR**1**.

- STDLABEL.PROC
- STDLABUP.PROC
- STDLABUS.PROC
- DTRPOWR.PROC
- DTR$DYNn.Z
- CICSICCF.Z
- VTAMSTRT.Z

DTSECTAB.PHASE is **not** copied from IJSYSRS.SYSLIB to IJSYSR1 in order to make sure stage 2 runs without security.

Note: If this step ends with a return code 8, you can ignore the error.

Step 18 (DTRFSU18)

Service Refresh

Catalogs jobs which have to run in stage 2. Also, the VSE/POWER startup job that later is used in stage 2 is cataloged. It uses your own POWER phase and your allocation values, if specified.

Step 19 (DTRFSU19)

Does a cleanup for stage 1 and deletes all FSU jobs from stage 1. In case of problems, refer to "Tips and Hints If Errors Should Occur" on page 54.

Step 1A (DTRCLFSU)

Service Refresh
    If you have generated your own supervisor, you must reassemble
    it and catalog it into the IJSYSR1 library at this point.

    If you have your own POWER phase, you can assemble it now if
    stored in IJSYSRS.SYSLIB on DOSRES. The phase should also be
    cataloged into IJSYSR1.SYSLIB.

Release Upgrade from z/VSE 4.3 or z/VSE 5.1
    If you have your own POWER phase, FSU will use the
    IBM-provided phase IPWPOWER during FSU stage 2. You should
    reassemble the POWER phase and catalog it into PRD2.CONFIG
    (which is recommended) now, or after stage 2 has finished. If your
    private POWER phase is cataloged into IJSYSRS, you must copy it
    into IJSYSR2.SYSLIB (on DOSRES) while IPLed from SYSWK1
    (stage 2), or into IJSYSR1.SYSLIB if assembled at the end of stage 1.

Stage 1 of the FSU has finished successfully.

FSU Installation - Stage 2

Note that the IPL procedure of stage 2 brings up a system that can only be used
for FSU installation. **Stage 2 runs without security (even in a secured system)**.
This is achieved by not copying DTSECTAB.PHASE into IJSYSR1.SYSLIB. The
system issues an informational message that the security phase could not be
loaded, which is expected.

Systems with security on, should not place jobs that access secured resources, into
the RDR queue as long as security is not yet active. The reason is, that security
information (userid) is not propagated when security is not active. During FSU
stage 2 (IPLed from SYSWK1) the system is brought up without security. For all
jobs submitted at this stage security propagation does not take place. These jobs
have to be resubmitted from an administrator userid when the FSU is finished and
the system is IPLed again with security active. For example, you have to resubmit
CEEWARC after FSU. You should resubmit startup jobs for CICS or DB2 for
example, using the provided skeletons.

IPL from the refreshed SYS.NEW.RES (which is normally on SYSWK1). To begin
stage 2, you must use a JCL procedure that has been tailored especially for FSU. To
use this JCL procedure, do the following:

- IPL your system by using the load parameter on the load panel. You can also
  use the re-ipl command REIPL ccu,LOADP=..P. This is also possible if your VSE
  system runs under VM. For a detailed description of the load panel, refer to the
  chapter that describes automatic installation for native and VM users in *z/VSE
  Installation*.

VM Users

If your system is running under VM, you must specify LOADPARM L.P on either
the LOAD panel or by using the IPL command in VM.

1. **IPL cuu** (SYSWK1) **LOADPARM L.P**
2. Press **ENTER**
3. Enter **IPL=$IPLxxx,JCL=$$JCLFSU**
4. Press **ENTER**

Installing a Service Refresh Through FSU

In case your IPL procedure still uses another supervisor than $$A$SUPI, enter the
following:
```
IPL=$IPLxxx,JCL=$$JCLFSU,STOP=SUP
```

You are now prompted to modify the supervisor to $$A$SUPI.

In case of SCSI disk devices, you must enter the definitions required to access the
SCSI disk by using the SET LOADDEV command. For more details, refer to the
chapter that describes how to perform an automatic installation in *z/VSE
Installation*.

Note:

1. Consult *z/VSE Guide to System Functions* for more information about the LOAD
   parameter (LOADPARM).
2. You use your normal IPL procedure, but you **must** use the $$JCLFSU
   procedures. The VSIZE value in your IPL procedure must have at least 150 MB.

After IPL, the following tasks are performed:

Release Upgrade from z/VSE 4.3 or z/VSE 5.1

In case of a release upgrade from z/VSE 4.3, or z/VSE 5.1 the system
issues the following message:
```
1Q0HD  IF SPOOL FILE MIGRATION TO V9R2 IS INTENDED REPLY 'YES', ELSE 'NO'
```

Enter 'YES' to have the queue files converted.

Security on

In case your system is usually started with security set to 'on', the
following message is displayed:
```
1QFFD  VSE/POWER WARMSTART AND VSE ACCESS CONTROL NOT
       ACTIVATED(SEC=NO). DO YOU WISH TO CONTINUE? (YES/NO)
```

Enter 'YES' to continue. (If you enter NO, the FSU will end.)
- Load all FSU stage 2 jobs.
- Execute all FSU stage 2 jobs. All jobs run automatically in partition BG.
- Start up a mini system, which allows you to perform the necessary steps in
  stage 2 of FSU.

Stage 2 has the following job sequence. Note, that the step numbers consist of a *2*
(to indicate stage *2*) and the job numbers (*1* to *8*). In case of problems, refer to
"Tips and Hints If Errors Should Occur" on page 54.

The VSE/POWER job names are shown in parentheses:

Step 21 (DTRSTFSU)
Builds the Job Manager environment.

Step 22 (DTRFSU22)

Service Refresh

1. Selectively restores the VSE/ICCF DTSFILE. (Only the members
   serviced by IBM since the last refresh are replaced in this step.)
2. Selectively restores the language-dependent members in the
   VSE/ICCF DTSFILE. (Only the members **serviced** by IBM since
   the last refresh are replaced in this step.)

Release Upgrade from z/VSE 4.3 or z/VSE 5.1
1. Selectively restores the VSE/ICCF DTSFILE (all non-language-dependent members are restored).
2. Selectively restores all language-dependent members of the VSE/ICCF DTSFILE.

Step 23 (DTRFSU23)

Refreshes the Language Environment® (LE/VSE) code into PRD2.SCEEBASE .

Step 24A (DTRFSU4A)
Refreshes the base programs in PRD1.BASE.

Step 24B (DTRFSU4B)
Restores the online message explanation file.

Step 24C (DTRFSU4C)
Following are the steps.
1. Refreshes the text repository file.
2. Updates the CICS CSD (CICS System Definition File). Groups VSETYPE, VSETERM, VSEAI62, CEE, FCPSP, VSETERM1, and VSESPG, are upgraded. In addition, CICS Transaction Server related definitions are updated using the UPGRADE command of the DFHCSDUP utility. All updated groups remain appended to the list VSELIST. User defined groups are not affected.
3. For a version or release upgrade deletes and defines the PTF file with new recordformat, if it is in the VSESPUC catalog.

Step 24D (DTRFSU4D) – TCPIPC
Refreshes the TCP/IP code into PRD2.TCPIPC and updates the group TCPIP in the CSD file.

Step 25 (DTRFSU25) – this step is optional
Backs up and restores the entire VSE/ICCF DTSFILE. This reorganizes the DTSFILE for better VSE/ICCF performance. In case a virtual tape is used for backup, the same address is used. But it is recommended to use a real tape.

Step 26 (DTRFSU26)

Service Refresh
Copies the system work library IJSYSR1 to the system library IJSYSRS and renames system procedures.

Release Upgrade from z/VSE 4.3 or z/VSE 5.1
Personal history part is updated with new refresh level.

Step 27 (DTRFSU27)
Updates work history file locations and copies the work history file to the system history file.

Note: Code replacement is completed. The job sequence now processes startup information.

Step 28 (DTRCLFSU)
Starts cleanup job. DTSECTAB.PHASE is recataloged into IJSYSRS.SYSLIB (if appropriate).

When the last step completes, the following message is displayed:

Stage 2 of FSU is successfully finished.

After the completion message, a job step for **teleprocessing preparation** follows. The job to process the startup information does the following:

1. **It generates a** *basic startup* **for CICS**.

   Note: You can regenerate your own CICS tables and options later.

   During this step, you have to respond to certain messages.

   If you specified TCP/IP during initial installation the following VTAM definitions are optional.
   Reply Yes to the following message if you want to specify VTAM terminals.
   Message IESI0098D DO YOU WANT TO SPECIFY VTAM DEFINITIONS ? YES/NO

   You need to define the following:
   - Specify whether the local control unit is an SNA control unit (**YES**/**NO**).

     If you reply **YES**, define:
     – Control unit address (cuu)
     – Terminal type (for example, 24x80)
     – Up to three local VTAM terminals (port number).

     If you reply **NO**, define up to three local VTAM terminals (*cuu*).

2. **It starts CICS/ICCF and VTAM (for VTAM users)** in basic startup mode for test purposes.

Post-Stage 2 Processing

Once CICS/ICCF and VTAM are started, you can customize your system. Such as VSE/ICCF, VSE/POWER, CICS tables, options and all other tables stored in PRD2.CONFIG.

Do not forget to update your default selection panels and your default application profiles.

To update your selection panels, proceed as follows:

1. Sign on to the z/VSE Interactive Interface using the user ID **SYSA**.

2. To access the dialog, start with the *Function Selection* panel and select:

   2 (Resource Definition)

   1 (User Interface Tailoring)

   2 (Maintain Selection Panels)

Administrator Fast Path	Synonym
212	Default: **SPM** Yours:

3. A filter panel is shown, where you can press ENTER to get a list of all default selection panels, or enter the name or the first characters of a specific selection panel you want to list.

4. Press PF6 (System).

To update your application profiles, proceed as follows:

1. If not already done, sign on to the z/VSE Interactive Interface using the user ID **SYSA**.

2. To access the dialog, start with the *Function Selection* panel and select:

> **2** (Resource Definition)
>
> **1** (User Interface Tailoring)
>
> **3** (Maintain Application Profiles)

Administrator Fast Path	Synonym
213	Default: **APM** Yours:

3. A filter panel is shown, where you can press ENTER to get a list of all default application profiles, or enter the name or the first characters of a specific application profile you want to list.
4. Press PF6 (System).

Note: Following updates reflect changes to procedures. Any update of sublibrary IJSYSRS.SYSLIB will be lost, because the update is done on volume SYSWK1, which was already copied back to DOSRES. Any changed procedure should be copied to IJSYSR2.SYSLIB (DOSRES).

In z/VSE 5.2 TCP/IP is installed in library PRD2.TCPIPC instead of PRD1.BASE. To adapt changed TCP/IP library proceed as follows:

1. Update the following procedures and jobs using the changed skeletons from ICCF library 59.
   SKCICS
   > Change the LIBDEF chain for the CICS startup job.

   SKCICS2
   > Change the LIBDEF chain for the startup job for second CICS.

   SKLIBCHN
   > Change the LIBDEF procedure.
2. Check, if you had made own updates to the following members.
   SKTCPSTR
   > TCP/IP startup job.

   SKVTASTJ
   > Virtual tape server.

   SKVCSSTJ
   > Connector server startup job.

   If this is the case, please redo your changes using the new skeletons.

When you have finished the preceding tasks, perform the following steps:

- Shut down all partitions except for the VSE/POWER partition. Shut down CICS with **CEMT P SHUT I** to force a CICS cold start after IPL.
- Shut down the VSE/POWER partition.

IPL from DOSRES

At this stage, IPL your system from DOSRES and proceed with the steps described in the following sections.

You can also use the re-ipl command REIPL ccu,LOADP=..P.

Starting with this IPL, every change that you perform (affecting IJSYSRS.SYSLIB) goes into IJSYSRS.SYSLIB (and no longer into IJSYSR**1**.SYSLIB). The following section describes such changes.

Note: VM Users have to reinstall the VM/VSE Interface routines using skeleton SKVMVSE as described in *z/VSE Installation*.

Check the TCP/IP related jobs

After FSU has finished, please check and update the following procedures and jobs using the changed skeletons from ICCF library 59 if you use them.
The skeletons reflect the changed library where TCP/IP resides.

SKSTGDPS
> Change the LIBDEF chain to start the GDPS CLIENT.

SKSTMAS
> Change the LIBDEF chain for the SNMP Monitoring Agent.

SKSTTRAP
> Change the LIBDEF chain for the SNMP version 1 trap client.

SKTCCOON
> Change the LIBDEF chain for COBOL online programs with TCP/IP.

SKTCCOBA
> Change the LIBDEF chain for COBOL batch programs with TCP/IP.

SKTCPLBA
> Change the LIBDEF chain for PL/I batch programs with TCP/IP.

SKTCPLON
> Change the LIBDEF chain for PL/I online compile with TCP/IP.

Recataloging Procedures, Books, and Tables

Recatalog or recompile the following, using the new skeletons:

- Startup procedures.
- VTAM books. You can modify the startup book ATCSTR00 to include the parameter IOBUF31=YES, (for details refer to *z/VSE Planning*).
- CICS tables.
- LE/VSE runtime options CEEDOPT and CEEWCOPT (see library 62).
- Node error program IESZNEPX (see library 59).
- Auto install program IESZATDX (see library 59.)
- Assembly options using ASMADOPT (see library 59), if you do not use the default values. The High Level Assembler (HLASM) assembly option has been changed with z/VSE 4.1 to use workfiles as default option.
- VSE Modified Assembler Options SKASMOPT (see library 59).

To access the dialog, start with the *Function Selection* panel and select:

> **2** (Resource Definition)
>
> **4** (Hardware Configuration and IPL)
>
> **1** (Configure Hardware))

Administrator Fast Path	Synonym
241	Default: Yours:

On the *Configure Hardware* panel, press **PF5**. Select the objects you want to regenerate. The dialog then generates a job for this purpose. The dialog is described in *z/VSE SNA Networking Support*.

For user modified tables, submit your own jobs for any tables you use as you already did in your system before FSU. Note that you should review the output of the job to make sure everything is correct. Error messages in the job output might result in problems during IPL.

Create SA dump tape. Remember to create a new SA dump tape or disk with the new code. Use the dialog 461 or 462 to do so. See *z/VSE Guide for Solving Problems* for more information.

Merge Transaction Security Definitions

Starting with z/VSE 3.1.x, a new security concept has been introduced. If you have not migrated to the new security concept, it is recommended to merge the security definitions before migrating to the new concept. For further details on the new security concept, refer to *z/VSE Administration*.

If you have already migrated to the new security concept, the following step is not necessary.

Release Upgrade from z/VSE 4.3 or z/VSE 5.1

If you are still using the DTSECTXN based security concept (the old concept), do not migrate on panel TAS$SEC4. Enter 2 (no) to display panel TAS$SECF.

In order to make sure all system provided CICS transactions are defined, use the *Define Transaction Security* dialog to update the CICS transaction security definitions. For a detailed description, refer to *z/VSE Administration*.

To access the dialog, start with the *Function Selection* panel and select:
- **2** (Resource Definition)
- **8** (Security Maintenance)
- **5** (Define Transaction Security)

Press PF6 (Merge) to merge the transaction security definitions.

Secondary CICS TS

If a secondary CICS TS partition is defined and used, the CICS TS local and global catalogs must be redefined. Either skeleton SKPREPC2 in library 59 must be executed once more, or the related catalogs must be redefined and a CSD upgrade must be done by using SKCSDFC2.

Further Considerations

- Reinstalling MQSeries requires certain actions that are documented in the according PTFs. For example:
  - Customize the MQJSETUP.Z sample JCL and submit the job. Then run CICS transaction MQSU before starting MQSeries.
  - Use the new UPDATE function of MQPUTIL before starting MQSeries. Refer to the member MQDOCU.Z updated in this PTF for a description of the UPDATE function.
- Users who install a service refresh should consider reinstalling the VSE extended base programs and VSE optional programs after FSU is completed. This ensures that all licensed programs have matching service levels.
- Reapply those PTFs that you have found during down-level check.
- If you use SDAID and receive the message 4C26I THE SDAID AREA IS TO SMALL, you have to increase the SDSIZE parameter from 64K to 96K with the IPL SYS command. You can use the Tailor IPL Procedure dialog (Fast Path **242**) to increase the SYS SDSIZE parameter.
- If the PTF.FILE is not in the user catalog, please adapt it to the new record format using skeleton SKPTFILE (blocksize changed from 80 to 10320).

Tips and Hints If Errors Should Occur

The FSU dialog creates a job sequence which runs under control of Job Manager, described in *z/VSE Installation*.

If a problem occurs and you cannot resume, delete the following jobs in the VSE/POWER reader queue before resubmitting the original job from the VSE/ICCF library:

- DTRSTFSU
- DTRCLFSU
- All jobs beginning with the prefix DTRFSU.

If you do not do this, the Job Manager might not work correctly at a later time.

To temporarily stop the FSU job sequence and exit from the Job Manager you have to proceed as follows:

- Enter **x EXIT** (where *x* is the reply ID).
- Restore the generation library using the *Install Generation Feature* dialog (Fast Path **13**).
- Enter **R RDR,DTRFSUAB** to release the FSU stage 1 job again.
- Enter **x RESUME** (where *x* is the reply ID) to restart processing.

For more information on Job Manager, refer to *z/VSE Installation*. If you cannot resume, you also should consider the following items, depending on whether you exit during stage 1 or stage 2 of the FSU process:

An easy way to return the complete system to a pre-FSU level is to save the two system disks DOSRES and SYSWK1 before you start performing an FSU. *z/VSE System Utilities* describes how to do a backup of the system disks.

The following describes some general examples of problems that might occur during FSU. Remember: sometimes you might prefer to simply reset your whole system, rather than to repair it during FSU, especially if problems occur during stage 2 of FSU.

Problems During Preparation

If *FSU Preparation* is performed on a z/VSE release not supported anymore, a subsequent preparation will receive the message:

```
FSU FROM AN OLDER RELEASE IS NOT POSSIBLE.
```

This situation can be resolved submitting the following restore job, with an **installation tape of a supported release**:

```
* $$ JOB JNM=SRV$FSU,DISP=D
// JOB SRV$FSU
// ASSGN SYS005,cuu              <-- address  of tape with installation tape
// MTC REW,SYS005
// EXEC LIBR,PARM='MSHP'
RESTORE IJSYSR1.SYSLIB.SRV$FSU.PHASE:IJSYSRS.SYSLIB -
   REP=YES TAPE=SYS005
/*
/&
* $$ EOJ
```

After submitting the job you can continue the *FSU Preparation*.

Problems During Stage 1

Your system has not been changed up to the refresh of PRD1.MACLIB and the Generation Feature (for Generation Feature users only). If you exit either during or after job step DTRFSU15, you must restore the old PRD1.MACLIB. If you exit either during or after job step DTRFSU16 you must restore the old Generation library PRD2.GEN1.

Since your current system is still up and running, you can solve the problem, for example, by defining space that is necessary for restoring the Generation Feature. Retry FSU again by entering RESUME, if the job DTRFSUAB is still active. If it is not active, release the job to continue processing.

Problems During Stage 2

Depending on the failing job and on the kind of problem, the following list can be used as hints to overcome an error situation.

- If you receive the error message BG-0000 0I11D PREVIOUS COMMAND INVALID during IPL for the ADD statement (for example ADD A58:A59,TPA,05), you can do one of the following:
  - Enter an ADD statement without a mode setting or with a valid mode setting. For a list of valid mode settings refer to the ADD statement in *z/VSE System Control Statements*.
  - If you currently don'tt need the tape device, you can ignore the error, and update your Hardware Configuration using Fast Path 241 after the FSU has finished.
- If the IPL fails due to wrong specifications in the startup procedure, switch back to the pre-FSU system on DOSRES. Be especially aware of changed allocation values, which have resulted in a larger VSIZE for Environments A and B. Change the related startup procedure or job. Run stage 1 of FSU again.

  Alternatively, you can copy the modified members to IJSYSR1.SYSLIB and start stage 2 of FSU again.

  Release Upgrade:
  1. In case of a Release Upgrade from z/VSE 4.3 or z/VSE 5.1 the VSE/POWER files have already been migrated, save the contents of the migrated files using the POFFLOAD command.
  2. In case of a Release Upgrade from z/VSE 4.3 or z/VSE 5.1 with migrated power files, VSE/POWER needs a COLD start if IPL is done from DOSRES (modify startup). After COLD start, VSE/POWER queues can be reloaded using the POFFLOAD command.
- If VSE/POWER is not started, check the VSE/POWER phase and parameters. If necessary, adjust the VSE/POWER parameters via the dialog and re-run stage 1.
- If job **DTRFSU22** fails, the DTSFILE has been changed already. In this case, you must restore the original DTSFILE to reset. This is true for both steps of the job.

  You can also consider to reset to the next step (DTRFSU23) to restore the DTSFILE later on using skeleton SKICFRST available in VSE/ICCF library 59.
- If job **DTRFSU23** fails and the reason is not enough VSAM space for PRD2, the following actions might help to solve the problem:
  - Check the VSAM space requirements in the section that describes the VSE/VSAM Space Considerations for a Release Upgrade via an FSU in *z/VSE Planning*.
  - Define VSE/VSAM space for the master catalog and RESUME later.

The new facility IDCONS allows you to enter IDCAMS commands especially to define VSAM space interactively on the console.

```
// EXEC IDCONS
```

IDCONS then prompts you to define space. For example, enter the following statements:

```
DEFINE SPACE ( TRACKS (xxxx) -
ORIGIN (yyyy )          -
VOLUME(DOSRES))         -
CATALOG(VSAM.MASTER.CATALOG)
```

Important: In case you only have space on other volumes than DOSRES or SYSWK1, you additionally need to alter the related library to allow extension on the new volume (in IDCONS):

```
ALTER VSE.PRD2.LIBRARY.DATA ADDVOLUMES(SYSWK2) -
CATALOG(VSAM.MASTER.CATALOG)
```

– For other reasons than space, and if a resume is not possible, restore the system sublibrary PRD2.SCEEBASE to reestablish the current level of z/VSE's base programs and LE/VSE code.

- If job **DTRFSU4A** fails due to lack of VSAM storage, proceed as described for job DTRFSU23. For other reasons, restore system sublibrary PRD1.BASE to reestablish the current level of the z/VSE base programs. Do not try to startup VTAM or CICS from DOSRES with the new code in PRD1.BASE installed.

- If job **DTRFSU4A** fails due to insufficient storage in the partition (message M0631) and your history file is highly fragmented, defragment the history file (Fast Path 147).

- If job **DTRFSU4B** fails, you might not have specified enough space for the online message file. In this case, you must define additional space for the VSE/VSAM master catalog (IJSYSCT) and proceed as described for job DTRFSU23.

- If job **DTRFSU4C** fails, the Text Repository file might not have been loaded completely. You might have to define additional space for the VSE/VSAM user catalog VSESPUC. Check the job output and try to RESUME.

  Important: A general problem of job DTRFSU4C might be space for the Text Repository workfile. Space must be defined as specified in the dialog *Verify Location of Serviced Files* (Fast Path 141). If no space is available on the volume specified, resetting to the next step is the only way out.

  If the CSD file migration fails, you can reset the job sequence to the next job to proceed with TCP/IP. The CSD file can be migrated later. A possible reason for a job failure could be an old level (CICS TS 1.1.0) of the compiled phase DFHFCTSP in PRD2.CONFIG.

- If job **DTRFSU4D** fails due to insufficient VSAM space, you should carry out the actions as described for job DTRFSU23 and try to RESUME. For other reasons, reset to the next job and install TCP/IP later. The next job is either DTRFSU25 if reorganization of DTSFILE is specified, or DTRFSU26.

- If job **DTRFSU25** fails, you can reset to DTRFSU26 (in case the backup failed) or try to restore the DTSFILE once more after analyzing and repairing the problem.

- If job **DTRFSU26** fails, check the job output and try to RESUME.

- If job **DTRFSU27** fails, check the job output and try to RESUME. If any product information is missing in the work history file, analyze the problem and restart with stage 1 of FSU.

- If job **DTRFSU28** fails, check the job output and try to RESUME. If only the basic startup of VTAM and CICS fails, perform an IPL from DOSRES. You can repair the basic startup later. The tasks described before the heading "Post-Stage 2 Processing" on page 50 need to be performed immediately after CICS is up.

Problems related to TCP/IP

In z/VSE 5.2 TCP/IP is installed in library PRD2.TCPIPC instead of PRD1.BASE. This might cause problems if the LIBDEF chain does not contain PRD2.TCPIPC. In this case you can

- Check, if a wrong job is in the RDR queue , and you can submit the correct job.
- Check, if you have cataloged a correct job and load it via DTRIINIT.
- Use the appropriate skeleton in ICCF library 59 to submit a correct job.

Chapter 6. Using System History File and Nickname Dialogs

This chapter provides information on service dialogs that help you organize your system before or after you have applied service to it.

Selection 4 – Retrace History File

The *Retrace History File* dialog prints selected information from the system history file.

To access the dialog, start with the *Function Selection* panel and select:

1 (Installation)

4 (IBM Service)

4 (Retrace History File)

Administrator Fast Path	Synonym
144	Default: Yours:

This displays a panel with eight options. Select the type of information you want printed. Note that the header of each retrace list shows the z/VSE refresh level.

```
 IESADMSL.IESERHFS          RETRACE HISTORY FILE
                                              APPLID: DBDCCICS
   Enter the number of your selection and press the ENTER key:

        1  Retrace
        2  Retrace Products
        3  Retrace Components
        4  Retrace PTFs
        5  Retrace APARs
        6  Retrace Members
        7  Retrace Component ID
        8  Lookup PTF/APAR




 PF1=HELP          3=END       4=RETURN            6=ESCAPE(U)
                   9=ESCAPE(m)

 ==> _                                        PATH: 144
```

Figure 11. Panel - Retrace History File

1. Retrace

Administrator Fast Path	Synonym
1441	Default: Yours:

This dialog creates a job to print the following information from the system history file:

- A list of all products and components installed

- A sorted list of all local fixes and applied PTFs
- An APAR cross-reference list
- A member cross-reference list
- Comments from any RESOLVES statements

2. Retrace Products

Administrator Fast Path	Synonym
1442	Default: Yours:

The dialog creates a job to print the following information for each product installed in the system:

- Installation date
- Components within the product
- Comments, if any
- Residence of the product

3. Retrace Components

Administrator Fast Path	Synonym
1443	Default: Yours:

This dialog creates a job to print information for all installed components:

- Component ID
- Release level
- Installation date
- List of PTFs and APARs applied
- All generated members
- Residence of the components

4. Retrace PTFs

Administrator Fast Path	Synonym
1444	Default: Yours:

This dialog creates a job to print all applied PTFs by sequence number. For each PTF number, the list output provides the following information:

- PTF number
- Indication if PTF was revoked
- Component to which PTF applies
- Affected modules
- Resolved APARs
- Prerequisites and corequisites
- Negative prerequisites
- PTFs which supersede the PTF

- PTFs which are superseded by the PTF

5. Retrace APARs

Administrator Fast Path	Synonym
1445	Default: Yours:

This dialog creates a job to print all APARs which have been fixed by either a PTF or APAR fix. For each APAR number, the list output provides the following information:
- APAR number
- Affected component
- PTF number (for PTF fix)
- Application date
- Affected modules (for APAR fix)

6. Retrace Members

Administrator Fast Path	Synonym
1446	Default: Yours:

This dialog creates a job to print all phases, relocatable modules, and macros which are affected by a PTF or an APAR fix. The list output provides the following information:
- Module name
- Component to which a module belongs
- Application date of PTF/APAR fix
- PTF/APAR number affecting a module

7. Retrace Component ID

Administrator Fast Path	Synonym
1447	Default: Yours:

This dialog creates a job to print information about selected components. You can print information about one or more components by entering

    1 (Retrace Component)

in the option column next to the component that you want to retrace.

The list output provides the following information for each selected component:
- Component ID
- Release level
- Installation date
- List of PTFs and APARs applied
- All generated members

8. Lookup PTF/APAR

Administrator Fast Path	Synonym
1448	Default: Yours:

This dialog allows you to find out interactively whether a PTF or APAR is installed. In the following panel (*Retrace History File* - List of Components) you must enter

1 (Lookup PTF/APAR)

in the option column next to the component for which you wish to retrace the PTFs/APARs.

The next panel lets you enter up to 10 PTF/APAR numbers. Type in the PTF/APAR number(s) in the provided fields and then press ENTER.

These are possible results:

```
PTF/APAR   |--  RESULT  ---------------------------|

UD12345    NOT APPLIED FOR THIS COMPONENT
UD23456    LOCAL FIX
DY34567    FIXED IN PTF UP12345
DY12345    APPLIED
```

Note that you have two choices:

1. If you enter:

   1 (YES)
   in the *EXTENDED SEARCH* field on the panel on which you specify the PTFs/APARs,

   the system scans the whole system history file and not only the history of the component you specified. If the system retraces the specified PTF/APAR in a component other than the one you specified, the component ID displays on the result panel.

2. If you enter

   2 (NO)
   in the *EXTENDED SEARCH* field,

   the system restricts the scope of the retrace to only one component.

Selection 5 – Personalize History File

The *Personalize History File* dialog updates personalized information for the system history file. You first entered MSHP information when you completed the initial installation of z/VSE. You should use this dialog whenever you want to update the information.

To access the dialog, start with the *Function Selection* panel and select:

1 (Installation)
4 (IBM Service)
5 (Personalize History File)

Administrator Fast Path	Synonym
145	Default: Yours:

In this dialog you can enter data that is unique to your installation (see the panel shown in Figure 12).

Type in your name, address and telephone number in the required fields. In the last field, type in the name of the person who is responsible for maintaining the system. This is usually the system administrator. After entering all required information, press **ENTER**.

```
ADM$LB1                    PERSONALIZE HISTORY FILE

Enter the required data and press ENTER.

This panel accepts information for your system history file.
This information is placed in the general header record of the
history file.


CUSTOMER NAME..... _____

ADDRESS........... _____

PHONE NUMBER...... _____

PROGRAMMER NAME... _____



PF1=HELP      2=REDISPLAY  3=END
```

Figure 12. Panel - Personalize system history file

Selection 6 – Change Nicknames

This function allows you to change the nicknames displayed on the panels related to products and components.

To access the dialog, start with the *Function Selection* panel and select:
- **1** (Installation)
- **4** (IBM Service)
- **6** (Change Nicknames)

Administrator Fast Path	Synonym
146	Default: Yours:

You will now get a panel of all products that are in the system. This list includes:
- The nickname of the product, for example *EREP*
- The product ID, for example *260E00*

Note that the system is shipped with an initial setting for the nicknames. Table 6 on page 64 shows you a list of the predefined nicknames that z/VSE offers. The service dialogs use these nicknames for the licensed programs included in z/VSE.

Now you can do one of the following:
- Select **option 2** for a particular product.

Changing Nicknames

> This option displays another FULIST panel, showing a component or a list of components of one product with the nickname(s) of the component(s) and component ID(s).
>
> – Type **3** in the option column if you want to change a component nickname.
> – In the *Change Component Nickname* panel, enter the new nickname that identifies the component in the future. The nickname can have 1 to 8 characters.

- Select **option 3** for a particular product.

  This option lets you change the nickname of a product.

  – In the *Change Product Nickname* panel, enter the new nickname that identifies the product in the future. The nickname can have 1 to 8 characters.

Table 6. System Nicknames Used by Service Dialogs

Product/Component	NICKNAME
Device Support Facilities	DSF/VSE
CICS TS	CICS/TS
DITTO/ESA for VSE	DITTO
VTAM	VTAM
High Level Assembler for VSE	HLASM
EREP	EREP
TCP/IP	TCP/IP
OSA/SF	OSA/SF
LE/VSE Runtime Base Code	LE/BASE
LE/VSE Cobol Runtime	LE/COB
LE/VSE PL/I Runtime	LE/PLI
LE/VSE C Base	LE/C

Selection 7 – Defragmentation of History File

This option allows you to defragment your History File.

To access the dialog, start with the *Function Selection* panel and select:

1 (Installation)

4 (IBM Service)

7 (Defragmentation of History File)

Administrator Fast Path	Synonym
147	Default: Yours:

```
 SUB$PRO5                      JOB DISPOSITION

 Enter the required data and press ENTER.


JOB DESTINATION.............. 2            Enter 1 to submit the job to batch.
                                          Enter 2 to file in library.
                                          Enter 3 to do both.
JOB NAME..................... DEFRAG       The name under which the job will be
                                          saved in VSE/ICCF.
PRIORITY..................... 3            Priority 0-9 for this job.
CLASS........................ 0            Changing * has no effect.
DISPOSITION.................. D            D,H,K or L. Changing * has no effect.
JOB ACCOUNTING............... _____
HOLD LIST IN QUEUE........... 2            Enter 1 to hold output in list queue.
                                          Enter 2 to print output immediately
TIME EVENT SCHEDULING........ 2            Enter 1 if TIME EVENT SCHEDULING
                                          required, otherwise enter 2.
OTHER PARAMETERS............. 2            Enter 1 to change any other POWER JOB
                                          parameters, otherwise enter 2.

 PF1=HELP      2=REDISPLAY  3=END
```

Figure 13. Panel - Defragmentation of system history file

This option copies the history file to the work history file and merges it back into an empty system history file. This merge performs a defragmentation of the file.

Chapter 7. Installing Service Changes (without Dialogs)

It is recommended to use the dialogs provided by the Interactive Interface for installing service changes to z/VSE. When you are a workstation user, use the VSE Workdesk windows to install service changes to z/VSE. If you cannot use the dialogs for some reason, this chapter shows selected example jobs for installing the service changes without them.

Note: The product IDs, phase and macro names used in the following examples do **not reflect a system environment**. They are used for demonstration purposes only.

The z/VSE Interactive Interface dialogs keep their own information, therefore the changes that you make using MSHP might not be reflected in the dialogs.

IBM distributes service changes for z/VSE either of the following:
- One or a number of Program Temporary Fixes (PTFs) on tape (or cartridge).
- An Authorized Program Analysis Report (APAR) fix.

A PTF contains one or more phases, modules, or macros, which replace a corresponding existing phase, module, or macro. An APAR is an update to a phase, a module, or a macro. How to handle PTFs and APARs is summarized below.

- **Handling PTFs**

  PTFs are shipped on a distribution tape or can be downloaded from the Internet (ShopzSeries).

  The electronic shipment contains the following:
  1. Readme file
  2. PTF cover letter
  3. PTFs

  The cover letter is in text format. The PTF files are in binary format. Refer to Appendix C, "Downloading PTFs from the Internet," on page 87 for details.

  The tape (also called service tape) contains the following, in the sequence listed:
  1. A tape history file or a null file
  2. A null file
  3. A null file
  4. An EXCLUDE list or a null file
  5. PTF cover letters or a null file
  6. PTFs
  7. A null file.

  Note: MSHP can install PTFs from a service tape only if the PTFs are stored on the tape as the sixth file with a block size of 10320 bytes.

  When you receive a service tape:

  1. Have MSHP produce a listing of the PTF cover letter file, and read this printout carefully. This helps you in planning and performing the task of installing required PTFs.
  2. Install those PTFs which correct system problems. For preventive service, you can install all PTFs supplied by IBM if this is desirable.

The service tape as shipped by IBM might not include PTFs whose installation on your system is a **prerequisite** for one or more of the PTFs you want to install. In this case, MSHP does not install the affected PTF and informs you by a message.

If you know that prerequisite PTFs exist on another service tape and that these PTFs are not yet installed, have MSHP INCLUDE the additional tape(s) for the prerequisite PTFs in the installation step.

You might also receive a single PTF from IBM. You get this PTF in the form of an executable job which invokes MSHP.

- **Handling APARs or Local Fixes**

  Normally, such fixes are not distributed in machine readable form. You install them using the CORRECT function of MSHP.

- **Revokable or Irrevocable Installation**

  The REVOKABLE option of MSHP produces so called **backout jobs** of PTFs that you install. Use the option only if you install just one or a few PTFs for the purpose of solving a specific problem.

  MSHP writes backout jobs onto the tape mounted on the drive to which SYS004 is assigned. You can use this tape as input to MSHP should you need to back out a PTF. Installing a backout PTF amounts to a re-installation of the library member(s) replaced by installation of the PTF.

  Note: Do not install the backout job for a PTF that is a pre- or corequisite for other PTFs or has comparable APAR/Local Fix dependencies.

  This chapter gives sample jobs for both the installation of PTFs and the installation of backout jobs.

- **Preparing the System**

  Before the installation of service changes, you should produce a backup on tape of the sublibrary (or program) on which you want to install a service change. Perform a Librarian BACKUP run to get a backup of the system library. When you have one program stored in one sublibrary, perform an MSHP BACKUP run to get a backup of an individual program. If several programs are in one sublibrary, you get a backup of the sublibrary.

  There is no need to define the libraries or sublibraries that hold the products which are to be serviced. MSHP establishes required search chains based on the information recorded in the history file. It uses the services of the Librarian to actually delete replaced members and to catalog new, replacement members.

Service-Related Activities

This chapter includes a discussion of activities that IBM recommends after successful completion of a service run. It gives sample jobs for history-related activities for which there occasionally might be a need at your location.

Sample Jobs

Where applicable, the sample jobs in this chapter include statements for running the installation jobs under VSE/POWER. If your service partition is not under VSE/POWER control when you run these jobs, the system ignores those statements.

Handling PTFs

This activity includes the installation of PTFs and the backout of installed PTFs, if this is necessary.

Installing PTFs from a Service Tape

For the installation of PTFs, IBM recommends that you use the z/VSE dialogs. If you do not want to use the dialogs for some reason, proceed as follows:

1. **List the PTF cover letter(s) from tape or from disk**

   Submit a job similar to the ones shown in Figure 14 and Figure 15. The jobs produce, on the device assigned to SYSLST, a printout of the content of the PTF files requested by the MSHP LIST statement.

```
          * $$ JOB JNM=LSTSVCE,CLASS=0
          // JOB   LIST SERVICE INFORMATION
     (a)  // ASSGN SYS006,cuu
          // EXEC  MSHP
     (b)  LIST SERVICETAPE -
               NODOCUMENT COVER SEPARATE
     (c)  PTF=(UD12345,UD45678,...)
          /*
          /&
          * $$ EOJ
```

Figure 14. Sample Job for Listing the PTF Cover Letter(s) from Tape

```
          * $$ JOB JNM=DOCPRINT,DISP=D,PRI=3,                              C
          * $$ NTFY=YES,                                                   C
          * $$ LDEST=*,                                                    C
          * $$ CLASS=0
          // JOB DOCPRINT  PRINT SERVICE FILE DOCUMENTATION
                    .
                    .
                    .
          // EXEC MSHP
     (d)  LIST SERVICETAPE FROMDISK XREF,COVER,CONT
          /*
          /&
          * $$ EOJ
```

Figure 15. Sample Job for Listing the PTF Cover Letter(s) from Disk

```
          * $$ JOB JNM=DOCPRINT,DISP=D,PRI=3,                              C
          * $$ NTFY=YES,                                                   C
          * $$ LDEST=*,                                                    C
          * $$ CLASS=0
          // JOB DOCPRINT  PRINT SERVICE TAPE DOCUMENTATION
          *    PTF COVER LETTERS ACTION
          // PAUSE  MOUNT THE SERVICE TAPE ON TAPE DRIVE 181.
          // ASSGN SYS006,181
          // MTC REW,SYS006
          // EXEC MSHP
     (b)    LIST SERVICETAPE NODOC,NOXREF,COVER,CONT,ACTION
           /*
           /*
           /&
           * $$ EOJ
```

Figure 16. Sample Job for Printing the ACTION Information from Tape

Note:

a. Assign SYS006 to the tape drive on which you mounted the service tape.

b. The meanings of the command parameters are:

NODOCUMENT
Suppresses a printout of the service document.

COVER
Produces a printout of PTF cover letters

SEPARATE
Causes a new page to be started for each PTF cover letter that is to be printed.

ACTION
Prints only the ACTION information that is listed in the cover letter.

c. This detail control statement causes MSHP to print only the cover letters of those PTFs that are specified. If you omit this statement, MSHP prints all PTF cover letters.

d. If the LIST function is used from disk, file IJSYSPF is used. XREF means the cross references should be printed, CONT means printing should be contiguous.

2. **Examine the PTF cover letter(s)**

The information contained in the printed list helps you decide which of the PTFs on the tape are to be installed.

3. **Set up the installation job**

You might want to install only those PTFs which correct a problem or prevent a potential problem. In that case, you should prepare a list of the PTFs you want to be included (using INCLUDE statements) or excluded (using EXCLUDE statements).

Write INCLUDE statements in the form

```
INCLUDE PTF=(UDnnnnn,UDnnnnn,...)
```

Write EXCLUDE statements in the form

```
EXCLUDE PTF=(UDnnnnn,UDnnnnn,...)
```

To install **all** PTFs on the service tape, do not supply any INCLUDE or EXCLUDE statements.

4. **Run the installation job**

To install the required PTFs, set up and submit a job similar to the ones shown in Figure 17 and Figure 18 on page 71. The samples assume that the product sublibrary must be available and accessible.

```
     * $$ JOB JNM=INSTSVE,CLASS=0
     // JOB   INSTALL SERVICE
 (1) // ASSGN SYS006,cuu
     // EXEC  MSHP
 (2) INSTALL SERVICE TAPES=2
 (3) INCLUDE PTF=(UD12345,UD45678,...)
     /*
     /&
     * $$ EOJ
```

Figure 17. Sample Job for Installing PTFs from Tape

```
// DLBL IJSYSHF,'WORK.HIST.FILE'
// EXTENT SYS018,SYSWK1,1,0,960,75
// ASSGN SYS018,DISK,VOL=SYSWK1,SHR
// DLBL IJSYSPF,'PTF.FILE',,VSAM,CAT=VSESPUC
*        ---- APPLY SERVICE ----
// EXEC MSHP
INSTALL SERVICE FROMDISK
/*
```

Figure 18. Sample Job for Installing PTFs from Disk

Note:

1. Assign SYS006 to the tape drive on which you mounted the service tape.

2. You can supply the statement with REVOKABLE specified (for example, INSTALL SERVICE REVOKABLE). Do this only if you install just one or a few PTFs and you are sure that none of the PTFs being installed is a requirement for other PTFs or has APAR/Local Fix dependencies. An attempt to revoke a PTF with such dependencies might result in a downleveled (inoperative) system.

   Specifying REVOKABLE requires that you mount an extra tape to which MSHP can write the created backout job(s). Assign SYS004 to the tape drive you use.

   TAPES=2 indicates to MSHP that it is to scan and process two service tapes. You can specify up to nine service tapes. The tapes are mounted one after the other, in response to MSHP's mount request, on the tape drive assigned to SYS006. For more than three PTF tapes, refer to *z/VSE System Control Statements* for instructions on how you should mount these tapes.

3. PTFs defined here are installed by MSHP. All other PTFs stored on the service tape are not installed. If you were using an EXCLUDE statement, MSHP would install all PTFs not defined in the statement.

Restarting a PTF Installation Run

The installation of PTFs might require modules to be link-edited into phases. Before this link-editing under MSHP control starts, MSHP takes a checkpoint. Should this link-editing fail, then MSHP terminates PTF installation, but allows you to set up the installation job again at the recorded checkpoint. To restart the installation process at this checkpoint, submit a job similar to the one shown below:

```
* $$ JOB JNM=RSTRTSV,CLASS=0
// JOB   INSTALL SERVICE
// EXEC  MSHP
INSTALL SERVICE RESTART
/*
/&
* $$ EOJ
```

Figure 19. Sample Job for Restarting a PTF Installation Run

For a restart, MSHP needs no input other than the INSTALL statement as shown in the above sample job.

Installing a Backout PTF

When a PTF is applied with the REVOKABLE option, a backout PTF is generated on tape. This backout PTF can be used to reestablish the original code by recataloging the phases that were replaced by the former PTF. When the backout PTF is written to tape, one PTF per component is created.

To install a backout PTF, you proceed in nearly the same way as for the installation of a PTF from service tape:

1. Mount the MSHP created backout tape.

2. Submit a job similar to the one shown in Figure 20.

For the MSHP job to complete successfully, it is necessary that:

- The product sublibrary must be available and accessible.

```
      * $$ JOB JNM=INSTBKO,CLASS=0
      // JOB    INSTALL BACKOUT PTF
 (1)  // ASSGN  SYS006,cuu
      // EXEC   MSHP
      INSTALL BACKOUT
 (2)  INCLUDE PTF=UD12345
      /*
      /&
      * $$ EOJ
```

Figure 20. Sample Job for Installing a Backout PTF

Note:

1. Assign SYS006 to the tape drive on which you mounted the backout tape.

2. MSHP installs the backout PTF that corresponds to the PTF whose number you specify. Any other backout PTF stored on the backout tape is not installed. If you use an EXCLUDE statement, MSHP installs any backout PTF whose number is not specified in the statement.

Handling APARs and Local Fixes

An **APAR** is provided by IBM with an **official APAR number**.

A **local fix** can be provided by IBM or it has been developed locally. The local fix number can be **chosen freely** according to the APAR naming conventions. MSHP records the installation of any fix in the system history file. Later on, you might want to install a PTF that resolves an APAR fix. In this case, the PTF installs correctly.

If you have installed a locally developed fix or if you install a PTF that does not resolve an applied APAR fix, remove the fix by using MSHP's UNDO function **before** you install the PTF. You can do this if you installed the fix with REVOKABLE specified either explicitly or by default. Reinstall and adapt the fix after having installed the PTF, should this be necessary.

If you install a fix revokable, MSHP also does the following in addition:

- For phases and modules, it records the old and the new data.

- For macros, it writes the affected macros to SYSPCH before actually altering them.

When CORRECT request is specified for a macro, also specify the TYPE operand in the AFFECTS statement. The default of TYPE=E only applies to some products still containing E-type macros, handled by the former basic Assembler language, which is not available anymore since VSE/ESA 2.1. CORRECT for E-type macros will therefore not work, changes can only be made by first de-editing the macros using ESERV, catalog as A-macro and then changing the A-macro.

For further information about applying APARs and local fixes, refer to Appendix B, "Skeletons for Installing APARs/Local Fixes," on page 81.

Service-Run Complete Activities

When you have finished installing a PTF or an APAR/Local Fix, then:

1. Create a new backup tape for each of the sublibraries that have been changed.

2. Obtain a listing of the system history file. Submit a job similar to this sample job:

```
* $$ JOB JNM=LSTHIST, CLASS=0
// JOB RETRACE
// EXEC MSHP
RETRACE
/*
/&
* $$ EOJ
```

Figure 21. Sample Job for Service-Run Complete

History-File Related Service Activities

This section describes how to:

- Archive the update of a library member in the system history file.
- Handle a history-file-full situation.
- Record the (new) residence of a previously installed system component.

Archiving an Update in the History File

The need for this might arise if, because of unusual circumstances, you change a member in the system sublibrary without using MSHP. At a later point in time, this change might have become an APAR fix which MSHP requires for the installation of a PTF. MSHP cannot install this PTF until you have recorded your change as the installation of the required APAR fix.

To record the change in your system's history file, run a job similar to this one:

```
* $$ JOB JNM=ARCHUPD,CLASS=0
// JOB   ARCHIVE UPDATE
// EXEC  MSHP
ARCHIVE  1111-222-33-444 APAR=DY12345
AFFECTS  MODULE=IJWCCDZ
/*
/&
* $$ EOJ
```

Figure 22. Sample Job for Archiving an Update

The above sample job records, in the system history file, a change by APAR DY12345. It records this change for module IJWCCDZ of the component 1111-222-33 on level 444.

Handling a History-File-Full Situation

It might happen that your system's history file becomes full during installation of service changes; MSHP indicates this by a message. To recover, run a job like the one in Figure 23 on page 74. Run the job in your system's background partition.

```
        * $$ JOB JNM=CRTEHST,CLASS=0
        // JOB CREATE NEW HISTORY FILE
        // EXEC  MSHP
        CREATE HISTORY AUXILIARY
 (1) DEFINE HISTORY AUX EXT=xx:yy
        COPY HISTORY SYSTEM AUXILIARY
        /*
 (2) ***** Making the New History File Accessible *****
        // OPTION STDLABEL=DELETE
        IJSYSHF
        /*
        // OPTION STDLABEL=ADD
        // DLBL IJSYSHF,'VSE.SYSTEM.HISTORY.FILE',99/365
        // EXTENT ,vol-id,,,xx,yy
        /*
        /&
        * $$ EOJ
```

Figure 23. Sample Job for Handling a History-File-Full Situation

Note:

1. With this command, you define an area, on the same volume, where MSHP is to copy the contents of the full history file.

   Replace xx with the begin of the extent. Replace yy with the size of blocks/tracks. Make sure you define the area yy significantly larger than the history file that has become full.

2. The remaining statements up to (but not including) /& make the new history file accessible by MSHP. Omit these statements if at your location you do not have permanently stored label information for the system history file. You might then have to change cataloged procedures instead.

To make the new information available for the Interactive Interface: Run the dialog *Verifying Location of Involved Serviced Files* (Fast Path 141) with the new extents. (Refer also to "Selection 1 – Verify Location of Involved Serviced Files" on page 13). To make the new information available after the next IPL, you must update the STDLABEL.PROC.

Changing the Residence of a Program

If you move a program (or a system component) to another sublibrary, this change must be recorded in the system history file. To have MSHP record the changed residence, use a job similar to this one:

```
        * $$ JOB JNM=RECRES,CLASS=0
        // JOB    RECORD NEW RESIDENCE
        // EXEC   MSHP
        RESIDENCE PRODUCT=222444 -
                PRODUCTION=DSFSLIB.PROD -
                GENERATION=DSFSLIB.GENE
        /*
        /&
        * $$ EOJ
```

Figure 24. Sample Job for Changing the Residence of a Program

The above sample job assumes that the program has been moved to a different sublibrary.

Note that most products do not have a GENERATION part. The RESIDENCE statement for such a product only has a PRODUCTION=xxx part.

Applying Service to VSE Programs that Were Installed Without Using Dialogs

You can install VSE programs without using the Interactive Interface dialogs. But to **use the dialogs later** to apply maintenance to these products, you first need to run program **DTRIPST**.

DTRIPST reads the system history file and updates member **DTRIHIST.Z** in **IJSYSRS.SYSLIB.**

DTRIHIST.Z refreshes member **ADM$SHIP** in VSE/ICCF library 50 whenever a dialog that needs this table is used. ADM$SHIP must contain entries for all the components or products that have been installed. If this is not the case, the maintenance dialogs issue the message:

```
'PRODUCT NOT INSTALLED'
```

DTRIPST is a phase. All you have to do to run this program is to release a pause job and start the DTRIPST program:

R RDR,PAUSEBG

(press **END/ENTER**)

0   EXEC DTRIPST

When the program is ready, enter

0

to continue.

Appendix A. Fixing Programs Not Built in MSHP Format

Some programs (products) are not built in MSHP format. Most of these programs come from software manufacturers other than IBM.

Programs that are not in MSHP format do not have any history file information delivered with the program code. In the following text, such programs generally are referred to as *non-IBM programs.*

There are two ways to fix such programs:
- By using the skeleton SKARCHIV provided in VSE/ICCF library 59 to archive a dummy entry in the system history file, and then applying the local fix.
- By using the PATCH function of MSHP, interactively from the system console or via input from SYSIPT. For details, refer to "Using the PATCH Function" on page 78 and to the example in "The PATCH Statement" on page 79.

Using Skeleton SKARCHIV

To correct a non-IBM program using the skeleton, do the following:

1. In the history file, archive a dummy entry for each library where you have installed a non-IBM program in the history file. After the product is defined to MSHP using SKARCHIV, you can service it using MSHP type fixes like PTFs or local fixes as described in Appendix B, "Skeletons for Installing APARs/Local Fixes," on page 81.

The skeleton shown in Figure 25 shows SKARCHIV, the skeleton you use to archive the dummy entry.

```
* $$ JOB JNM=ARCHIVE,CLASS=0,DISP=D,PRI=8,LDEST=*
// JOB ARCHIVE
*
* THIS JOB CREATES AN USER PRODUCT-ID AND COMPONENT-ID ENTRY
* INTO SYSTEM HISTORY FILE
*
// EXEC MSHP
ARCHIVE zapuse[1]
/*  USER PRODUCT-ID ENTRY  */
COMPRISES 9999-zap-it[1]
RESOLVE 'ENTRY FOR USER/OEM PROGRAM CORRECTIONS'
ARCHIVE 9999-zap-it-use[1]
/*  USER COMPONENT-ID ENTRY  */
RESIDENCE PRODUCT=zapuse[1]
PRODUCTION=prd2.user1[1]
/*
// EXEC DTRIPST,SIZE=500K
/&
* $$ EOJ
```

Figure 25. z/VSE Skeleton SKARCHIV

ARCHIVE
This statement is used to create entries in the system history file.

---

1. 'zapuse', '9999-zap-it', '9999-zap-it-use' and 'prd2.user1' are values which you must change.

COMPRISES

This statement is used to specify the component(s) comprised in the shipped product and the library members that make up the component(s). The information is entered in the system history file. A separate COMPRISES statement must be entered for each component contained in the shipped product.

RESOLVE

This statement associates a comment with a product, a PTF, an APAR/Local Fix, or a generated member. It is also used to indicate which APARs are fixed by a PTF.

RESIDENCE

This statement defines the names of the production and generation sublibraries in which the named product reside. This information is recorded in the history file for the service application.

In Figure 25 on page 77, '9999-zap-it' is the complete component ID. It always has the form xxxx-yyy-zz. Note that the first ARCHIVE statement always uses yyy plus the component level code (CLC), which is in this case *use*. The COMPRISES statement uses the complete component ID xxxx-yyy-zz. The second ARCHIVE statement uses the complete component ID xxxx-yyy-zz, plus the component level code. The RESIDENCE statement uses yyy, plus the component level code.

Note: You must run the ARCHIVE job once for **every** non-IBM program for which you want to apply a fix. Use the CORRECT and AFFECTS PHASE commands. Change the product ID each time.

Using the PATCH Function

This function may be useful if your task involves *application programming* and you want to do hands-on debugging. The function allows you to change (patch) a phase stored in a sublibrary. You can use the function from the system console or via input from SYSIPT.

A phase that you patch may or may not be under control of MSHP. In either case, MSHP does not record the change in your system's history. However, MSHP writes a warning message to the system console for MSHP-controlled phases.

Note: It is not recommended to use the PATCH function for MSHP-controlled members.

Using the Function from the Console

An example of using the PATCH function from the console is below. Assume that:
1. A phase named PAYPRNT is to be changed at location X'0404' (relative to the beginning of the phase).
2. The phase is stored in sublibrary PAYAPPL of library WEEKLY.

Figure 26 on page 79 shows how the patch operation is started and controlled from the system's console. Note that this is **just an example. You may get some other messages**. In Figure 26 on page 79:
- Operator-submitted data is indicated by an arrow ($\Rightarrow$) to the left of that data.
- Numbers within parentheses in the left hand margin refer to explanations in the lower half of Figure 26 on page 79.
- Pressing **END/ENTER** is indicated by a square bullet (▪).

The PATCH Statement

To patch a phase, you use the function control statement **PATCH**. *z/VSE System Control Statements* describes this statement in detail.

The following related detail control statements are:

Required:
> AFFECTS (the member to be changed)
>
> The AFFECTS statement must precede the other optional detail control statements.

Optional:
> ALTER (identifies the change to be made) SCAN (used to view portion of the members)

In the following example, pressing END/ENTER is indicated by a square bullet (▪).

```
      ⇒ 0 // job patch a phase  ▪
      ⇒ 0 // exec mshp  ▪
        BG 000 M005D ENTER CONTROL STATEMENT, OR PRESS END/ENTER TO QUIT
      ⇒ 0 patch sublib=weekly.payappl  ▪
        BG 000 M015D ENTER DETAIL STATEMENT OR "?"
      ⇒ 0 affects phase=payprnt  ▪
        BG 000 M105I UPDATE OF PHASE   PAYPRNT   IN PROGRESS
        BG 000 M015D ENTER DETAIL STATEMENTS OR "?"
(1)  ⇒ 0 ?  ▪
(2)     BG 000 M017I LIST OF ALLOWED DETAIL STATEMENTS:
                    ALTER (AL)      - IDENTIFIES MODIFICATION TO BE MADE
                    SCAN (SC)       - IDENTIFIES SCANNING TO BE PERFORMED
                    CANCEL (CANCEL) - TO QUIT
        BG 000 M015D ENTER DETAIL STATEMENTS OR "?"
(3)  ⇒ 0 scan 03fc  ▪
        BG 000 M145I 2034D20110003002FFFF40C18B5040C1   '..K...... A.. A+'
        BG 000 M015D ENTER DETAIL STATEMENTS OR "?"
(4)  ⇒ 0 scan arg=ffff  ▪
        BG 000 M144I SCAN CONTINUES FROM OFFSET 0003FC
        BG 000 M147I SCAN DATA:  FOUND AT OFFSET = 000404
        BG 000 M145I FFFF40C18B5040C14E58F0C1624110C1   '.. A.. A+.0A...A'
(5)  ⇒ 0 alter 0404 ffff:05ef  ▪
(6)     BG 000 M138I CHANGE ADDRESS: 000404
(6)     BG 000 M139I OLD DATA: FFFF
(6)     BG 000 M140I NEW DATA: 05EF
        BG 000 M015D ENTER DETAIL STATEMENTS OR "?"
        BG 000 M017I LIST OF ALLOWED DETAIL STATEMENTS:
                    ALTER (AL)      - IDENTIFIES MODIFICATION TO BE MADE
                    SCAN (SC)       - IDENTIFIES SCANNING TO BE PERFORMED
(7)                 CANCEL (CANCEL) - TO QUIT
                    PRESS END/ENTER TO FINISH THE CORRECTION
        BG 000 M015D ENTER DETAIL STATEMENTS OR "?"
      ⇒ 0  ▪
        BG 000 M106I UPDATE OF PHASE   PAYPRNT   SUCCESSFULLY FINISHED
        BG 000 M041I FUNCTION COMPLETED
        BG 000 M005D ENTER CONTROL STATEMENT, OR PRESS END/ENTER TO QUIT
      ⇒ 0  ▪
        BG 000 M009I MSHP EXECUTION COMPLETED
```

Figure 26. Example of an MSHP Patch Operation

Note:

1. Requests MSHP to display help information. MSHP responds by an information message as shown.

2. This is the heading line for a list of allowed detail control statements. For a listed control statement, MSHP shows the allowed abbreviation in parentheses. AFF is the allowed abbreviation for AFFECTS.

3. Requests MSHP to display 16 bytes of data beginning at the address 03FC. In this example, the scan begins at location 03FC (and not at location 0404) to give you some information on the environment of location 0404.

4. Requests MSHP to scan for a string of two bytes containing X'FFFF'.

5. Requests 'FFFF' at the location 0404 to be checked and replaced by a branch and link register instruction.

6. Messages M138I, M139I, and M140I are MSHP's confirmation of the requested change.

7. A response of CANCEL (to quit) at this point causes the function to end, leaving the phase unchanged. If this were done in the above example, the two bytes at location 0404 would still contain 'FFFF'.

Using the Function via Input from SYSIPT

To have MSHP do the same change in phase PAYPRNT as shown in Figure 26 on page 79, submit a job as follows:

```
// JOB PATCH A PHASE
// EXEC MSHP
   PATCH SUBLIB=WEEKLY.PAYAPPL
   AFFECTS PHASE=PAYPRNT
   ALTER 0404 FFFF:05EF
/*
/&
```

Appendix B. Skeletons for Installing APARs/Local Fixes

z/VSE provides skeletons that help you apply APARs or local fixes. The skeletons are provided in VSE/ICCF library 59.

Note: Before you change a skeleton, copy it to your primary VSE/ICCF library. Make your changes to that copy, not to the original.

The skeletons:
- SKSRVSRC
- SKSRVPHS
- SKSRVMOD

can be used to apply a local fix for phases, object modules and source type code.

Skeleton SKUNDO removes any fix.

In addition, for scanning for a certain hexcode within a phase, skeleton SKSRVSCN may be used.

Handling a Fix for a Phase

This section gives sample jobs for the installation of fixes to the various types of library members. These jobs assume that permanently stored label information exists for the affected libraries. For a list of component names – IBM assigned numbers such as 1111-222-33-444, which are used in the first sample jobs – refer to your MSHP RETRACE COMPONENTS listing. In this listing, the component names would be printed, for example, as *1111-22233*.

MSHP can be requested to *expand* a *phase* or a *module* if this phase or module consists of only one CSECT.

The first sample job shown in Figure 27 corrects phase IPW$$OT. The job causes MSHP to expand the phase by 100 bytes. The second sample job shown in Figure 30 on page 83 removes this fix.

Installing the Fix

```
     * $$ JOB JNM=CORPHASE,CLASS=0
     // JOB CORRECT PHASE
     // EXEC MSHP
(1)  CORRECT 1111-222-33-444  :  DY21001
     AFFECTS PHASES=IPW$$OT EXPAND=100
(2)  ALTER F0 9200B0F8:92F180F8
(3)  ALTER 6FA 00000000:4700C426
     RESOLVES 'ERROR ON TAPE OPEN'
     /*
     /&
     * $$ EOJ
```

Figure 27. Sample Job for Installing an APAR/Local Fix to a Phase

This allows you to add code to the end of the affected phase or module. You do this by specifying, in the AFFECTS statement, the number of bytes by which the phase or module is to be expanded. Figure 27 on page 81 shows an AFFECTS statement used to expand a phase by 100 bytes.

Note:

1. *1111-222-33* is the component identifier and *444* the component level code; *DY21001* is the APAR number assigned to the fix.
2. *F0* is the (hexadecimal) displacement into the phase of the data that is to be altered. This example alters the operands of an MVI (92) instruction from 00B0F8 to F180F8.
3. The statement inserts the specified BC (47) instruction into the expansion area.

Additional Sample Jobs

```
* $$ JOB JNM=SRVPHAS,DISP=D,PRI=3,
* $$ NTFY=YES,
* $$ CLASS=0
* $$ LST DISP=H
// JOB SRVPHAS  APPLY APAR/LOCAL FIX TO A PHASE
* PHASE IESFPIP WILL BE EXPANDED BY 14 BYTES AS AN EXAMPLE, THE
* PATCH AREA HOWEVER WILL NOT BE USED.
// EXEC MSHP
 CORRECT 5686-CF8-01-01C : AM00001 IRR
 AFF PHASES = IESFPIP EXP = 14
 ALTER 13A0 4770 : 4780
 RESOLVES -
UPPERCASE TRANSLATION AFTER PF4
/*
/&
* $$ EOJ
```

Figure 28. Sample Job for Installing an APAR/Local Fix to a Phase

```
* $$ JOB JNM=SRVPHAS2,DISP=D,PRI=3,
* $$ NTFY=YES,
* $$ CLASS=0
* $$ LST DISP=H
// JOB SRVPHAS2  APPLY APAR/LOCAL FIX TO A PHASE
* APPLY THE PATCH REVOKABLE
// EXEC MSHP
 CORRECT 5686-CF8-01-01C : AM00002 REV
 AFF PHASES = IESIES01
 ALTER 35C 4870 : 4770
 RESOLVES -
TEST PATCH
/*
/&
* $$ EOJ
```

Figure 29. Sample Job for Installing an APAR/Local Fix to a Phase

Removing the Fix

```
      * $$ JOB JNM=UNDO,CLASS=0
      // JOB UNDO FIX
      // OPTION CATAL
      // EXEC MSHP
  (1) UNDO 1111-222-33-444  :  DY21001
      /*
      /&
      * $$ EOJ
```

Figure 30. Sample Job for Removing an APAR/Local Fix

Note: This job causes MSHP to restore the original instruction and to record the APAR (in the history file) as having been revoked.

Handling a Fix for an Object Deck

Figure 31 shows a sample job for fixing an object deck.

The fix affects the module IESFPIP and is applied irrevocable.

```
* $$ JOB JNM=SRVOBJ,DISP=D,PRI=3,
* $$ NTFY=YES,
* $$ CLASS=0
* $$ LST DISP=H
// JOB SRVOBJ  APPLY APAR/LOCAL FIX TO AN OBJECT DECK
* CSECT 02 SHOULD BE CHANGED LINKBOOK IS IESFPIPL
* FIX IS APPLIED IRREVOCABLE
// EXEC MSHP
 CORRECT 5686-CF8-01-01C : AM00003 IRR
 AFF MODULE = IESFPIP ESDID = 02
 ALTER 23C 4770 : 4780
 INVOLVES  LINK = IESFPIPL
 RESOLVES -
SAMPLE FIX FOR OBJ
/*
/&
* $$ EOJ
```

Figure 31. Sample Job for Installing a Fix for an Object Deck

Handling a Fix for a Macro or Source Code

Figure 32 on page 84 shows a sample job for fixing a macro in the system source statement library.

MSHP does the fix with the REVOKABLE option effective **(by default)**. Therefore, before it alters this macro, MSHP writes a revoke job for the macro to the device assigned to SYSPCH.

The sample job assumes that permanent assignments exist for the required system work files on disk (SYS001 through SYS004) and to a tape drive (SYSPCH).

```
        * $$ JOB JNM=CORSORCA,CLASS=0
        // JOB CORRECT SOURCE MACRO
    (1) // PAUSE ASSGN SYSPCH TO TAPE FOR BACKOUT CREATION
        // EXEC MSHP
        CORRECT 1111-222-33-444 :  PP73336
        AFFECTS MACROS=DLZCKOPT TYPE=A
        DELETE : 000400
        INSERT 450
              LCLB@B(9),@NGP
        @B(9)  SETB  (@PIO(@P))
        /$
        /*
        /&
        * $$ EOJ
```

Figure 32. Sample Job for Installing a Fix for a Macro

Note: (1) Gives the operator a chance to enter an ASSGN command for SYSPCH.

The following job applies a fix to a source type member. The fix is applied revocable. The original source code is written to tape 181. The statement 200 until 250 is replaced by three new lines.

```
* $$ JOB JNM=SRVSOURC,DISP=D,PRI=2,
* $$ NTFY=YES,
* $$ CLASS=0
* $$ LST DISP=H
// JOB SRVSOURC  APPLY APAR/LOCAL FIX TO A SOURCE TYPE MEMBER
* APPLICATION IS DONE REVOKABLE, THE SAVE VERSION IS WRITTEN TO
* TAPE
// PAUSE  MOUNT A TAPE ON 181 FOR THE REVOKABLE FIX FOR AM00004.
// ASSGN SYSPCH,181
// EXEC MSHP
 CORRECT 5686-066-06-75J : AM00004 REV
 AFF MACRO = SGINF TYPE = A
 REPLACE 000200 : 000250
SCRIONA DS    0H                 SCREEN I/O START ADDRESS
        STM   R14,R1,CICSREGS    SAVE CICS REGISTERS
        EXEC CICS ADDRESS CSA(CSABAR) , GET ADDRESS CSA
/$
 RESOLVES -
SAMPLE APAR FOR A SOURCE
/*
// MTC WTM,SYSPCH
* 181 NOW HAS THE OUTPUT FOR THE REVOKABLE APPLICATION OF
* AM00004 AND A TAPE MARK.  181 IS LEFT POSITIONED AFTER THE
* TAPE MARK FOR FURTHER OUTPUT.  EITHER UNLOAD 181 OR
* RUN YOUR NEXT CORRECTIVE SERVICE JOB.
// PAUSE
/*
/&
* $$ EOJ
```

Figure 33. Sample Job for Installing a Fix for a Source Type Member

Removing a Fix for a Macro or Source Code

To remove a fix for a macro:
- Create a job with the VSE/POWER JECL statements as shown in Figure 34 on page 85, if this is possible.
- Start the revoke run with the MSHP produced job on the device assigned to SYSIN (if on tape) or SYSRDR and SYSIPT (if in card image).

```
 * $$ JOB JNM=UNDO,CLASS=0,DISP=H

 *****MSHP generated jobs*****

 // JOB DY17291  (UNDO) xx/xx/xx
 // EXEC MSHP
 UNDO 1111-222-33-444 : DY17291
 DATA
  ...  ...  ...
 /$
 /*
 /&
 // JOB DY18456  (UNDO) xx/xx/xx
 // EXEC MSHP
 UNDO 1111-222-33-444 : DY18456
 DATA
  ...  ...  ...
 /$
 /*
 /&
 * $$ EOJ
```

Figure 34. Sample Job for Removing a Fix from an Edited Macro

Appendix C. Downloading PTFs from the Internet

This chapter describes the steps that are required to prepare the system, if you downloaded the PTFs from the Internet. The shipment you received consists of the following files:

- a readme file
- the PTF cover letter
- the PTF file

The readme and cover letter files are plain text and you can read them without transferring the files to your VSE system first. The PTF file includes all the PTFs ordered, and it must be transferred to the VSE system in binary mode. The PTF file might be compressed by the ZIP facility, and therefore you must first decompress it before you can transfer it to your VSE system.

Transferring the PTF file to the VSE system can be done in different ways. It is recommended to transfer it to a VSAM file called IJSYSPF, which is defined on the VSE system and used by the PTF application dialog (since VSE/ESA 2.4 or later). For older VSE systems, a tape is required as application medium. In this case, you need a VSE library as intermediate storage.

Note: PTF files should not be transferred directly to the VSE system. Read "Restrictions" on page 89 for details.

You can use the PTF file residing on a remote system (for example, a PC) by using a virtual tape. (See PTF application dialog.) This avoids transferring it to the VSE system. The PTF file must be stored as a zipped file (file extension .zip), or as a PTF file (file extension .ptf) on the PC or a Linux on System z® server. If you are using a zipped file, only one .ptf file can be packed into this file.

The following section describes the various possibilities to transfer the PTF file to the VSE system.

Transferring the PTF by using TCP/IP

You can use TCP/IP to transfer the PTF file directly to disk in the VSAM file IJSYSPF. IJSYSPF is defined in the VSAM user catalog. Note that depending on the amount of PTFs to be applied, the size of the file could be too small or VSAM is running out of space. The file must be defined in TCP/IP as follows:

```
DEFINE FILE,TYPE=ESDS,DLBL=IJSYSPF,PUBLIC='PTF.FILE'
```

You must specify the parameter binary and a logical record length of 10320 (quote site lrecl 10320). The following shows a sample protocol of such a transfer:

Downloading PTFs from the Internet

```
C:\Temp>ftp 9.164.170.30                    <-- IP address or hostname of VSE system
Connected to 9.164.170.30
220-TCP/IP for VSE -- Version 01.04.00 -- FTP Daemon
Copyright (c) 1995,2001 Connectivity Systems Incorporated
220 Service ready for new user.
User (9.164.155.2:(none)): SYSA             <-- enter your user id here
331 User name okay, need password.
Password:                                   <-- enter your password here
230 User logged in, proceed.
ftp> binary                                 <-- switch to binary mode
200 Command okay.
ftp> quote site lrecl 10320                    <-- record size of your file
200 Command okay.
ftp> quote site recfm f                     <-- record format of your file
200 Command okay.
ftp> put ptffile.bin PTF.FILE               <-- enter your filenames
```

Note: In z/VSE 5.2 the logical record length is 10320. In previous versions of z/VSE it was 80. Be sure to use the correct LRECL value. You might redefine the PTF file using the skeleton SKPTFILE.

Apply the PTF by using the z/VSE dialog. The service medium is disk.

Transferring the PTF File to Disk

If you transfer the PTF file to disk, an intermediate medium is required, which can be a VSE library member, or the Host Transfer File (HTF).

Transfer the PTF file to a library member or into the HTF using IND$FILE transfer. The following send command transfers the PTF file to the PRIMARY library:

```
SEND PTFFILE.BIN PTFFILE Z (FILE=LIB L=PRIMARY S=SUF LRECL=10320
```

The following send command transfers the PTF file to the HTF:

```
SEND PTFFILE.BIN PTFFILE Z (FILE=HTF LRECL=10320
```

From the intermediate medium, the PTF file can be transferred to disk. This is done using a DITTO job, as shown in the following sample job, in case the intermediate medium is the VSE library:

```
* $$ JOB JNM=TEST,CLASS=0,DISP=D
// JOB TEST
* PTF MEMBER OF LIBRARY IS COPIED (AND REBLOCKED) TO VSAM FILE
* IJSYSPF
*
// UPSI 1
// EXEC DITTO
$$DITTO LV   LIBIN=PRIMARY.SUF
             ,MEMBERIN=PTFFILE.Z
             ,FILEOUT=IJSYSPF
             ,REUSE=REUSE

/*
/&
* $$ EOJ
```

In case the intermediate medium is the HTF, you can use the Move Utility (Fast Path **383**) to move the PTF from HTF to VSAM. If the HTF is used, you must define the IJSYSPF file to CICS. After the move to VSAM, you must close IJSYSPF:

```
CEMT SET FILE(IJSYSPF) CLOSE
```

Apply the PTF by using the z/VSE dialog. The service medium is disk.

Transferring the PTF File to Tape

If you transfer the PTF file to tape, an intermediate medium is required, which must be a library member.

Transfer the PTF file to a library member as described in the previous section. The following send command transfers the PTF file to the PRIMARY library:

```
SEND PTFFILE.BIN PTFFILE Z (FILE=LIB L=PRIMARY S=SUF LRECL=10320
```

From this library, the DITTO job copies the PTF file on the tape with the address 181 as shown in the following sample:

```
* $$ JOB JNM=TEST,CLASS=0,DISP=D
// JOB TEST
* PTF FILE MUST BE COPIED TO SUBLIBRARY PRIMARY.SUF
* AS MEMBER PTFFILE.Z, FILE TRANSFER MUST BE BINARY LRECL 10320
*
// UPSI 1
// PAUSE - PLEASE MOUNT SCRATCH TAPE ON 181
// EXEC DITTO
$$DITTO REW OUTPUT=181
$$DITTO WTM OUTPUT=181,NTMKS=5
$$DITTO LT LIBIN=PRIMARY.SUF,MEMBERIN=PTFFILE.Z,OUTPUT=181,
$$DITTO     RECFMOUT=FB
$$DITTO WTM OUTPUT=181,NTMKS=2
$$DITTO RUN OUTPUT=181
/*
/&
* $$ EOJ
```

Apply the PTF by using the z/VSE dialog. The service medium is tape.

Note: If you are running z/VSE under z/VM®, you can transfer the PTF to VM (binary, lrecl 10320), and use the exec showed below to move it to a tape. To apply the PTF, use the z/VSE dialog.

```
* THIS EXEC CREATES A SERVICE TAPE
CP REW 181
TAPE WTM 5
FILEDEF * CLEAR
FILEDEF IN DISK &1 &2 &3
FILEDEF OUT TAP1 ( RECFM FB LRECL 80 BLOCK 10320 DEN 38K
MOVEFILE IN OUT
TAPE WTM 2
CP REW 181
```

Figure 35. Creating a Service Tape Under VM

Restrictions

PTF files should not be transferred directly to the VSE system (POWER reader), because the following restrictions apply:

1. PRE- and CO-requirements may not be fulfilled due to wrong order of PTFs, CO-requirements will not work at all.
2. POWER JECL included in the data part of the PTFs will be processed while running the job. This results in modified contents of the PTF and also might cause incompleteness of code.

Downloading PTFs from the Internet

3. Missing POWER JECL in the PTF file causes POWER to extract VSE JOB cards, this results in modified contents of the PTF and to incompleteness due to job termination.

4. Indirect application is not supported.

Appendix D. Preparing for Initial Installation or FSU from Virtual Tape

In addition to ordering z/VSE on tapes, you can order it on DVD or download from the Internet. Both methods are based on virtual tape format (AWS tape format). The installation is similar for both methods. With electronic delivery, you must first download the files from the IBM website (ShopzSeries) to your workstation and unzip the files. If z/VSE is distributed on DVD, the data can be used directly on your workstation.

The following table lists the file names that are used for the z/VSE tape images:

Table 7. File Names of Tape Images

Tape Image	DVD	Electronic Delivery (downloaded from internet and unzipped)
Base Tape	VSE*vrm*YY.aws	VSE*vrm*YY.aws
Extended Base Tape	VSE*vrm*XB.aws	VSE*vrm*XB.aws
DB2 Help Tape	DB2SERVE.aws	DB2SERVE.aws
Optional Products Tape	VSE*vrm*OP.aws	VSE*vrm*OP.aws

The first DVD contains the files for the z/VSE Base, Extended Base, and DB2 Help tape images. The optional products are on the second DVD. For electronic delivery, the Download Director creates a directory called `DownloadDirector` and downloads the zipped virtual tape image files (AWSTape image files) to this directory (for example, `C:\DownloadDirector`).

Preparing the Installation

Before you can start to install z/VSE 5.2, you have to perform the following steps:
1. Prepare the virtual tape file for the base tape.
2. Depending on whether you want to perform an initial installation or an FSU,
   - copy the file containing the base tape to a real tape (see "Preparing the Virtual Tape Files on the Workstation"), or
   - transfer the file to VSAM for an FSU via virtual tape.
3. Install the extended base products and the optional products to complete the installation.

Preparing the Virtual Tape Files on the Workstation

Unzip the downloaded files by using a compression utility (for example, PKUNZIP). The zipped files contain one virtual tape image file each. If you received z/VSE on DVD, the virtual tape files are already in virtual tape format (aws) and ready to use.

The following sections describe the steps that are required to install or upgrade z/VSE from DVD.

91

Preparing for Initial Installation or FSU

You can start the FSU from the virtual tape, or copy the base tape, which is in AWS tape format, to a real tape. If you plan to install z/VSE 5.2 on a system that is capable of reading tape images in virtual tape format, you do not need to copy the image to a real tape or VSAM. You can directly continue with the initial installation or FSU, by using the distribution file on an emulated tape device.

In order to copy the files to a real tape, the cartridge must be capable to hold at least 250 MB.

Refer to the *z/VSE Program Directory*, or to the installation instructions available on the z/VSE website for the latest information about the required APARs.

To copy the virtual tape to a real tape, use the VSE Virtual Tape Support. This can be done either from the remote site (workstation) or via VSAM and FTP and subsequent DITTO tape-to-tape copy.

A document with operation considerations of the VSE Virtual Tape Support is available on the z/VSE Service and Support web page.

Using a Remote Tape Image on a Workstation to Copy the Base Tape to Real Tape

Perform the following steps to copy the tape image. Run the VSE job, as shown below, to copy the base tape image in virtual tape format residing on a workstation to a real tape. The job uses a virtual tape at address *cuu1* and performs a DITTO tape-to-tape copy to a real tape at *cuu2*. See the example below for the parameters you must specify. Make sure that the VSE Virtual Tape Server is started on your workstation.

```
* $$ JOB JNM=COPYTAPE,CLASS=0,DISP=D
// JOB COPYTAPE
// PAUSE PLEASE MOUNT A TAPE ON <cuu2>
// UPSI 1
DVCDN <cuu1>
// PAUSE - DEVICE DOWN OK
// VTAPE START,UNIT=<cuu1>,LOC=<ip-address>,FILE='<tape image file>',READ
DVCUP <cuu1>
// ASSGN SYS010,<cuu2>,08 * OUTPUT MEDIA, COMPRESSED
// ASSGN SYS011,<cuu1>      * INPUT   MEDIA
// EXEC DITTO
$$DITTO REW OUTPUT=SYS010
$$DITTO REW OUTPUT=SYS011
$$DITTO TT INPUT=SYS011,OUTPUT=SYS010,NFILES=<39>
/*
DVCDN <cuu1>
// VTAPE STOP,UNIT=<cuu1>
DVCUP <cuu1>
/&
* $$ EOJ
```

Before submitting the job, you must enter the following parameters:

<cuu1>
   Device address of the input tape (virtual).

<cuu2>
   Device address of the output tape (real).

<ip-address>
   IP address of the workstation where the VSE Virtual Tape Server is running.

<tape image file>
>   File name and path of the remote virtual tape image containing the z/VSE
>   base tape. (For example, `C:\DownloadDirector\VSEvrmYY.AWS`)

Note:

1. `NFILES=<39>` is only valid for the base tape. The value of the extended base tape is 49. For details on the extended base tape refer to the *z/VSE Program Directory*.

2. FSU can also be done by using a real tape or a virtual tape on VSAM.

After you copied the tape image to a real tape, continue with the installation or FSU of the z/VSE 5.2 base tape.

Using a VSAM Tape Image on a z/VSE System

Perform the following steps to use the VSAM tape image on your z/VSE system:

* Define a virtual tape file in VSAM using skeleton SKVTAPE (ICCF library 59). You need about 250 MB of VSAM space to copy the base tape.

* Transfer the base tape in virtual tape format (type aws) to VSAM using FTP. For more details, see "Uploading the Base Tape to VSAM for FSU via Virtual Tape."

* Copy the VSAM base virtual tape to a real tape with a DITTO job as described in "Using a Remote Tape Image on a Workstation to Copy the Base Tape to Real Tape" on page 92 with the following VTAPE command:

  ```
  // VTAPE START,UNIT=<cuu1>,LOC=VSAM,FILE='<tape image file>',READ
  ```

  <tape image file> is the label name of the VSAM file containing the tape image (for example, VTAPE1).

After copying the tape image to a real tape, continue with the installation or FSU of the z/VSE 5.2 base tape.

Uploading the Base Tape to VSAM for FSU via Virtual Tape

If you are performing a release upgrade by using a virtual tape, use the skeleton SKVTAPE, provided in ICCF library 59, to define a VSAM ESDS cluster. The catalog should have about 250 MB free space.

Upload the tape image into a VSAM data set. The base tape is loaded into the VSAM data set using TCP/IP FTP. To do this you need to have an FTP daemon defined in the TCP/IP for VSE/ESA partition. Use the following commands for the upload:

```
C:\DownloadDirector>ftp 10.2.3.45          <-- IP address or hostname of VSE system
Connected to 10.2.3.45.
220-TCP/IP for VSE -- Version 01.05.00 -- FTP Daemon
Copyright (c) 1995,2001 Connectivity Systems Incorporated
220 Service ready for new user.
User (10.2.3.45:(none)): sysa             <-- enter your user id here
331 User name okay, need password.
Password:                                 <-- enter your password
230 User logged in, proceed.
ftp> bin                                  <-- switch to binary mode
200 Command okay.
ftp> quote site lrecl 32758               <-- record size of your file
200 Command okay.
ftp> quote site recfm v                   <-- set record format to variable
200 Command okay.
ftp> put VSEvrmYY.AWS VSE.VTAPE.FILE      <-- enter your filenames
```

After loading the base tape you can start the FSU from virtual tape.

Installing Additional Products

Products from the extended base tape as well as optional products can be installed from the remote server (workstation) using the Interactive Interface dialog. The related virtual tape files can also be transferred to VSAM by using FTP. Installation from a VSAM virtual tape is also done using the Interactive Interface dialog. If required, the virtual tape can also be copied to a real tape, the number of files to be copied depends on the amount of products on the tape. The DITTO tape-to-tape copy job can specify 999 files, but this will generate an error message on the virtual input tape.

To install the DB2 Help feature, either specify a virtual tape address referring to the remote virtual tape file during the DB2 installation, or copy the virtual DB2 Help tape to a real tape.

Additional Information for Receiving z/VSE from the Internet

Depending on your order, it can contain items that are packaged in zipped format. In this case, perform the following:

1. Download the *xxxxxxx*.zip file to your workstation.
2. Extract the files using an unzip function.
3. Browse the extracted material for files like `readme.txt`, `index.htm`, or `index.html` to get started.

Most of the extracted files can be used directly. In some cases your order might contain ADDITIONAL MATERIAL or ADDITIONAL PUBLICATIONS that were originally packaged on DVD. These might be provided as ISO 9660 images with a file extension of .iso. An ISO 9660 DVD-ROM image is a single large file that is an exact representation of the data and programs as they appear on a DVD, reflecting both the content and logical format.

To use .iso files, you have two options:

1. Create a physical DVD. This requires that your workstation has a DVD-write capability and software that supports ISO 9660 format. When you create the physical DVD, this is an exact copy of the original DVD and has all of the characteristics of the original image (for example, special file names, and - if applicable - it is a bootable DVD).
2. Use virtual DVD software. Virtual DVD software emulates your computer's DVD drive, enabling you to execute programs, view and use data provided in the DVD image directories and files. This is an alternative to creating a physical DVD. This software must support .iso files.

Read the license agreements and follow the procedures that are specific to any software that you use to process these packages.

File Names and Other Considerations When Using Virtual Tapes

If the required Linux, UNIX, or Windows file for a remote virtual tape does not exist, it will be automatically created after the corresponding VTAPE START command has been submitted. When assigning file names you must observe certain rules and characteristics as outlined below.

Linux and UNIX Considerations

Linux and UNIX are *case sensitive*, but job streams created on the z/VSE

host (using dialogs of the Interactive Interface) are in *capital letters*. It may be therefore necessary to edit such job streams to adapt the filename to the Linux or UNIX conventions.

Windows Considerations

Windows file names may contain blanks, therefore the filename must be enclosed in quotes. A quote within a filename must be coded as two single quotes. For example:

```
FILE='D:\John''s\Virtual Tapes\vt001401.001'
```

Windows file names can have more than 100 characters in length. Since the limit for remote files is 100 characters, you may specify FILE='filename' twice or even three times. The filename is concatenated in storage, thus allowing for a file name length of 200 or even 300. The following example is equivalent to the previous example:

```
FILE='D:',FILE='\John''s\Virtual Tapes\',FILE='vt001401.001'
```

Example

This is an example of a job that has been generated by the *Prepare for Installation* dialog:

```
* $$ JOB JNM=INSPRE,DISP=D,PRI=3,                                     C
* $$ NTFY=YES,                                                        C
* $$ LDEST=*,                                                         C
* $$ CLASS=0
// JOB INSPRE  SCAN OPTIONAL PRODUCT TAPE
// LIBDEF PHASE,SEARCH=(PRD1.BASE,IJSYSRS.SYSLIB)
* *
* *      PREPARE ADDITIONAL PROGRAM INSTALLATION
* *      -  SCAN PROGRAM TAPE
* *
* *  VIRTUAL TAPE SPECIFIED, NO REAL TAPE DRIVE REQUIRED ON
* *               280
// EXEC IJBVTDLG
UNIT=280,
LOC=123.123.123.123,
FILE='DATASET*******************************'
FILE='**************************************'
FILE='**************************************'
FILE='**************************************'
FILE='**************************************'
FILE='**************************************'
FILE='************************'
READ                                                                 /*
// ASSGN SYS006,280
// MTC REW,SYS006
// EXEC DTRIPRE,PARM='VDDR=280'
/*
// ASSGN SYS006,UA
// VTAPE STOP,UNIT=280
/&
* $$ EOJ
```

Before it is submitted, the above job can be stored in a library and edited as required. For example, you might need to use the SET CASE MIXED command to produce the statements shown below:

```
:
FILE='my*TEST.file****************************'
FILE='**************************************'
FILE='**************************************'
FILE='**************************************'
```

Installing from Virtual Tape

```
FILE='**************************************'
FILE='**************************************'
FILE='**********************'
  .
  .
  .
```

When the job is finally submitted, a mixed-case file name will therefore be used.

Glossary

This glossary includes terms and definitions for IBM z/VSE.

The following cross-references are used in this glossary:
1. See refers the reader from a term to a preferred synonym, or from an acronym or abbreviation to the defined full form.
2. See also refers the reader to a related or contrasting term.

To view glossaries for other IBM products, go to www.ibm.com/software/globalization/terminology.

A

Access Control Logging and Reporting. An IBM licensed program to log all attempts of access to protected data and to print selected formatted reports on such attempts.

access control table (DTSECTAB). A table that is used by the system to verify a user's right to access a certain resource.

access list. A table in which each entry specifies an address space or data space that a program can reference.

access method. A program, that is, a set of commands (macros) to define files or addresses and to move data to and from them; for example VSE/VSAM or VTAM.

account file. A disk file that is maintained by VSE/POWER containing accounting information that is generated by VSE/POWER and the programs running under VSE/POWER.

addressing mode (AMODE). A program attribute that refers to the address length that a program is prepared to handle on entry. Addresses can be either 24 bits or 31 bits in length. In 24 bit addressing mode, the processor treats all virtual addresses as 24-bit values; in 31 bit addressing mode, the processor treats all virtual addresses as 31-bit values. Programs with an addressing mode of ANY can receive control in either 24 bit or 31 bit addressing mode.

administration console. In z/VSE, one or more consoles that receive all system messages, except for those that are directed to one particular console. Contrast this with the user console, which receives only those messages that are directed to it, for example messages that are issued from a job that was submitted

with the request to echo its messages to that console. The operator of an administration console can reply to all outstanding messages and enter all system commands.

alternate block. On an FBA disk, a block that is designated to contain data in place of a defective block.

alternate index. In systems with VSE/VSAM, the index entries of a given base cluster that is organized by an alternate key, that is, a key other than the prime key of the base cluster. For example, a personnel file preliminary ordered by names can be indexed also by department number.

alternate library. An interactively accessible library that can be accessed from a terminal when the user of that terminal issues a connect or switch library request.

alternate track. A library, which becomes accessible from a terminal when the user of that terminal issues a connect or switch (library) request.

AMODE. Addressing mode.

APA. All points addressable.

APAR. Authorized Program Analysis Report.

appendage routine. A piece of code that is physically located in a program or subsystem, but logically and extension of a supervisor routine.

application profile. A control block in which the system stores the characteristics of one or more application programs.

application program. A program that is written for or by a user that applies directly to the user's work, such as a program that does inventory control or payroll. See also batch program and online application program.

AR/GPR. Access register and general-purpose register pair.

ASC mode. Address space control mode.

ASI (automated system initialization) procedure. A set of control statements, which specifies values for an automatic system initialization.

attention routine (AR). A routine of the system that receives control when the operator presses the Attention key. The routine sets up the console for the input of a command, reads the command, and initiates the system service that is requested by the command.

automated system initialization (ASI). A function that allows control information for system startup to be cataloged for automatic retrieval during system startup.

autostart. A facility that starts VSE/POWER with little or no operator involvement.

auxiliary storage. Addressable storage that is not part of the processor, for example storage on a disk unit. Synonymous with external storage.

B

B-transient. A phase with a name beginning with $$B and running in the Logical Transient Area (LTA). Such a phase is activated by special supervisor calls.

bar. 2 GigyByte (GB) line

basic telecommunications access method (BTAM). An access method that permits read and write communication with remote devices. BTAM is not supported on z/VSE.

BIG-DASD. A subtype of Large DASD that has a capacity of more than 64 K tracks and uses up to 10017 cylinders of the disk.

block. Usually, a block consists of several records of a file that are transmitted as a unit. But if records are very large, a block can also be part of a record only. On an FBA disk, a block is a string of 512 bytes of data. See also a control block.

block group. In VSE/POWER, the basic organizational unit for fixed-block architecture (FBA) devices. Each block group consists of a number of 'units of transfer' or blocks.

C

CA splitting. Is the host part of the VSE JavaBeans, and is started using the job STARTVCS, which is placed in the reader queue during installation of z/VSE. Runs by default in dynamic class R. In VSE/VSAM, to double a control area dynamically and distribute its CIs evenly when the specified minimum of free space get used up by more data.

carriage control character. The fist character of an output record (line) that is to be printed; it determines how many lines should be skipped before the next line is printed.

catalog. A directory of files and libraries, with reference to their locations. A catalog may contain other information such as the types of devices in which the files are stored, passwords, blocking factors. To store a library member such as a phase, module, or book in a sublibrary. See also VSE/VSAM catalog.

cell pool. An area of virtual storage that is obtained by an application program and managed by the callable cell pool services. A cell pool is located in an address space or a data space and contains an anchor, at least one extent, and any number of cells of the same size.

central location. The place at which a computer system's control device, normally the systems console in the computer room, is installed.

chained sublibraries. A facility that allows sublibraries to be chained by specifying the sequence in which they must be searched for a certain library member.

chaining. A logical connection of sublibraries to be searched by the system for members of the same type (phases or object modules, for example).

channel command word (CWW). A doubleword at the location in main storage that is specified by the channel address word. One or more CCWs make up the channel program that directs data channel operations.

channel program. One or more channel command words that control a sequence of data channel operations. Execution of this sequence is initiated by a start subchannel instruction.

channel scheduler. The part of the supervisor that controls all input/output operations.

channel subsystem. A feature of 370-XA and Enterprise Systems Architecture that provides extensive additional channel (I/O) capabilities over the System/370.

channel to channel attachment (CTCA). A function that allows data to be exchanged
1. Under the control of VSE/POWER between two virtual VSE machines running under VM or
2. Under the control of VTAM between two processors.

character-coded request. A request that is encoded and transmitted as a character string. Contrast with *field-formatted request*.

checkpoint.
1. A point at which information about the status of a job and the system can be recorded so that the job step can be restarted later.
2. To record such information.

CICS (Customer Information Control System). An IBM program that controls online communication between terminal users and a database. Transactions that are entered at remote terminals are processed concurrently by user-written application programs. The program includes facilities for building, using, and servicing databases.

CICS ECI. The CICS External Call Interface (ECI) is one possible requester type of the *CICS business logic interface* that is provided by the CICS Transaction Server for VSE/ESA. It is part of the CICS client and allows workstation programs to CICS function on the z/VSE host.

CICS EXCI. The EXternal CICS Interface (EXCI) is one possible requester type of the *CICS business logic interface* that is provided by the CICS Transaction Server for VSE/ESA. It allows any BSE batch application to call CICS functions.

CICS system definition (CSD) file. Is the host part of the VSE JavaBeans, and is started using the job STARTVCS, which is placed in the reader queue during installation of z/VSE. Runs by default in dynamic class R. See CSD.

CICS Transaction Server for VSE/ESA. A z/VSE base program that controls online communication between terminal users and a database. This is the successor system to CICS/VSE.

CICS TS. CICS Transaction Server

CICS/VSE. Customer Information Control System/VSE. No longer shipped on the Extended Base Tape and no longer supported, cannot run on z/VSE 5.1.

class. In VSE/POWER, a group of jobs that either come from the same input device or go to the same output device.

cluster controller. A hardware unit to control the input/output operations of more than one device that is connected to it. A cluster controller might be run by a program that is stored and executed in the unit; for example, the IBM 3601 Finance Communication Controller. Or it might be controlled entirely by hardware; for example, the IBM 3272 Control Unit.

Common Connector Framework (CCF). Is part of IBM's *Visual Age for Java*, and allows connections to remote hosts to be created and maintained. The CCF classes are contained in the VSEConnector.jar file and are used internally by the VSE JavaBeans. CCF is important for multitier architectures where, for example, servlets run on a middle-tier platform. Because CCF allows open connections to be kept in a pool, this avoids the time that is involved in opening and closing TCP/IP connection to the remote z/VSE host each time a servlet is invoked.

CMS. Conversational monitor system running on z/VM.

common library. A library that can be interactively accessed by any user of the (sub)system that owns the library.

communication adapter. A circuit card with associated software that enables a processor, controller, or other device to be connected to a network.

communication region. An area of the supervisor that is set aside for transfer of information within and between programs.

component.
1. Hardware or software that is part of a computer system.
2. A functional part of a product, which is identified by a component identifier.
3. In z/VSE, a component program such as VSE/POWER or VTAM.
4. In VSE/VSAM, a named, cataloged group of stored records, such as the data component or index component of a key-sequenced file or alternate index.

component identifier. A 12-byte alphanumeric string, uniquely defining a component to MSHP.

conditional job control. The capability of the job control program to process or to skip one or more statements that are based on a condition that is tested by the program.

connect. To authorize library access on the lowest level. A modifier such as "read" or "write" is required for the specified use of a sublibrary.

connection pooling. Introduced with an z/VSE 5.1 update to manage (reuse) connections of the z/VSE database connector in CICS TS.

ConnectionManager class. Is part of CCF, and identifies the connection to a remote z/VSE host: it holds connections between the middle-tier and the remote z/VSE server. Servlets can reserve a connection from the pool, work with it and give it back later. This is performed internally using VSE JavaBeans.

connector. In the context of z/VSE, a connector provides the middleware to connect two platforms: Web Client and z/VSE host, middle-tier and z/VSE host, or Web Client and middle-tier.

connector (e-business connector). A piece of software that is provided to connect to heterogeneous environments. Most connectors communicate to non-z/VSE Java-capable platforms.

container. Is part of the JVM of application servers such as the IBM WebSphere Application Server, and facilitates the implementation of servlets, EJBs, and JSPs, by providing resource and transaction management resources. For example, an EJB developer must not code against the JVM of the application server, but instead against the interface that is provided by the container. The main role of a container is to act as an intermediary between EJBs and clients, Is the host part of the VSE JavaBeans, and is started using the job

STARTVCS, which is placed in the reader queue during the installation of z/VSE. Runs by default in dynamic class R. and also to manage multiple EJB instances. After EJBs have been written, they must be stored in a container residing on an application server. The container then manages all threading and client-interactions with the EJBs, and co-ordinate connection- and instance pooling.

control interval (CI). A fixed-length area of disk storage where VSE/VSAM stores records and distributes free space. It is the unit of information that VSE/VSAM transfers to or from disk storage. For FBA it must be an integral multiple to be defined at cluster definition, of the block size.

control program. A program to schedule and supervise the running of programs in a system.

conversational monitor system (CMS). A virtual machine operating system that provides general interactive time sharing, problem solving, and program development capabilities and operates under the control of z/VM.

count-key-data (CKD) device. A disk device that store data in the record format: count field, key field, data field. The count field contains, among others, the address of the record in the format: cylinder, head (track), record number, and the length of the data field. The key field, if present, contains the record's key or search argument. CKD disk space is allocated by tracks and cylinders. Contrast with *FBA disk device. See also extended count-key-data device.*

cross-partition communication control. A facility that enables VSE subsystems and user programs to communicate with each other; for example, with VSE/POWER.

cryptographic token. Usually referred to simply as a *token*, this is a device, which provides an interface for performing cryptographic functions like generating digital signatures or encrypting data.

cryptography.
1. The transformation of data to conceal its meaning.
2. In computer security, the principles, means, and methods for encrypting 'plaintext' and Is the host part of the VSE JavaBeans, and is started using the job STARTVCS, which is placed in the reader queue during installation of z/VSE. Runs by default in dynamic class R.decrypting 'ciphertext'.

D

data block group. The smallest unit of space that can be allocated to a VSE/POWER job on the data file. This allocation is independent of any device characteristics.

data conversion descriptor file (DCDF). With a DCDF, you can convert individual fields within a

record during data transfer between a PC and its host. The DCDF defines the record fields of a particular file for both, the PC and the host environment.

data import. The process of reformatting data that was used under one operating system such that it can subsequently be used under a different operating system.

Data Interfile Transfer, Testing, and Operations (DITTO) utility. An IBM program that provides file-to-file services for card I/O, tape, and disk devices. The latest version is called DITTO/ESA for VSE.

Data Language/I (DL/I). A database access language that is used with CICS.

data link. In SNA, the combination of the link connection and the link stations joining network noes, for example, a z/Architecture channel and its associated protocols. A link is both logical and physical.

data security. Is the host part of the VSE JavaBeans, and is started using the job STARTVCS, which is placed in the reader queue during installation of z/VSE. Runs by default in dynamic class R. See *access control*.

data set header record. In VSE/POWER abbreviated as DSHR, alias NDH or DSH. An NJE control record either preceding output data or, in the middle of input data, indicating a change in the data format.

data space. A range of up to 2 gigabytes of contiguous virtual storage addresses that a program can directly manipulate through ESA/370 instructions. Unlike an address space, a data space can hold only user data; it does not contain shared areas, system data, or programs. Instructions do not execute in a data space, although in a program can reside in a data space as nonexecutable code. Contrast with address space.

data terminal equipment (DTE). In SNA, the part of a data station that serves a data source, data sink, or both.

database connector. Is a function introduced with z/VSE 5.1.1, which consists of a client and server part. The client provides an API (CBCLI) to be used by applications on z/VSE, the server on any Java capable platform connects a JDBC driver that is provided by the database. Both client and server communicate via TCP/IP.

Database 2 (DB2). An IBM rational database management system.

DB2-based connector. Is a feature introduced with VSE/ESA 2.5, which includes a customized DB2 version, together with VSAM and DL/I functionality, to provide access to DB2, VSAM, and DL/I data, using DB2 Stored Procedures.

DB2 Runtime only Client edition. The Client Edition for z/VSE comes with some enhanced features and improved performance to integrate z/VSE and Linux on System z.

DB2 Stored Procedure. In the context of z/VSE, a DB2 Stored Procedure is a Language Environment (LE) program that accesses DB2 data. However, from VSE/ESA 2.5 onwards you can also access VSAM and DL/I data using a DB2 Stored Procedure. In this way, it is possible to exchange data between VSAM and DB2.

DBLK. Data block.

DCDF. Data conversion descriptor file.

deblocking. The process of making each record of a block available for processing.

dedicated (disk) device. A device that cannot be shared among users.

device address.
1. The identification of an input/output device by its device number.
2. In data communication, the identification of any device to which data can be sent or from which data can be received.

device driving system (DDS). A software system external to VSE/POWER, such as a CICS spooler or PSF, that writes spooled output to a destination device.

Device Support Facilities (DSF). An IBM supplied system control program for performing operations on disk volumes so that they can be accessed by IBM and user programs. Examples of these operations are initializing a disk volume and assigning an alternative track.

device type code. The four- or five-digit code that is used for defining an I/O device to a computer system.

dialog. In an interactive system, a series of related inquiries and responses similar to a conversation between two people. For z/VSE, a set of panels that can be used to complete a specific task; for example, defining a file.

dialog manager. The program component of z/VSE that provides for ease of communication between user and system.

digital signature. In computer security, encrypted data, which is appended to or part of a message, that enables a recipient to prove the identity of the sender.

Digital Signature Algorithm (DSA). The Digital Signature Algorithm is the US government-defined standard for digital signatures. The DSA digital signature is a pair of large numbers, computed using a set of rules (that is, the DSA) and a set of parameters such that the identity of the signatory and integrity of the data can be verified. The DSA provides the capability to generate and verify signatures.

directory. In z/VSE the index for the program libraries.

direct access. Accessing data on a storage device using their address and not their sequence. This is the typical access on disk devices as opposed to magnetic tapes. Contrast with *sequential access*.

disk operating system residence volume (DORSES). The disk volume on which the system sublibrary IJSYSRS.SYSLIB is located including the programs and procedures that are required for system startup.

disk sharing. An option that lets independent computer systems uses common data on shared disk devices.

disposition. A means of indicating to VSE/POWER how a job input or output entry is to be handled: according to its local disposition in the RDR/LST/PUN queue or its transmission disposition when residing in the XMT queue. A job might, for example, be deleted or kept after processing.

distribution tape. A magnetic tape that contains, for example, a preconfigured operating system like z/VSE. This tape is shipped to the customer for program installation.

DITTO/ESA for VSE. Data Interfile Transfer, Testing, and Operations utility. An IBM program that provides file-to-file services for disk, tape, and card devices.

DSF. Device Support Facilities.

DSH (R). Data set header record.

dummy device. A device address with no real I/O device behind it. Input and output for that device address are spooled on disk.

duplex. Pertaining to communication in which data can be sent and received at the same time.

DU-AL (dispatchable unit - access list). The access list that is associated with a z/VSE main task or subtask. A program uses the DU-AL associated with its task and the PASN-AL associated with its partition. See also *PASN-AL*.

dynamic class table. Defines the characteristics of dynamic partitions.

dynamic partition. A partition that is created and activated on an 'as needed' basis that does not use fixed static allocations. After processing, the occupied space is released. Dynamic partitions are grouped by class, and jobs are scheduled by class. Contrast with *static partition*.

dynamic partition balancing. A z/VSE facility that allows the user to specify that two or more or all partitions of the system should receive about the same amount of time on the processor.

dynamic space reclamation. A librarian function that provides for space that is freed by the deletion of a library member to become reusable automatically.

E

ECI. See *CICS ECI*.

emulation. The use of programming techniques and special machine features that permit a computer system to execute programs that are written for another system or for the use of I/O devices different from those that are available.

emulation program (EP). An IBM control program that allows a channel-attached 3705 or 3725 communication controller to emulate the functions of an IBM 2701 Data Adapter Unit, or an IBM 2703 Transmission Control.

end user.
1. A person who makes use of an application program.
2. In SNA, the ultimate source or destination of user data flowing through an SNA network. Might be an application program or a terminal operator.

Enterprise Java Bean. An EJB is a distributed bean. "Distributed" means, that one part of an EJB runs inside the JVM of a web application server, while the other part runs inside the JVM of a web browser. An EJB either represents one data row in a database (entity bean), or a connection to a remote database (session bean). Normally, both types of an EJB work together. This allows to represent and access data in a standardized way in heterogeneous environments with relational and non-relational data. See also *JavaBean*.

entry-sequenced file. A VSE/VSAM file whose records are loaded without respect to their contents and whose relative byte addresses cannot change. Records are retrieved and stored by addressed access, and new records are added to the end of the file.

Environmental Record Editing and Printing (EREP) program. A z/VSE base program that makes the data that is contained in the system record file available for further analysis.

EPI. See *CICS EPI*.

ESCON Channel (Enterprise Systems Connection Channel). A serial channel, using fiber optic cabling, that provides a high-speed connection between host and control units for I/O devices. It complies with the ESA/390 and System z I/O Interface until z114. The zEC12 processors do not support ESCON channels.

exit routine.
1. Either of two types of routines: installation exit routines or user exit routines. Synonymous with exit program.
2. See *user exit routine*.

extended addressability. See *31 bit addressing*. The ability of a program to use virtual storage that is outside the address space in which the program is running. Generally, instructions and data reside in a single address space - the primary address space. However, a program can have data in address spaces other than the primary or in data spaces. (The instructions remain in the primary address space, while the data can reside in another address space, or in a data space.) To access data in other address spaces, a program must use access registers (ARs) and execute in access register mode (AR mode).

extended recovery facility (XRF). In z/VSE, a feature of CICS that provides for enhanced availability of CICS by offering one CICS system as a backup of another.

External Security Manager (ESM). A priced vendor product that can provide extended functionality and flexibility that is compared to that of the Basic Security Manager (BSM), which is part of z/VSE.

F

FASTCOPY. See *VSE/Fast Copy.*

fast copy data set program (VSE/Fast Copy). See *VSE/Fast Copy*.

fast service upgrade (FSU). A service function of z/VSE for the installation of a refresh release without regenerating control information such as library control tables.

FAT-DASD. A subtype of Large DASD, it supports a device with more than 4369 cylinders (64 K tracks) up to 64 K cylinders.

FCOPY. See *VSE/Fast Copy.*

fence. A separation of one or more components or elements from the remainder of a processor complex. The separation is by logical boundaries. It allows simultaneous user operations and maintenance procedures.

fetch.
1. To locate and load a quantity of data from storage.
2. To bring a program phase into virtual storage from a sublibrary and pass control to this phase.
3. The name of the macro instruction (FETCH) used to accomplish 2. See also *loader.*

Fibre Channel Protocol (FCP). A combination of hardware and software conforming to the Fibre Channel standards and allowing system and peripheral

connections via FICON and FICON Express feature cards on IBM zSeries processors. In z/VSE, zSeries FCP is employed to access industry-standard SCSI disk devices.

fragmentation (of storage). Inability to allocate unused sections (fragments) of storage in the real or virtual address range of virtual storage.

FSU. Fast service upgrade.

FULIST (FUnction LIST). A type of selection panel that displays a set of files and/or functions for the choice of the user.

G

generation. See *macro generation.*

generation feature. An IBM licensed program order option that is used to tailer the object code of a program to user requirements.

GETVIS space. Storage space within partition or the shared virtual area, available for dynamic allocation to programs.

guest system. A data processing system that runs under control of another (host) system. On the mainframe z/VSE can run as a guest of z/VM.

H

hard wait. The condition of a processor when all operations are suspended. System recovery from a hard wait is impossible without performing a new system startup.

hash function. A hash function is a transformation that takes a variable-size input and returns a fixed-size string, which is called the hash value. In cryptography, the hash functions should have some additional properties:
- The hash function should be easy to compute.
- The hash function is one way; that is, it is impossible to calculate the 'inverse' function.
- The hash function is collision-free; that is, it is impossible that different input leads to the same hash value.

hash value. The fixed-sized string resulting after applying a *hash function* to a text.

High-Level Assembler for VSE. A programming language providing enhanced assembler programming support. It is a base program of z/VSE.

home interface. Provides the methods to instantiate a new EJB object, introspect an EJB, and remove an EJB instantiation., as for the remote interface is needed because the deployment tool generates the

implementation class. Every Session bean's home interface must supply at least one *create()* method.

host mode. In this operating mode, a PC can access a VSE host. For programmable workstation (PWS) functions, the Move Utilities of VSE can be used.

host system. The controlling or highest level system in a data communication configuration.

host transfer file (HTF). Used by the Workstation File Transfer Support of z/VSE as an intermediate storage area for files that are sent to and from IBM personal computers.

HTTP Session. In the context of z/VSE, identifies the web-browser client that calls a servlet (in other words, identifies the connection between the client and the middle-tier platform).

I

ICCF. See *VSE/ICCF.*

ICKDSF (Device Support Facilities). A z/VSE base program that supports the installation, use, and maintenance of IBM disk devices.

include function. Retrieves a library member for inclusion in program input.

index.
1. A table that is used to locate records in an indexed sequential data set or on indexed file.
2. In, an ordered collection of pairs, each consisting of a key and a pointer, used by to sequence and locate the records of a key-sequenced data set or file; it is organized in levels of index records. See also *alternate index.*

input/output control system (IOCS). A group of IBM supplied routines that handle the transfer of data between main storage and auxiliary storage devices.

integrated communication adapter (ICA). The part of a processor where multiple lines can be connected.

integrated console. In z/VSE, the service processor console available on IBM System z server that operates as the z/VSE system console. The integrated console is typically used during IPL and for recovery purposes when no other console is available.

Interactive Computing and Control Facility (ICCF). An IBM licensed program that serves as interface, on a time-slice basis, to authorized users of terminals that are linked to the system's processor.

interactive partition. An area of virtual storage for the purpose of processing a job that was submitted interactively via VSE/ICCF.

Interactive User Communication Vehicle (IUCV).
Programming support available in a VSE supervisor for operation under z/VM. The support allows users to communicate with other users or with CP in the same way they would with a non-preferred guest.

intermediate storage.   Any storage device that is used to hold data temporarily before it is processed.

IOCS.   Input/output control system.

IPL.   Initial program load.

irrecoverable error.   An error for which recovery is impossible without the use of recovery techniques external to the computer program or run.

IUCV.   Interactive User Communication Vehicle.

J

JAR.   Is a platform-independent file format that aggregates many files into one. Multiple applets and their requisite components (.class files, images, and sounds) can be bundled in a JAR file, and then downloaded to a web browser using a single HTTP transaction (much improving the download speed). The JAR format also supports compression, which reduces the files size (and further improves the download speed). The compression algorithm that is used is fully compatible with the ZIP algorithm. The owner of an applet can also digitally sign individual entries in a JAR file to authenticate their origin.

Java application.   A Java program that runs inside the JVM of your web browser. The program's code resides on a local hard disk or on the LAN. Java applications might be large programs using graphical interfaces. Java applications have unlimited access to all your local resources.

Java bytecode.   Bytecode is created when a file containing Java source language statements is compiled. The compiled Java code or "bytecode" is similar to any program module or file that is ready to be executed (run on a computer so that instructions are performed one at a time). However, the instructions in the bytecode are really instructions to the *Java Virtual Machine*. Instead of being interpreted one instruction at a time, bytecode is instead recompiled for each operating-system platform using a just-in-time (JIT) compiler. Usually, this enables the Java program to run faster. Bytecode is contained in binary files that have the suffix **.CLASS**

Java servlet.   See *servlet.*

JHR.   Job header record.

job accounting interface.   A function that accumulates accounting information for each job step, to be used for charging the users of the system, for planning new applications, and for supervising system operation more efficiently.

job accounting table.   An area in the supervisor where accounting information is accumulated for the user.

job catalog.   A catalog made available for a job by means of the file name IJSYSUC in the respective DLBL statement.

job entry control language (JECL).   A control language that allows the programmer to specify how VSE/POWER should handle a job.

job step.   In 1 of a group of related programs complete with the JCL statements necessary for a particular run. Every job step is identified in the job stream by an EXEC statement under one JOB statement for the whole job.

job trailer record (JTR).   As VSE/POWER parameter JTR, alias NJT. An NJE control record terminating a job entry in the input or output queue and providing accounting information.

K

key.   In VSE/VSAM, one or several characters that are taken from a certain field (key field) in data records for identification and sequence of index entries or of the records themselves.

key sequence.   The collating sequence either of records themselves or of their keys in the index or both. The key sequence is alphanumeric.

key-sequenced file.   A VSE/VSAM file whose records are loaded in key sequence and controlled by an index. Records are retrieved and stored by keyed access or by addressed access, and new records are inserted in the file in key sequence.

KSDS.   Key-sequenced data sets. See *key-sequenced file.*

L

label.
1. An identification record for a tape, disk, or diskette volume or for a file on such a volume.
2. In assembly language programming, a named instruction that is generally used for branching.

label information area.   An area on a disk to store label information that is read from job control statements or commands. Synonymous with *label area.*

Language Environment for z/VSE.   An IBM software product that is the implementation of Language Environment on the VSE platform.

language translator. A general term for any assembler, compiler, or other routine that accepts statements in one language and produces equivalent statements in another language.

Large DASD. A DASD device that
1. Has a capacity exceeding 64 K tracks and
2. Does not have VSAM space created prior to VSE/ESA 2.6 that is owned by a catalog.

LE/VSE. Short form of Language Environment for z/VSE.

librarian. The set of programs that maintains, services, and organizes the system and private libraries.

library block. A block of data that is stored in a sublibrary.

library directory. The index that enables the system to locate a certain sublibrary of the accessed library.

library member. The smallest unit of a data that can be stored in and retrieved from a sublibrary.

line commands. In VSE/ICCF, special commands to change the declaration of individual lines on your screen. You can copy, move, or delete a line declaration, for example.

linkage editor. A program that is used to create a phase (executable code) from one or more independently translated object modules, from one or more existing phases, or from both. In creating the phase, the linkage editor resolves cross-references among the modules and phases available as input. The program can catalog the newly built phases.

linkage stack. An area of protected storage that the system gives to a program to save status information in a branch or a program call.

link station. In SNA, the combination of hardware and software that allows a node to attach to and provide control for a link.

loader. A routine, commonly a computer program, that reads data or a program into processor storage. See also *relocating loader*.

local shared resources (LSR). A VSE/VSAM option that is activated by three extra macros to share control blocks among files.

lock file. In a shared disk environment under VSE, a system file on disk that is used by the sharing systems to control their access to shared data.

logical partition. In LPAR mode, a subset of the server unit hardware that is defined to support the operation of a system control program.

logical record. A user record, normally pertaining to a single subject and processed by data management as a unit. Contrast with *physical* record, which may be larger or smaller.

logical unit (LU).
1. A name that is used in programming to represent an I/O device address. *physical unit (PU), system services control point (SSCP), primary logical unit (PLU), and secondary logical unit (SLU).*
2. In SNA, a port through which a user accesses the SNA network,
   a. To communicate with another user and
   b. To access the functions of the SSCP. An LU can support at least two sessions. One with an SSCP and one with another LU and might be capable of supporting many sessions with other LUs.

logical unit name. In programming, a name that is used to represent the address of an input/output unit.

logical unit 6.2. A SNA/SDLC protocol for communication between programs in a distributed processing environment. LU 6.2 is characterized by
1. A peer relationship between session partners,
2. Efficient utilization of a session for multiple transactions,
3. Comprehensive end-to-end error processing, and
4. A generic Application Programming Interface (API) consisting of structured verbs that are mapped into a product implementation.

logons interpret interpret routine. In VTAM, an installation exit routine, which is associated with an interpret table entry, that translates logon information. It also verifies the logon.

LPAR mode. Logically partitioned mode. The CP mode that is available on the Configuration (CONFIG) frame when the PR/SM feature is installed. LPAR mode allows the operator to allocate the hardware resources of the processor unit among several logical partitions.

M

macro definition. A set of statements and instructions that defines the name of, format of, and conditions for generating a sequence of assembler statements and machine instructions from a single source statement.

macro expansion. See *macro generation*

macro generation. An assembler operation by which a macro instruction gets replaced in the program by the statements of its definition. It takes place before assembly. Synonymous with *macro expansion*.

macro (instruction).
1. In assembler programming, a user-invented assembler statement that causes the assembler to

process a set of statements that are defined previously in the macro definition.

2. A sequence of VSE/ICCF commands that are defined to cause a sequence of certain actions to be performed in response to one request.

maintain system history program (MSHP). A program that is used for automating and controlling various installation, tailoring, and service activities for a VSE system.

main task. The main program within a partition in a multiprogramming environment.

master console. In z/VSE, one or more consoles that receive all system messages, except for those that are directed to one particular console. Contrast this with the *user* console, which receives only those messages that are specifically directed to it, for example messages that are issued from a job that was submitted with the request to echo its messages to that console. The operator of a master console can reply to all outstanding messages and enter all system commands.

maximum (max) CA. A unit of allocation equivalent to the maximum control area size on a count-key-data or fixed-block device. On a CKD device, the max CA is equal to one cylinder.

memory object. Chunk of virtual storage that is allocated above the bar (2 GB) to be created with the IARV64 macro.

message. In VSE, a communication that is sent from a program to the operator or user. It can appear on a console, a display terminal or on a printout.

MSHP. See maintain system history program.

multitasking. Concurrent running of one main task and one or several subtasks in the same partition.

MVS. Multiple Virtual Storage. Implies MVS/390, MVS/XA, MVS/ESA, and the MVS element of the z/OS (OS/390) operating system.

N

NetView. A z/VSE optional program that is used to monitor a network, manage it, and diagnose its problems.

network address. In SNA, an address, consisting of subarea and element fields, that identifies a link, link station, or NAU. Subarea nodes use network addresses; peripheral nodes use local addresses. The boundary function in the subarea node to which a peripheral node is attached transforms local addresses to network addresses and vice versa. See also *network name*.

network addressable unit (NAU). In SNA, a logical unit, a physical unit, or a system services control point.

It is the origin or the destination of information that is transmitted by the path control network. Each NAU has a network address that represents it to the path control network. See also *network name, network address*.

Network Control Program (NCP). An IBM licensed program that provides communication controller support for single-domain, multiple-domain, and interconnected network capability. Its full name is ACF/NCP.

network definition table (NDT). In VSE/POWER networking, the table where every node in the network is listed.

network name.
1. In SNA, the symbolic identifier by which users refer to a NAU, link, or link station. See also *network address*.
2. In a multiple-domain network, the name of the APPL statement defining a VTAM application program. This is its network name, which must be unique across domains.

node.
1. In SNA, an end point of a link or junction common to several links in a network. Nodes can be distributed to host processors, communication controllers, cluster controllers, or terminals. Nodes can vary in routing and other functional capabilities.
2. In VTAM, a point in a network that is defined by a symbolic name. Synonymous with *network node*. See *major node and minor node*.

node type. In SNA, a designation of a node according to the protocols it supports and the network addressable units (NAUs) it can contain.

O

object module (program). A program unit that is the output of an assembler or compiler and is input to a linkage editor.

online application program. An interactive program that is used at display stations. When active, it waits for data. Once input arrives, it processes it and send a response to the display station or to another device.

operator command. A statement to a control program, issued via a console or terminal. It causes the control program to provide requested information, alter normal operations, initiate new operations, or end existing operations.

optional licensed program. An IBM licensed program that a user can install on VSE by way of available installation-assist support.

output parameter text block (OPTB). in VSE/POWER's spool-access support, information that

is contained in an output queue record if a * $$ LST or * $$ PUN statement includes any user-defined keywords that have been defined for autostart.

P

page data set (PDS). One or more extents of disk storage in which pages are stored when they are not needed in processor storage.

page fixing. Marking a page so that it is held in processor storage until explicitly released. Until then, it cannot be paged out.

page I/O. Page-in and page-out operations.

page pool. The set of page frames available for paging virtual-mode programs.

panel. The complete set of information that is shown in a single display on terminal screen. Scrolling back and forth through panels like turning manual pages. See also *selection panel*.

partition balancing, dynamic. A z/VSE facility that allows the user to specify that two or more or all partitions of the system should receive about the same amount of time on the processor.

PASN-AL (primary address space number - access list). The access list that is associated with a partition. A program uses the PASN-AL associated with its partition and the DU-AL associated with its task (work unit). See also *DU-AL*.

Each partition has its own unique PASN-AL. All programs running in this partition can access data spaces through the PASN-AL. Thus a program can create a data space, add an entry for it in the PASN-AL, and obtain the ALET that indexes the entry. By passing the ALET to other programs in the partition, the program can share the data space with other programs running in the same partition.

PDS. Page data sets.

phase. The smallest complete unit of executable code that can be loaded into virtual storage.

physical record. The amount of data that is transferred to or from auxiliary storage. Synonymous with *block*.

PNET. Programming support available with VSE/POWER; it provides for the transmission of selected jobs, operator commands, messages, and program output between the nodes of a network.

POWER. See *VSE/POWER*.

pregenerated operating system. An operating system such as z/VSE that is shipped by IBM mainly in object code. IBM defines such key characteristics as the size of the main control program, the organization, and size of libraries, and required system areas on disk. The customer does not have to generate an operating system.

preventive service. The installation of one or more PTFs on a VSE system to avoid the occurrence of anticipated problems.

primary address space. In z/VSE, the address space where a partition is executed. A program in primary mode fetches data from the primary address space.

primary library. A VSE library owned and directly accessible by a certain terminal user.

printer/keyboard mode. Refers to 1050 or 3215 console mode (device dependent).

Print Services Facility (PSF)/VSE. An access method that provides support for the advanced function printers.

private area. The virtual space between the shared area (24 bit) and shared area (31 bit), where (private) partitions are allocated. Its maximum size can be defined during IPL. See also *shared area*.

private memory object. Memory object (chunk of virtual storage) that is allocated above the 2 GB line (bar) only accessible by the partition that created it.

private partition. Any of the system's partitions that are not defined as shared. See also *shared partition*.

production library.
1. In a pre-generated operating system (or product), the program library that contains the object code for this system (or product).
2. A library that contains data that is needed for normal processing. Contrast with *test library*.

programmer logical unit. A logical unit available primarily for user-written programs. See also *logical unit name*.

program temporary fix (PTF). A solution or by-pass of one or more problems that are documented in APARs. PTFs are distributed to IBM customers for preventive service to a current release of a program.

PSF/VSE. Print Services Facility/VSE.

PTF. See *Program temporary fix*.

Q

Queue Control Area (QCA). In VSE/POWER, an area of the data file, which might contain:
- Extended checkpoint information
- Control information for a shared environment.

queue file. A direct-access file that is maintained by VSE/POWER that holds control information for the spooling of job input and job output.

R

random processing. The treatment of data without respect to its location on disk storage, and in an arbitrary sequence that is governed by the input against which it is to be processed.

real address area. In z/VSE, processor storage to be accessed with dynamic address translation (DAT) off

real address space. The address space whose addresses map one-to-one to the addresses in processor storage.

real mode. In VSE, a processing mode in which a program might not be paged. Contrast with *virtual mode*.

recovery management support (RMS). System routines that gather information about hardware failures and that initiate a retry of an operation that failed because of processor, I/O device, or channel errors.

refresh release. An upgraded VSE system with the latest level of maintenance for a release.

relative-record file. A VSE/VSAM file whose records are loaded into fixed-length slots and accessed by the relative-record numbers of these slots.

release upgrade. Use of the FSU functions to install a new release of z/VSE.

relocatable module. A library member of the type object. It consists of one or more control sections cataloged as one member.

relocating loader. A function that modifies addresses of a phase, if necessary, and loads the phase for running into the partition that is selected by the user.

remote interface. In the context of z/VSE, the remote interface allows a client to make method calls to an EJB although the EJB is on a remote z/VSE host. The container uses the remote interface to create client-side stubs and server-side proxy objects to handle incoming method calls from a client to an EJB.

remote procedure call (RPC).
1. A facility that a client uses to request the execution of a procedure call from a server. This facility includes a library of procedures and an external data representation.
2. A client request to service provider in another node.

residency mode (RMODE). A program attribute that refers to the location where a program is expected to reside in virtual storage. RMODE 24 indicates that the

program must reside in the 24-bit addressable area (below 16 megabytes), RMODE ANY indicates that the program can reside anywhere in 31-bit addressable storage (above or below 16 megabytes).

REXX/VSE. A general-purpose programming language, which is particularly suitable for command procedures, rapid batch program development, prototyping, and personal utilities.

RMS. Recovery management support.

RPG II. A commercially oriented programming language that is specifically designed for writing application programs that are intended for business data processing.

S

SAM ESDS file. A SAM file that is managed in VSE/VSAM space, so it can be accessed by both SAM and VSE/VSAM macros.

SCP. System control programming.

SDL. System directory list.

search chain. The order in which chained sublibraries are searched for the retrieval of a certain library member of a specified type.

second-level directory. A table in the SVA containing the highest phase names that are found on the directory tracks of the system sublibrary.

Secure Sockets Layer (SSL). A security protocol that allows the client to authenticate the server and all data and requests to be encrypted. SSL was developed by Netscape Communications Corp. and RSA Data Security, Inc..

segmentation. In VSE/POWER, a facility that breaks list or punch output of a program into segments so that printing or punching can start before this program has finished generating such output.

selection panel. A displayed list of items from which a user can make a selection. Synonymous with *menu*.

sense. Determine, on request or automatically, the status or the characteristics of a certain I/O or communication device.

sequential access method (SAM). A data access method that writes to and reads from an I/O device record after record (or block after block). On request, the support performs device control operations such as line spacing or page ejects on a printer or skip some tape marks on a tape drive.

service node. Within the VSE unattended node support, a processor that is used to install and test a master VSE system, which is copied for distribution to

the unattended nodes. Also, program fixes are first applied at the service node and then sent to the unattended nodes.

service program. A computer program that performs function in support of the system. See with *utility program*.

service refresh. A form of service containing the current version of all software. Also referred to as a *system refresh*.

service unit. One or more PTFs on disk or tape (cartridge).

shared area. In z/VSE, shared areas (24 bit) contain the Supervisor areas and SVA (24 bit) and shared areas (31 bit) the SVA (31 bit). Shared areas (24 bit) are at the beginning of the address space (below 16 MB), shared area (31 bit) at the end (below 2 GB).

shared disk option. An option that lets independent computer systems use common data on shared disk devices.

shared memory objects. An option that lets independent computer systems uses common data on shared disk devices.

shared partition. In z/VSE, a partition that is allocated for a program (VSE/POWER, for example) that provides services and communicates with programs in other partitions of the system's virtual address spaces.

shared spooling. A function that permits the VSE/POWER account file, data file, and queue file to be shared among several computer systems with VSE/POWER.

shared virtual area (SVA). In z/VSE, a high address area that contains a list system directory list (SDL) of frequently used phases, resident programs that are shared between partitions, and an area for system support.

SIT (System Initialization Table). A table in CICS that contains data used the system initialization process. In particular, the SIT can identify (by suffix characters) the version of CICS system control programs and CICS tables that you have specified and that are to be loaded.

skeleton. A set of control statements, instructions, or both, that requires user-specific information to be inserted before it can be submitted for processing.

socksified. See *socks-enabled*.

Socks-enabled. Pertaining to TCP/IP software, or to a specific TCP/IP application, that understands the *socks protocol*. "Socksified" is a slang term for socks-enabled.

socks protocol. A protocol that enables an application in a secure network to communicate through a firewall via a *socks server*.

socks server. A circuit-level gateway that provides a secure one-way connection through a firewall to server applications in a nonsecure network.

source member. A library member containing source statements in any of the programming languages that are supported by VSE.

split. To double a specific unit of storage space (CI or CA) dynamically when the specified minimum of free space gets used up by new records.

spooling. The use of disk storage as buffer storage to reduce processing delays when transferring data between peripheral equipment and the processor of a computer. In z/VSE, this is done under the control of VSE/POWER.

Spool Access Protection. An optional feature of VSE/POWER that restricts individual spool file entry access to user IDs that have been authenticated by having performed a security logon.

spool file.
1. A file that contains output data that is saved for later processing.
2. One of three VSE/POWER files on disk: queue file, data file, and account file.

stacked tape. An IBM supplied product-shipment tape containing the code of several licensed programs.

standard label. A fixed-format record that identifies a volume of data such as a tape reel or a file that is part of a volume of data.

stand-alone program. A program that runs independently of (not controlled by) the VSE system.

startup. The process of performing IPL of the operating system and of getting all subsystems and applications programs ready for operation.

start option. In VTAM, a user-specified or IBM specified option that determines conditions for the time a VTAM system is operating. Start options can be predefined or specified when VTAM is started.

static partition. A partition, which is defined at IPL time and occupying a defined amount of virtual storage that remains constant. See also *dynamic partition*.

storage director. An independent component of a storage control unit; it performs all of the functions of a storage control unit and thus provides one access path to the disk devices that are attached to it. A storage control unit has two storage directors.

storage fragmentation. Inability to allocate unused sections (fragments) of storage in the real or virtual address range of virtual storage.

suballocated file. A VSE/VSAM file that occupies a portion of an already defined data space. The data space might contain other files. See also *unique file*.

sublibrary. In VSE, a subdivision of a library. Members can only be accessed in a sublibrary.

sublibrary directory. An index for the system to locate a member in the accessed sublibrary.

submit. A VSE/POWER function that passes a job to the system for processing.

SVA. See shared virtual area.

Synchronous DataLink Control (SDLC). A discipline for managing synchronous, code-transparent, serial-by-bit information transfer over a link connection. Transmission exchanges might be duplex or half-duplex over switched or non-switched links. The configuration of the link connection might be point-to-point, multipoint, or loop.

SYSRES. See system residence volume.

system control programming (SCP). IBM supplied, non-licensed program fundamental to the operation of a system or to its service or both.

system directory list (SDL). A list containing directory entries of frequently used phases and of all phases resident in the SVA. The list resides in the SVA.

system file. In z/VSE, a file that is used by the operating system, for example, the hardcopy file, the recorder file, the page data set.

System Initialization Table (SIT). A table in CICS that contains data that is used by the system initialization process. In particular, the SIT can identify (by suffix characters) the version of CICS system control programs and CICS tables that you have specified and that are to be loaded.

system recorder file. The file that is used to record hardware reliability data. Synonymous with *recorder file*.

system refresh. See *service refresh*.

system refresh release. See *refresh release*.

system residence file (SYSRES). The z/VSE system sublibrary IJSYSRS.SYSLIB that contains the operating system. It is stored on the system residence volume DORSES.

system residence volume (SYSRES). The disk volume on which the system sublibrary is stored and from which the hardware retrieves the initial program load routine for system startup.

system sublibrary. The sublibrary that contains the operating system. It is stored on the system residence volume (SYSRES).

T

task management. The functions of a control program that control the use, by tasks, of the processor and other resources (except for input/output devices).

time event scheduling support. In VSE/POWER, the time event scheduling support offers the possibility to schedule jobs for processing in a partition at a predefined time once repetitively. The time event scheduling operands of the * $$ JOB statement are used to specify the wanted scheduling time.

track group. In VSE/POWER, the basic organizational unit of a file for CKD devices.

track hold. A function that protects a track that is being updated by one program from being accessed by another program.

transaction.
1. In a batch or remote batch entry, a job or job step. 2. In CICS TS, one or more application programs that can be used by a display station operator. A given transaction can be used concurrently from one or more display stations. The execution of a transaction for a certain operator is also referred to as a task.
2. A given task can relate only to one operator.

transient area. An area within the control program that is used to provide high-priority system services on demand.

Turbo Dispatcher. A facility of z/VSE that allows to use multiprocessor systems (also called CEC: Central Electronic Complexes). Each CPU within such a CEC has accesses to be shared virtual areas of z/VSE: supervisor, shared areas (24 bit), and shared areas (31 bit). The CPUs have equal rights, which means that any CPU might receive interrupts and work units are not dedicated to any specific CPU.

U

UCB. Universal character set buffer.

universal character set buffer (UCB). A buffer to hold UCS information.

user console. In z/VSE, a console that receives only those system messages that are specifically directed to it. These are, for example, messages that are issued from a job that was submitted with the request to echo its messages to that console. Contrast with *master console*.

user exit. A programming service that is provided by an IBM software product that can be requested during the execution of an application program for the service of transferring control back to the application program upon the later occurrence of a user-specified event.

V

variable-length relative-record data set (VRDS). A relative-record data set with variable-length records. See also *relative-record data set*.

variable-length relative-record file. A VSE/VSAM relative-record file with variable-length records. See also *relative-record file*.

VIO. See virtual I/O area.

virtual address. An address that refers to a location in virtual storage. It is translated by the system to a processor storage address when the information stored at the virtual address is to be used.

virtual addressability extension (VAE). A storage management support that fives the user of VSE multiple address spaces of virtual storage.

virtual address space. A subdivision of the virtual address area available to the user for the allocation of private, nonshared partitions.

virtual disk. A range of up to 2 gigabytes of contiguous virtual storage addresses that a program can use as workspace. Although the virtual disk exists in storage, it appears as a real FBA disk device to the user program. All I/O operations that are directed to a virtual disk are intercepted and the data to be written to, or read from, the disk is moved to or from a data space.

Like a data space, a virtual disk can hold only user data; it does not contain shared areas, system data, or programs. Unlike an address space or a data space, data is not directly addressable on a virtual disk. To manipulate data on a virtual disk, the program must perform I/O operations.

virtual I/O area (VIO). An extension of the page data set; used by the system as intermediate storage, primarily for control data.

virtual mode. The operating mode of a program can be paged.

virtual partition. In VSE, a division of the dynamic area of virtual storage.

virtual storage. Addressable space image for the user from which instructions and data are mapped into processor storage locations.

virtual tape. In z/VSE, a virtual tape is a file (or data set) containing a tape image. You can read from or write to a virtual tape in the same way as if it were a physical tape. A virtual tape can be:
- A VSE/VSAM ESDS file on the z/VSE host side.
- A remote file on the server side; for example, a Linux, UNIX, or Windows file. To access such a remote virtual tape, a TCP/IP connection is required between z/VSE and the remote system.

volume ID. The volume serial number, which is a number in a volume label that is assigned when a volume is prepared for use by the system.

VRDS. Variable-length relative-record data sets. See *variable-length relative record file*.

VSAM. See *VSE/VSAM*.

VSE (Virtual Storage Extended). A system that consists of a basic operating system and any IBM supplied and user-written programs that are required to meet the data processing needs of a user. VSE and hardware it controls form a complete computing system. Its current version is called z/VSE.

VSE/Advanced Functions. As part of VSE Central Functions, a base program of z/VSE. A program that provides basic system control and includes the supervisor and system programs such as the Librarian and the Linkage Editor.

VSE Connector Server. Is the host part of the VSE JavaBeans, and is started using the job STARTVCS, which is placed in the reader queue during installation of z/VSE. Runs by default in dynamic class R.

VSE/DITTO (VSE/Data Interfile Transfer, Testing, and Operations Utility). An IBM licensed program that provides file-to-file services for disk, tape, and card devices.

VSE/ESA (Virtual Storage Extended/Enterprise Systems Architecture). The predecessor system of z/VSE.

VSE/Fast Copy. A utility program for fast copy data operations from disk to disk and dump/restore operations via an intermediate dump file on magnetic tape or disk.

VSE/FCOPY (VSE/Fast Copy Data Set program). An IBM licensed program for fast copy data operations from disk to disk and dump/restore operations via an intermediate dump file on magnetic tape or disk. There is also a stand-alone version: the FASTCOPY utility.

VSE/ICCF (VSE/Interactive Computing and Control Facility). An IBM licensed program that serves as interface, on a time-slice basis, to authorized users of terminals that are linked to the system's processor.

VSE/ICCF library. A file that is composed of smaller files (libraries) including system and user data, which can be accessed under the control of VSE/ICCF.

VSE JavaBeans. Are JavaBeans that allow access to all VSE-based file systems (VSE/VSAM, Librarian, and VSE/ICCF), submit jobs, and access the z/VSE operator console. The class library is contained in the *VSEConnector.jar* archive. See also *JavaBeans*.

VSE library. A collection of programs in various forms and storage dumps stored on disk. The form of a program is indicated by its member type such as source code, object module, phase, or procedure. A VSE library consists of at least one sublibrary, which can contain any type of member.

VSE/POWER. An IBM licensed program that is primarily used to spool input and output. The program's networking functions enable a VSE system to exchange files with or run jobs on another remote processor.

VSE/VSAM (VSE/Virtual Storage Access Method). An IBM access method for direct or sequential processing of fixed and variable length records on disk devices.

VSE/VSAM catalog. A file containing extensive file and volume information that VSE/VSAM requires to locate files, to allocate and deallocate storage space, to verify the authorization of a program or an operator to gain access to a file, and to accumulate use statistics for files.

VSE/VSAM managed space. A user-defined space on disk that is placed under the control of VSE/VSAM.

W

wait for run subqueue. In VSE/POWER, a subqueue of the reader queue with dispatchable jobs ordered in execution start time sequence.

wait state. The condition of a processor when all operations are suspended. System recovery from a hard wait is impossible without performing a new system startup. See *hard wait*.

Workstation File Transfer Support. Enables the exchange of data between IBM Personal Computers (PCs) linked to a z/VSE host system where the data is kept in intermediate storage. PC users can retrieve that data and work with it independently of z/VSE.

work file. A file that is used for temporary storage of data being processed.

Numerics

24-bit addressing. Provides addressability for address spaces up to 16 megabytes.

31-bit addressing. Provides addressability for address spaces up to 2 gigabytes.

64-bit addressing. Provides addressability for address spaces up to 2 gigabytes and above. See also *24-bit addressing*.

Index

Numerics

Readers' Comments — We'd Like to Hear from You

IBM z/VSE
System Upgrade and Service
Version 5

Publication No. SC34-2639-01

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments 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 state on this form.

Comments:

Thank you for your support.

Submit your comments using one of these channels:
- Send your comments to the address on the reverse side of this form.
- Send a fax to the following number: +49-7031-163456
- Send your comments via email to: s390id@de.ibm.com
- Send a note from the web page: http://www.ibm.com/systems/z/os/zvse/

If you would like a response from IBM, please fill in the following information:

_____   _____
Name   Address

_____   _____
Company or Organization

_____   _____
Phone No.   Email address

Readers' Comments — We'd Like to Hear from You

SC34-2639-01

IBM®

Fold and Tape | **Please do not staple** | Fold and Tape

BUSINESS REPLY MAIL

FIRST-CLASS MAIL   PERMIT NO. 40   ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Deutschland Research & Development GmbH
Department 3282
Schoenaicher Strasse 220
71032 Boeblingen
Germany

Fold and Tape | **Please do not staple** | Fold and Tape

SC34-2639-01

IBM®

Product Number: 5609-ZV5

Printed in USA