IMS 15

Installation
(2024-08-30 edition)

GC27-6788-00

Note

Before you use this information and the product it supports, read the information in “Notices” on page 159.

2024-08-30 edition.

This edition applies to IMS 15 (program number 5635-A06), IMS Database Value Unit Edition, V15.01.00 (program number 5655-DS5), IMS Transaction Manager Value Unit Edition, V15.01.00 (program number 5655-TM4), and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

About this information

These topics provide guidance information for preparing for an IMS installation process and running the IMS installation verification program (IVP). The topics also describe the sample applications that are provided with IMS.

This information is available in IBM® Documentation.

Prerequisite knowledge

Before using this information, you should have knowledge of either IMS Database Manager (DB) or IMS Transaction Manager (TM). You should also understand basic z/OS® and IMS concepts, your installation's IMS system, and have general knowledge of the tasks involved in project planning.

To learn about z/OS, see z/OS Basic Skills. For more resources, see IBM Z Education and Training.

To learn about IMS, see the IBM Press publication *An Introduction to IMS*, the resources listed for IBM Information Management System, and the variety of options available in IBM Training.

How new and changed information is identified

New and changed information in most IMS library PDF publications is denoted by a character (revision marker) in the left margin. The first edition (-00) of *Release Planning*, as well as the *Program Directory* and *Licensed Program Specifications*, do not include revision markers.

Revision markers follow these general conventions:

- Only technical changes are marked; style and grammatical changes are not marked.
- If part of an element, such as a paragraph, syntax diagram, list item, task step, or figure is changed, the entire element is marked with revision markers, even though only part of the element might have changed.
- If a topic is changed by more than 50%, the entire topic is marked with revision markers (so it might seem to be a new topic, even though it is not).

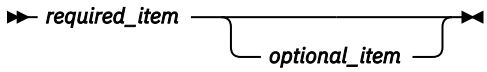
Revision markers do not necessarily indicate all the changes made to the information because deleted text and graphics cannot be marked with revision markers.

How to read syntax diagrams

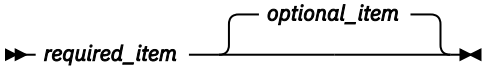
The following rules apply to the syntax diagrams that are used in this information:

- Read the syntax diagrams from left to right, from top to bottom, following the path of the line. The following conventions are used:
  - The >>--- symbol indicates the beginning of a syntax diagram.
  - The ---> symbol indicates that the syntax diagram is continued on the next line.
  - The >--- symbol indicates that a syntax diagram is continued from the previous line.
  - The --->< symbol indicates the end of a syntax diagram.
- Required items appear on the horizontal line (the main path).

- Optional items appear below the main path.

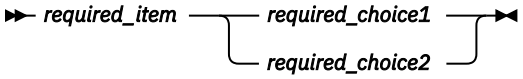


If an optional item appears above the main path, that item has no effect on the execution of the syntax element and is used only for readability.

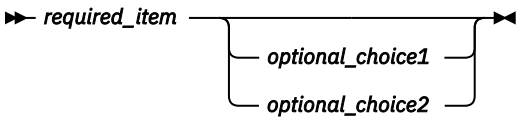


- If you can choose from two or more items, they appear vertically, in a stack.

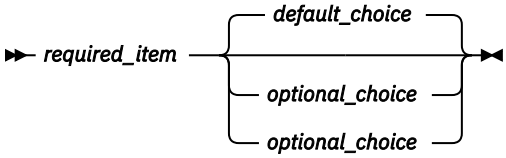
  If you *must* choose one of the items, one item of the stack appears on the main path.



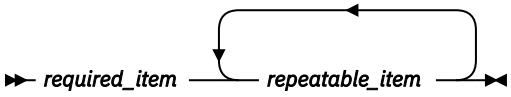
  If choosing one of the items is optional, the entire stack appears below the main path.



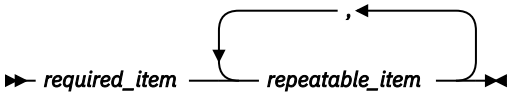
  If one of the items is the default, it appears above the main path, and the remaining choices are shown below.



- An arrow returning to the left, above the main line, indicates an item that can be repeated.

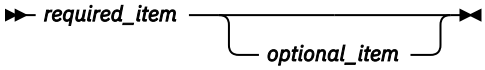


  If the repeat arrow contains a comma, you must separate repeated items with a comma.



  A repeat arrow above a stack indicates that you can repeat the items in the stack.
- Sometimes a diagram must be split into fragments. The syntax fragment is shown separately from the main syntax diagram, but the contents of the fragment should be read as if they are on the main path of the diagram.

  fragment-name



- In IMS, a b symbol indicates one blank position.
- Keywords, and their minimum abbreviations if applicable, appear in uppercase. They must be spelled exactly as shown. Variables appear in all lowercase italic letters (for example, *column-name*). They represent user-supplied names or values.

- Separate keywords and parameters by at least one space if no intervening punctuation is shown in the diagram.
- Enter punctuation marks, parentheses, arithmetic operators, and other symbols, exactly as shown in the diagram.
- Footnotes are shown by a number in parentheses, for example (1).

Accessibility features for IMS 15

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use information technology products successfully.

Accessibility features

The following list includes the major accessibility features in z/OS products, including IMS 15. These features support:

- Keyboard-only operation.
- Interfaces that are commonly used by screen readers and screen magnifiers.
- Customization of display attributes such as color, contrast, and font size.

Keyboard navigation

You can access IMS 15 ISPF panel functions by using a keyboard or keyboard shortcut keys.

For information about navigating the IMS 15 ISPF panels using TSO/E or ISPF, refer to the *z/OS TSO/E Primer*, the *z/OS TSO/E User's Guide*, and the *z/OS ISPF User's Guide Volume 1*. These guides describe how to navigate each interface, including the use of keyboard shortcuts or function keys (PF keys). Each guide includes the default settings for the PF keys and explains how to modify their functions.

Related accessibility information

Online documentation for IMS 15 is available in IBM Documentation.

IBM and accessibility

See the *IBM Human Ability and Accessibility Center* at www.ibm.com/able for more information about the commitment that IBM has to accessibility.

How to send your comments

Your feedback is important in helping us provide the most accurate and highest quality information. If you have any comments about this or any other IMS information, you can take one of the following actions:

- Submit a comment by using the DISQUS commenting feature at the bottom of any IBM Documentation topic.
- Send an email to imspubs@us.ibm.com. Be sure to include the book title and the publication number.
- Click the **Contact Us** tab at the bottom of any IBM Documentation topic.

To help us respond quickly and accurately, please include as much information as you can about the content you are commenting on, where we can find it, and what your suggestions for improvement might be.

Chapter 1. IMS installation overview

Installing IMS includes planning steps, ordering hardware and software, and running the installation verification program (IVP).

Prerequisites

It is assumed that you have experience working with the following products or environments:

- Product installation and service using SMP/E
- The z/OS environment:
  - Job Entry Subsystem (JES2 or JES3)
  - Job Control Language (JCL)
  - Utilities
  - Operations
  - System Display and Search Facility (SDSF)
- The Time Sharing Option (TSO) environment:
  - CLISTs and REXX EXECs
  - Interactive Systems Productivity Facility (ISPF)
- The Virtual Storage Access Method (VSAM) and the Integrated Catalog Facility (ICF)

Installation process

The following categories provide the end-to-end steps and processes that are recommended in order to install IMS and migrate the new version into production.

1. Release planning

   There are several things to consider before you order the software and hardware that is appropriate for a particular business. Most companies have a business plan that is used to address these issues and answer important questions. Some of these questions include:

   - Which release of IMS are you upgrading to?
   - Which release provides enough value to your company to warrant a migration?
   - When do you upgrade to the next release?
   - What are the hardware and personnel requirements that you need in order to upgrade?

   For help with these business decisions, consult the edition of *IMS Release Planning* specific to the version of IMS that you are considering for your migration. For the IMS and DB2® Tools requirements, go to z Systems software website.
2. Ordering the hardware and software

   After you determine your particular needs, you can order the products specific to your business plan. *IMS Version 15 Release Planning*, the product announcement letter, and *Program Directory for Information Management System Transaction and Database Servers* provide the hardware and software requirements specific to each release of IMS. You can order IMS by using the following distribution media:

   - Custom-Built Product Delivery Offering (CBPDO)

     The CBPDO product package consists of one logical tape (multiple volumes). A CBPDO package that includes IMS can also include other products in the same System Release (SREL). CBPDO also provides service for the products included with the product order.

The service includes all PTFs available within one week of order fulfillment. All PTFs are identified by one or more SOURCEIDs, including PUTyymm, RSUyymm, SMCREC, HIPER, and SMCCOR.

See the CBPDO "DBS Memo to User Extensions" (shipped with the CBPDO package) for more information.

- ServerPac

  ServerPac is an entitled software delivery package. It consists of products and service for which IBM performed the SMP/E installation steps and some of the post-SMP/E installation steps. To install the package on your system and complete the installation of the software it includes, use the CustomPac Installation dialog, which is the same dialog that is used for all CustomPac offerings, including SystemPac (dump-by-data-set format), ProductPac®, and RefreshPac.

  ServerPac provides the following:

  – Allocates, catalogs, and loads all the data sets
  – Sets up the SMP/E environment
  – Supplies a job to update PARMLIB (IEFSSNxx, PROGxx, and IEASVCxx)
  – Directs you to start the IVP

  Your ServerPac order includes the latest unintegrated service that is available at the time your order is created. Read the accompanying *ServerPac: Installing Your Order* document for information about the service included in your order and any preventive maintenance that you should perform after the installation. You must review Appendix C, *"Using the Package Reports"* in the document for the SMP/E reports. You might need to resolve FMIDs that are not included in your order, or PTF-in-errors (PEs) that cannot be integrated in your order because no fixes are currently available. Check to see whether the errors affect your system and if fixes are available. If fixes are not available, contact IBM Software Support for assistance.

3. Installation planning

   Before you install IMS, keep in mind the following considerations that are specific to each IMS customer. Determining these items are important to ensure successful installation and subsequent administration of IMS. The installation and planning tasks include:

   - Hardware requirements for the IMS product
   - Software requirements for the IMS product
   - Tools requirements for both IBM and non-IBM tools
   - Migration and compatibility requirements for the current release

   The following lists sources of installation and release planning information:

   - IBM IMS announcement letters (RFAs) on the IMS website
   - *IMS Version 15 Release Planning*
   - Preventative Service Planning (PSP bucket)
   - *Program Directory for Information Management System Transaction and Database Servers*

4. Installing the IMS product

   Installation of IMS and any preventive maintenance uses the SMP/E APPLY and ACCEPT processes. You then run the IVP to define and validate a sample IMS system by specifying your environment options and testing the sample applications.

   Multiple sources of documentation are available that describe installation procedures and currency of IMS maintenance. Some of this information is more current than others. The recommended order to obtain the most current information, from least to most current, is:

   a. CBPDO and ServerPac documentation

      You can get this information from:

      - IBM Software Support: 1-800-879-2755

- Shopz website

b. Preventive Service Planning (PSP bucket) information

You can get this information from:

- IBM Software Support: 1-800-879-2755
- Shopz website
- Preventive Service Planning buckets for mainframe operating environments website

c. *Program Directory for Information Management System Transaction and Database Servers*

You can get this information from:

- CBPDO or ServerPac documentation
- EMEA Order Support website

d. *IMS Version 15 Installation*

It is important to install the available IMS service, which is packaged in SMP/E format, for any new installation of IMS.

5. Running the IVP.

   The installation verification program is provided by IBM to test the product installation of IMS and verify that the major functions and features are working. The jobs and tasks of the IVP build a sample IMS system and provide several sample applications that verify specific components of IMS.

6. Deployment of your own system for test or production.

   After you install IMS and verify the installation by using the IVP, consider how to upgrade the new IMS system into a test system and eventually into production. Each customer has individual requirements and plans for rolling out their new IMS systems.

7. Testing the new test or production system.

   After the new version of the IMS product is implemented, test various functions to verify that the implementation is successful. The specific testing varies for each customer, but should include testing in both online and batch systems (as appropriate).

Related concepts

General planning information for IMS 15 (Release Planning)

Overview of the IMS system definition process (System Definition)

Related tasks

Installing IMS service on a single system (System Administration)

Related reference

DFSIX messages (Installation Verification Program dialog) (Messages and Codes)

G messages (system definition) (Messages and Codes)

Chapter 2. IMS installation verification program (IVP) overview

The installation verification program (IVP) is provided by IBM to test the product installation of IMS and verify that the major functions and features are working.

The jobs and tasks of the IVP build a sample IMS system and provide several sample applications that are used to verify specific components of IMS. Use the IVP to verify that IMS was installed properly and that the major functions and features of IMS are working.

During the IVP process, you run a combination of jobs and perform a set of tasks to create a fully executable sample IMS system. After the jobs and tasks of the IVP are run and completed successfully, you exercise the system by using the sample applications that are either provided by the IVP or downloaded from the IMS website. You submit the IVP jobs and tasks manually, verify the jobs, and perform the tasks that make up the IVP process.

An IVP job is JCL that you run. An IVP task is something that you need to manually perform. The IVP provides online help with step-by-step instructions for these jobs and tasks. It also provides customized JCL, IMS stage 1 system definition input, database descriptions (DBDs), program specification blocks (PSBs), Message Format Services (MFSs), and application programs. Some of these application programs are run through the IVP jobs and tasks.

Related concepts

IMS Syntax Checker (System Definition)

The IVP as an educational tool

In addition to an installation verification program, the IVP can assist you in learning the IMS system.

After the IVP jobs and tasks specific to your environment are completed, system programmers, application developers, and computer operators can use the IVP sample system, the sample applications, and the jobs and tasks as a training vehicle. In addition, you can use the IVP jobs and tasks to determine the steps that are needed for the deployment of an IMS application development environment, test, or production system.

The IVP can be used as a repeatable training environment and to provide guidance for performing the following types of jobs and tasks:

- Performing an IMS system definition (SYSDEF).
- Establishing IMS interfaces to z/OS and VTAM®.
- Preparing an IMS application system.
- Operating an IMS online system (for example, DBCTL, DB/DC, DB/DC with XRF, and DCCTL).
- Operating sequences that demonstrate both normal and emergency restarts.
- Demonstrating the conversion of a non-HALDB database to a HALDB database.
- Demonstrating the use of TSO Single Point of Control (SPOC).
- Demonstrating that the database resource adapter (DRA) interface module was assembled and placed in the IMS.SDFSRESL data set.
- Providing and demonstrating the setup and use of sample JMP and JBP applications.
- Providing and demonstrating the setup and use of parallel RECON access.
- Providing and demonstrating the setup and use of IMS Connect.
- Providing and demonstrating the setup and use of IMS Open Database.
- Providing and demonstrating the setup and use of Dynamic Resource Definition (DRD).
- Providing and demonstrating the setup and use of the IMSRSC repository.

- Providing and demonstrating the setup and use of the IMS Catalog.

Phases of the IVP process

The IVP process consists of four phases: initialization, variable-gathering, file-tailoring, and execution.

To verify that the new functions and features of IMS are working properly, you must complete all four of these phases. Initially, you must proceed through the phases sequentially. After you complete the variable-gathering and file-tailoring phases, however, you can move back and forth between these two phases out of sequence.

Initialization phase

The initialization phase begins each time you start the IVP dialog or change an option or suboption.

The IVP is driven from a set of ISPF tables that contain information about the variables, jobs, tasks, and the sequence of those jobs and tasks you need to execute. Because the tables are updated by the IVP dialog, working copies must be made the first time you use the IVP or when you change options later. This process is known as a table merge.

During the initialization phase, you can:

- Restart the dialog
- Select environment options
- Select suboptions
- Merge tables
- Select another phase (variable-gathering, file-tailoring, or execution)

Table merge process

A table merge is necessary the first time you run the IVP and any time you change existing environment options or suboptions that was not previously selected, and whenever the installation of service requires it.

The IVP is shipped with master tables in the IMS target and distribution libraries. The master tables contain all the possible IVP variables that can be used to create jobs and tasks. You select the environment and suboptions that are appropriate for your environment. The table merge process populates a set of customized tables from the master tables with the IVP variables, jobs, and tasks that are necessary for you to run the IVP based on your selected environment option and suboptions. Because the dialog updates these tables during its processing, the master tables must be copied into a user data set, INSTATBL. The table merge process performs this copy.

The table merge process is also used to update the INSTATBL data set with updates introduced with PTF service. The service contains a ++HOLD with a reason of ACTION when you must rerun the table merge process.

Optionally, you can run a table merge to reset the ! indicator that is displayed on the phase panels, such as the Execution phase (LST mode) panel. The table merge process does not change variable values that were changed by the copy-startup-variables process or by the CHG action in the variable-gathering phase.

Related concepts

Copy-startup-variable process

After the table-merge process is complete or bypassed, the dialog compares the startup variables with their corresponding table values.

Copy-startup-variable process

After the table-merge process is complete or bypassed, the dialog compares the startup variables with their corresponding table values.

If the table value is different and was not changed by a prior copy-startup-variables process or by the CHG action in the variable-gathering phase, the table value is updated with the startup value. This process is provided so you do not need to enter the same information several times.

The variables affected by this process are:

- The IVP data set high-level qualifier (HLQ) passed to the startup CLIST
- The DLB data set HLQ passed to the startup CLIST
- The SYS data set HLQ passed to the startup CLIST
- The current TSO user ID

  This user ID is used for the USER and NOTIFY job statement parameters.

Related concepts

Table merge process

A table merge is necessary the first time you run the IVP and any time you change existing environment options or suboptions that was not previously selected, and whenever the installation of service requires it.

Variable-gathering phase

In the variable-gathering phase, you choose the options that are used to produce the jobs and tasks necessary in the subsequent phases of the IVP, such as file-tailoring.

The user-modifiable variables that you use during customization (done in the next file-tailoring phase) of the installation materials are presented for review and modification. In this phase, you can perform the following functions:

- Modify the value that is associated with each variable.
- Refresh a variable to its distributed default value.
- View the online descriptions of the variables.
- Print the online help for variables to the ISPF list data set.
- Import variables from a previous release of IMS or another copy of IMS 15.
- Export variables from an installed release of IMS to either the same or the next release of IMS to be installed, to ease migration. For example, if you are currently using IMS Version 10, and want to migrate to IMS 15, you can export the variables from IMS Version 10 for use in IMS 15.

File-tailoring phase

The file-tailoring phase uses the ISPF file-tailoring services to combine the variables from the variable-gathering phase with skeletons from SDFSSLIB to create members (JCL and other materials) in INSTALIB.

The jobs, tasks, and INDEX items that are presented during the file-tailoring phase are specific to the selections that you made during initialization. The jobs and tasks are presented in the order in which they are to be performed. In addition to creating INSTALIB members, this phase serves as a directory for the various members of INSTALIB, SDFSSLIB, and SDFSISRC.

During the file-tailoring phase, you can perform the following functions:

- File tailor all or selected items.
- Browse INSTALIB, SDFSSLIB, or SDFSISRC members.
- Edit INSTALIB members.

- View the online help of the members.
- Print the online help for jobs, tasks, and INDEX items to the ISPF list data set.

Execution phase

The execution phase guides you step by step through the jobs and tasks that are necessary to complete the building and running of the IVP system that is based on options that you chose.

Only the jobs and tasks specific to the selections that you made during initialization are presented. The jobs and tasks are presented in the order in which they are to be performed.

During the execution phase, you can perform the following functions:

- Browse INSTALIB members.
- Edit INSTALIB members. Jobs can be submitted for execution from within edit mode.
- Submit INSTALIB members for execution. Successful job execution must be manually verified.
- File tailor an individual member.
- View the online help of the jobs and tasks.
- Print the online help for jobs and tasks to the ISPF list data set.
- Perform special processing routine setup for a task.

IVP output

The IVP system provides a rich and diverse set of jobs and tasks that, after completion, provide a fully functional IMS system that is integrated into the z/OS environment and tested in the selected environment from the IVP panels.

The IMS environments that you can select include BATCH, DBCTL, DB/DC, DB/DC with XRF, and DCCTL. Most of the major functions of IMS can be demonstrated and tested by using the IVP system. The IVP builds a viable sample IMS system in a controlled manner that is verifiable, robust, and accomplishes the following:

- Verifies that the IMS product itself, the maintenance, or both were successfully installed
- Implements and tests the z/OS and VTAM interfaces
- Builds and integrates IMS application systems
- Tests various IMS application systems
- Tests various functions and features selected
- Assembles the database resource adapter (DRA) interface module, which is used by DBCTL and Open Database Access (ODBA), and places the module in IMS.SDFSRESL

Examples of some of the functions and features that are demonstrated and tested include:

- Syntax Checker
- XRF
- IRLM
- Fast Path
- Shared Queues (Common Queue Server)
- High Availability Large Database (HALDB)
- IMS Connect
- IMS system restart and recovery
- Common Service Layer
- Enhanced Command Environment
- IMS DB resource adapter (previously known as the IMS JDBC Connector)

- Dynamic resource definition (DRD)
- IMS Open Database
- IMSRSC repository
- IMS Catalog

Related concepts

“Sample applications provided by the IVP” on page 25
The sample applications provided by the IVP verify the IMS product installation by using an IVP sample system.

CICS: Installing DBCTL, and defining CICS and IMS system resources

Related tasks

“Building a verifiable working sample IMS system by using the IVP” on page 11
You can use the IVP to build a verifiable working sample IMS system that helps you verify your installation and confirm that your IMS system is operational. You must complete several steps and processes before you can build a verifiable working sample IMS.

Accessing IMS databases through the ODBA interface (Communications and Connections)

Related reference

“IVP jobs and tasks” on page 97
This section identifies all the jobs and tasks that can be used during the IVP process.

Chapter 3. Building a verifiable working sample IMS system by using the IVP

You can use the IVP to build a verifiable working sample IMS system that helps you verify your installation and confirm that your IMS system is operational. You must complete several steps and processes before you can build a verifiable working sample IMS.

Prerequisites:

- Complete the product installation of IMS by following the instructions that come in the packaging offering that you selected (CBPDO or ServerPac).
- Ensure that the corresponding FMIDs for your environments are installed during IMS product installation using SMP/E. For example, if you use IRLM, ETO feature, or IMS Java™ On Demand features, the corresponding FMIDs must be installed.

Related concepts

“IVP output ” on page 8
The IVP system provides a rich and diverse set of jobs and tasks that, after completion, provide a fully functional IMS system that is integrated into the z/OS environment and tested in the selected environment from the IVP panels.

Starting the IVP dialog

Start the IVP dialog by issuing an EXEC command from either an ISPF dialog or the IMS application menu.

Starting the IVP by using the EXEC command from within ISPF

You can start the IVP dialog from within ISPF either by using partial syntax with a simple command or by using full syntax.

To start the IVP dialog using partial syntax:

1. Open an ISPF application dialog.
2. Issue the following TSO EXEC command in the ISPF panel, option 6:

```
------------------  TSO COMMAND PROCESSOR  ------------------
ENTER TSO COMMAND OR CLIST BELOW:
===> EXEC 'qqq.SDFSCLST(DFSIXC01)' 'HLQ(qqq)'
```

Figure 1. Simple command to start the IVP dialog from an ISPF panel

qqq is the high-level qualifier for the IVP, system, and distribution libraries.

You might need to use the full syntax invocation to specify additional parameters for your environment. The full syntax method lets you invoke the IVP startup CLIST and use the IVP system parameters, either the full set or a subset. Use the full syntax method, for example, if you need to specify different high-level qualifiers for the tables and other IVP systems data sets, or invoke the DEBUG parameter for diagnostic purposes. The IVP dialog dynamically allocates the data sets that are needed to support dialog processing; therefore, you do not need to put the IMS ISPF data sets in your TSO logon procedure.

After the command is invoked, the IMS welcome panel displays, followed by the IBM copyright panel. Press Enter to go to the IVP Environment Options panel.

Related reference

"TSO EXEC command syntax for invoking the IVP start-up CLIST" on page 156
The following syntax diagram illustrates how to invoke the IVP start-up CLIST using the TSO EXEC command.

Starting the IVP from the IMS Application Menu

You can start the IVP dialog from the IMS Application Menu.

To start the IMS Application Menu:

1. Open an ISPF application dialog.
2. Start the IMS Application Menu by issuing the following TSO EXEC command:

```
EXEC 'qqq.SDFSEXEC(DFSAPPL)' 'HLQ(qqq)'
```

   The IMS Application Menu opens:

```
  Help
-------------------------------------------------------------
                  IMS Application Menu
COMMAND ===>_____________________________________________________

 Select the desired application and press Enter.
   1 Single Point of Control (SPOC)
   2 Manage resources
   3 Reserved for future use
   4 HALDB Partition Definition utility (PDU)
   5 Syntax Checker for IMS parameters (SC)
   6 Installation Verification Program (IVP)
   7 IVP Export utility (IVPEX)
   8 IPCS with IMS Dump Formatter (IPCS)
   9 Abend Search and Notification (ASN)
```

   Figure 2. IMS Application Menu

3. In the IMS Application Menu, select Option 6 to start the IVP.

The IVP Environment Options panel displays.

Related tasks

"Selecting the environment options" on page 12
Select the options that apply to your environment. The IVP provides suboptions and tasks based on your choices to build a sample IMS system for installation verification.

Related reference

"REXX EXEC command syntax for starting the IMS Application Menu" on page 157
The following syntax applies to the REXX EXEC command for starting the IMS Application Menu.

Starting the IVP initialization phase

During the IVP initialization phase, you select the installation option and suboption values that the IVP uses to build customized tables of the specific jobs and tasks that need to be run.

These tables provide the input for the phases that follow. In addition, some variables are initialized in this phase in preparation for the variable-gathering phase.

Selecting the environment options

Select the options that apply to your environment. The IVP provides suboptions and tasks based on your choices to build a sample IMS system for installation verification.

The following figure shows the IVP Environment Options panel. This panel is referred to as the primary option menu for the IVP dialog.

```
IVP                IVP Environment Options                  IMS 15.1
Command ===>
   DFSIX023: DFSIXX01 - Prior session completed successfully for “DBB”
   Select the desired option and press ENTER
Option. .
    IVP Environments
  1.  DBB - Database Management (Batch)
  2.  DBC - Database Management (DBCTL)
  3.  DBT - Database and Transaction Management (DB/DC)
  4.  XRF - DB/DC with Extended Recovery Facility (DB/DC with XRF)
  5.  DCC - Transaction Management (DCCTL)
```

Figure 3. IVP Environment Options panel

To select an IVP environment option:

1. In the IVP Environment Options panel, type the number of the option you want.

   Each option in the environment options panel (except option 5) includes the options listed before it. For example, if you select option 3, you are building the IMS batch, DBCTL, and DB/DC IVP environments. Select the highest number that represents the system you want to build. Option 5 does not build the environments of options 1, 2, 3, and 4.

   The IVP Environment Options panel supports the following primary options:

   a. DBB - IMS batch environment

      This environment supports batch job access of IMS full-function databases. It can also be used to support Db2 for z/OS applications.

   b. DBC - IMS DBCTL environment

      This environment supports the online access of IMS full-function databases and DEDBs with batch-oriented BMPs. It can also be used as the basis for supporting IBM CICS® Transaction Server for z/OS/DBCTL, ODBA, Db2 for z/OS, batch, and other applications. This environment includes all the function of the DBB environment.

   c. DBT - IMS DB/DC environment

      This environment supports the online access of IMS full-function databases, DEDBs, and MSDBs. IMS DB/DC is a full IMS Transaction and Database Management environment supporting both message-driven and batch-oriented applications. It can also be used for supporting the CICS/DBCTL, ODBA, Db2 for z/OS, batch, and other applications. This environment includes all the function of the DBB and DBC environments.

   d. XRF - IMS XRF environment

      This option extends the DBT (DB/DC) environment to include XRF support. A single CPC configuration (active and alternate IMS subsystems on the same CPC) is used. It can also be used to support the IMS TM environment, CICS/DBCTL, ODBA, Db2 for z/OS, and batch applications. This environment includes all the function of the DBB, DBC, and DBT environments.

   e. DCC - IMS DCCTL environment

      IMS DCCTL is a full IMS Transaction Management environment that supports both message-driven and batch-oriented applications. It can be used as the basis for supporting Db2 for z/OS applications.

2. After you select an option, press **Enter** to continue.

   If you previously ran the IVP dialog and made a selection in the IVP Environment Options panel, the Environment Option Change Verification panel opens.

   If you did not previously run the IVP dialog, the Sub-option Selection panel opens. Skip the next topic on “Verifying an environment option change” on page 14 and proceed directly to “Selecting suboptions” on page 14.

Related tasks

“Starting the IVP from the IMS Application Menu” on page 12

You can start the IVP dialog from the IMS Application Menu.

Related reference

“IVP environment options” on page 131
During the initialization phase of the IVP process, you select the environment options specific to the needs of your business.

Verifying an environment option change

When you select an environment option that you did not select before, the Environment Option Change Verification panel opens.

The following figure shows that the new option XRF is selected and that the last selected option was DBB.

```
  Help
----------------------------------------------------------------
IVP -- Environment Option Change Verification - XRF--IMS 15.1
COMMAND ===>

 The Environment Option you have just chosen is not the same as
 the Option which was last active:


       XRF - Requested Option

       DBB - Previous Option


 To confirm your change of Options to XRF :  Press ENTER

 To return to the Environment Option Selection menu:  Press END
```

Figure 4. Environment Option Change Verification panel

To verify an environment option change:

1. Review the panel contents and requested option change.
2. If the requested option change is correct, press **Enter** to confirm your selection.

If the requested option is not correct, press **End** to return to the Environment Option Selection panel.

Related tasks

“Selecting the environment options” on page 12
Select the options that apply to your environment. The IVP provides suboptions and tasks based on your choices to build a sample IMS system for installation verification.

Selecting suboptions

Choose the suboptions that you want to add to your primary option selection.

Suboptions specify whether you want to use IRLM, Fast Path, and other IMS functions and features. Ensure that the corresponding FMIDs for selected suboptions are installed during IMS product installation using SMP/E.

The following figure depicts the IVP Sub-Options Selection panel of the IVP dialog.

```
  Help
-------------------------------------------------------------
 IVP -------- Sub-Option Selection - XRF ------- IMS 15.1
 COMMAND ===>

    Select the desired Sub-Options and press ENTER
 /  IRLM -  Use IRLM in IVP Applications
 /  FP -  Use Fast Path in IVP Applications
 /  ETO Feature Installed
    CQS - Add CQS to CSL Application
    RACF - Use RACF Security
    JAVA - Use JAVA Applications and Open Database
    PRA - Use Parallel RECON Access
    ICON - Use IMS Connect
    REPO - Use IMSRSC Repository
    COUT - Use Callout Applications

    NOTE:  Your Sub-Option selection affects the user variables,
    jobs, and tasks that will be presented.  If you later change
    your selection, you must redo the IVP Table Merge, Variable
    Gathering, File Tailoring, and Execution processes.
```

Figure 5. IVP Sub-Option Selection panel

To select a suboption:

1. In the panel, type a forward slash (/) next to the suboptions that you want to select. The supported suboptions are:
   - Use the initial resource lock manager (IRLM) in IVP Applications.

     If you select IRLM, the IVP creates a configuration for the IRLM.

     – The default is to use this suboption for DB batch, DBCTL, DB/DC, and DB/DC with XRF. This suboption is not available for DCCTL.

     – If you select this suboption, the IRLM is defined during system definition, and the IVP is run using the IRLM for the single-lock manager. If you do not select this suboption, the IRLM is not used, and program isolation (PI) is used as the single-lock manager.

     – Use of IRLM is required only if you plan to use block-level data sharing. The IVP is configured to support block-level data sharing. Optionally, you can elect to use the IRLM, instead of PI, as the single-lock manager.
   - Use Fast Path in IVP Applications

     If you select this option, the IVP adds the necessary jobs and tasks for the Fast Path sample application. The default is to use this suboption for DBCTL, DB/DC, and DB/DC with XRF. This suboption is not available for DCCTL.
   - ETO feature Installed

     If you select this option, the IVP adds the necessary jobs and tasks for IMS Extended Terminal Option (IMS ETO).

     – For DB/DC and DB/DC with XRF, the default is to use this suboption. This suboption is not available for DB batch or DBCTL.

     – For DCCTL, the default is not to use this suboption.
   - Add CQS to CSL Application

     If you select this option, the IVP adds the necessary jobs and tasks so that the CSL sample application uses CQS.

     The default is not to use this suboption.
   - Use RACF® Transaction Security

     If you select this suboption, the IVP builds the necessary jobs and tasks to define resources to RACF and to set up the use of several IMS security user exit routines.

     – The default is not to use this suboption.

- This suboption is not available for DB batch.
- You can modify the sample RACF resource definition task.
- The sample user exit routine always authorizes the user to the resources.

- Use Java Applications and Open Database

  If you select this suboption, the IVP adds the necessary jobs and tasks for the following:

  a. The execution of sample applications for the IMS solutions for Java development and for setting up the database used for the Java sample applications.

  b. Starting the sample Open Database application.

  The default is not to use this suboption.

  This suboption is not available for DB batch or DCCTL environments.

- Use Parallel RECON Access

  If you select this suboption, the IVP adds the necessary jobs and tasks for starting, initializing, tailoring, executing, and verifying the related services and components for parallel RECON access.

- Use IMS Connect

  If you select this suboption, the IVP adds the necessary jobs and tasks for starting the sample IMS Connect application.

- Use IMSRSC repository

  If you select this suboption, the IVP adds the necessary jobs and tasks for setting up the environment for running the IMSRSC repository.

- Use Callout Applications

  If you select this suboption, the IVP adds the necessary jobs and tasks for setting up the environment for running the callout samples, including the OTMA destination descriptor that is required to route the callout messages.

2. After you select the appropriate suboptions or accept the default suboptions that are displayed, press **Enter**.
3. The Table Merge Request panel is displayed, and you have the option of performing a table-merge to create a table that contains your selected options and tasks.

   If you change the selections that are displayed, the Sub-Option Change Verification panel opens. The dialog asks you to confirm your request for change. If you are changing the selections after you have completed the table-merge, variable gathering, file-tailoring, or execution phases, you must rerun the jobs and tasks in those phases.

Requesting a table merge

After you select an environment option and suboptions, the IVP dialog gives you the option of performing a table-merge.

To request a table merge:

1. In the Table Merge Request panel, type 1 and press **Enter**.

   While the table merge is in progress, the Table Merge in Progress panel opens and the keyboard is locked. This panel is updated as the tables are updated.

   The following figure shows an example of the Table Merge Process Indicator panel.

```
IVP ----- FT Table Merge In Progress - XRF ---- IMS 15.1


                     Table Merge Progress Indicator

                  Variable Gathering Table:  DFSIXBV1
                     Current row . . . . .:  Done.......
                     Percent completed . .:  100

                  File Tailoring Table . .:  DFSIXBF1
                     Current row . . . . .:  DFSIXS01
                     Percent completed . .:  19

                  Execution Table. . . . .:  DFSIXBE1
                     Current row . . . . .:  Patience...
                     Percent completed . .:  000



                  Please do not interrupt this process
```

Figure 6. Table merge progress indicator panel

2. After the table merge process completes, the Table Merge Completed panel is displayed. Press **Enter** to continue.

Related concepts

“Table merge process” on page 6

A table merge is necessary the first time you run the IVP and any time you change existing environment options or suboptions that was not previously selected, and whenever the installation of service requires it.

Selecting an IVP phase and positioning option

Select an IVP phase and choose to start or restart from either the beginning of an IVP phase or from the last known location within a phase.

The following figure shows the IVP Phase Selection panel of the IVP dialog.

```
  Help
-------------------------------------------------------------------------------
IVP                   IVP Phase Selection - XRF            IMS 15.1
COMMAND ===>

Select the desired Phase and positioning option and press ENTER

1_  1. Variable Export Utility (Export variables to a data set)

    VG - Variable Gathering-(Define user values for variables)
    2.  VG1 Start/Restart from the beginning of the phase
    3.  VG2 Start/Restart from the last known position within the phase

    FT - File Tailoring - (Create customized INSTALIB members)
    4.  FT1 Start/Restart from the beginning of the phase
    5.  FT2 Start/Restart from the last known position within the phase
    6.  FT3 Start/Restart from the beginning of a selected step

    EX - Execution - (Run the IVP jobs)
    7.  EX1 Start/Restart from the beginning of the phase
    8.  EX2 Start/Restart from the last known position within the phase
    9.  EX3 Start/Restart from the beginning of a selected step
```

Figure 7. IVP Phase Selection panel for an IMS XRF environment

Tips:

- The dialog always preselects a default. If you just completed the initialization phase, the default is to start from the beginning of the variable gathering phase. You can override the dialog's selection with your own by typing over the default selection.

- You must perform the variable gathering, file tailoring, and execution phases in sequence. However, you can exit from each phase and return to the IVP Phase Selection panel to select the next phase or return to a prior phase.

To select an IVP phase and position within a phase, in the IVP Phase Selection panel, type the number that is associated with the phase and location that you want to execute.

If you make an invalid phase selection, a notification panel opens to inform you of the error. Press **Enter** to return to the Phase Selection panel and type the appropriate selection.

Gathering variables

Gathering variables involves changes to prepare the JCL and other materials that are necessary for further customization in the file-tailoring phase.

When you enter the variable gathering phase, the IVP panel displays the variables based on your selections in the initialization phase. These variables are later used by the file-tailoring phase to customize the IVP to your environment and to create members in the INSTALIB data set. You can import variables from an earlier iteration of the IVP dialog by using the IVP Variable Gathering Export and Import facilities.

To complete variable-gathering:

1. Optional: Import variables from a previous IMS installation and verification with the IVP.
2. In the IVP Phase Selection panel, select option **1** or option **2**. Each selection within a phase provides a different positioning option and opens the Variable Gathering panel.
3. In the Variable Gathering (LST mode) panel, review the displayed variables. Use the display modes and action commands to make appropriate updates. You can browse, display variable descriptions, or edit any of the members on this panel. You can scroll multiple pages of variables by pressing PF7 and PF8. However, you cannot search for a particular variable. Press End at any time to return the IVP Phase Selection panel and save your updates.
4. When you are finished, press PF3 and then Enter to exit the phase.

The IVP Phase Selection panel displays and you can progress to the file-tailoring phase.

Related tasks

“Exporting and importing IVP variables” on page 18
Use the IVP Variable Export utility to export previously used IVP variables to a sequential data set so you can later import them.

Related reference

“Variable gathering dialog options” on page 133
Use the action commands and modes available for the variable-gathering phase.

Exporting and importing IVP variables

Use the IVP Variable Export utility to export previously used IVP variables to a sequential data set so you can later import them.

IVP variables can be exported and imported between IMS releases or between different IVP dialog sessions of the same IMS release. Use the IVP Variable Export utility, shown in Figure 8 on page 19, to copy or export a set of previously used IVP variables to a sequential data set. This data set can be subsequently imported to the IVP tables data set of the target IVP session.

To export variables from one IVP session and import them to the target IVP session:

1. Launch the IVP Variable Export utility by issuing the DFSIVPEX command from an ISPF panel.

   Tip: You can use the ISPF split screen capability to invoke the IVP Variable Export utility without exiting the IVP.

   a. Open an ISPF application dialog.

   b. Issue the following TSO EXEC command:

```
EXEC 'qqq.SDFSEXEC(DFSIVPEX)' 'HLQ(qqq)'
```

qqq is the high-level qualifier for the IMS system (SYS) libraries. The default is IVPSYS13. HLQ(*qqq*) identifies the high-level qualifier for the system libraries.

The IVP Variable Export Utility panel opens. The following figure shows the IVP Variable Export utility panel.

```
                   IVP Variable Export Utility
Command ===>

Enter the following information, then press enter.

_ 1. Select the IVP Environment
     1.  DBB - Database Management (Batch)
     2.  DBC - Database Management (DBCTL)
     3.  DBT - Database and Transaction Management (DB/DC)
     4.  XRF - DB/DC with Extended Recovery Facility (DB/DC with XRF)
     5.  DCC - Transaction Management (DCCTL)

  2. Specify the IVP High Level Qualifier (HLQ) of the INSTATBL data set
     ________________________________________________

  3. Specify the export data set. For a PDS, include the member name.
 If the dataset does not exist, you will be prompted to create the dataset.
     ________________________________________________
```

Figure 8. IVP Variable Export utility panel

Tip: You can also launch the IVP Variable Export utility by using one of the following methods:

- Select the export (**Exp**) action command on the Variable Gathering (LST mode) panel.
- Select option A in the Phase Selection panel.
- Select the IVP Variable Export Utility option from the IMS Application menu.

2. Provide the following information in the IVP Variable Export utility panel:
   a. Select the environment option. Use the same option that you selected during the Initialization phase of the IVP process. The environment option identifies which variables to export because the variables for each environment option are different.
   b. Select IVP High Level Qualifier (HLQ), which identifies the IVP table data set (INSTATBL) from which you are exporting the variables.
   c. Type the name of the export data set in TSO format.

      Enter the export data set name in the TSO data set format. You should use single quotation marks around the data set name. If the data set is a partitioned data set, include the member name. For example, if XXX.YYY.ZZZ is the partitioned data set and QQQ is the member name, type the following name:

      ```
      'XXX.YYY.ZZZ(QQQ)'
      ```

   Press **Enter** to export the variables in the current IVP environment to the target IVP session.
3. Optional: Allocate the export data sets from the IVP Export Data Set Allocation panel. If the export data set does not exist, the IVP Export Data Set Allocation panel opens as shown in the following figure.

```
                  IVP Export Data Set Allocation
Command ===>

Export data set does not exist.

Select an option to allocate the data set:

  1.  DSUTIL - ISPF data set utility panel (3.2)

  2.  ALLOC - Allocate using TSO allocate command

TSO Allocate Command:
ALLOC DATASET ('IMS.IVP.EXPORT14')  NEW CATALOG SPACE(1 1) TRACKS RECFM(F B)
LRECL(80) BLKSIZE(0)_____________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
```

Figure 9. IVP Export Data Set Allocation panel

a. Select one of the following options to allocate the data set:

   i) **DSUTIL**: If you select the DSUTIL option, the ISPF Utility Data Set Utility panel opens. Specify the following attributes for the export data set:

   - DSORG: Sequential or partitioned
   - RECFM: FB
   - LRECL: 80
   - BLKSIZE: Multiple of 80

   ii) **ALLOC**: If you select the ALLOC option, type the name of the data set in the **TSO Allocate Command** field. The data set name that you specify on the panel is used to issue the TSO ALLOCATE command to allocate the data set. If the export data set name includes a member name, the TSO ALLOCATE command allocates a PDS data set. You can edit the command on the panel before you select this option.

b. Press **Enter** to allocate the data set. The IVP Variable Export Utility panel opens with a message indicating that the data set was successfully allocated.

c. Press **PF3** or **End** to return to the IVP Variable Gathering panel.

4. Optional: If the current IVP environment does not match the environment in which the variables were exported, the IVP Import Environment Mismatch panel opens. You can choose to continue the import process or cancel it. The exported variables are associated with their specific IVP environment.

   The following figure shows the IVP Import Environment Mismatch panel.

```
                  IVP Import Environment Mismatch      IMS 15.1
IVP
Command ===>

The current IVP environment and the export data set IVP environment do not match.
Current Environment:
Export Environment:

Select an option:
    1. Continue import
    2. Cancel import
```

Figure 10. IVP Import Environment Mismatch panel

If a mismatch occurs between the IVP environments or the IMS releases, the following processing occurs:

- Any variable that is not valid in the current IMS release or for the current IVP environment and suboptions being processed is ignored.
- Any variable with a value that is specified in the export data set is replaced with the export value, even if you have modified that variable.

- The value of each of the variables is checked against the valid values for the variable in the release that is being processed.
- After the import process finishes, any variable with a value not specified in the export data set remains unchanged from its value before the import.

5. Import the variables to the target IVP session from the export data set.
   a. In the Variable Gathering (LST mode) panel, issue the import action command (**Imp**) in the action field of any variable in the panel. This command imports all the variables from an IVP export data set; it does not import a specific variable.

      The IVP Export Data Set Name panel displays to prompt you for the name of the IVP export data set.
   b. Type the name of the export data set name in the TSO data set format. Use single quotation marks around the data set name. If the data set is a partitioned data set, include the member name.

The IVP variables from your previous IVP environment are imported into your new IMS release.

Related tasks

Making global changes to variables

Use the export and import process during the variable gathering phase to make global changes to variables (for example, to change the release from "11" to "12") before you import them into a new IMS system.

Making global changes to variables

Use the export and import process during the variable gathering phase to make global changes to variables (for example, to change the release from "11" to "12") before you import them into a new IMS system.

To make global changes to variables before you import them into a new IMS:

1. Export the variables into an export data set by using the method that is appropriate for your version of IMS.

   The contents of the export data set might look like this:

```
000001 <ivpenv>DBT</ivpenv>
000002 <var>IXUMCP2</var> <val>IMSIVP.IVP11,IMSIVP,DFLT,CYL,3</val>
000003 <var>IXUMCP1</var> <val>IMSIVP.IVP11,IMSIVP,DFLT,CYL,3</val>
000004 <var>IXUSPL3</var> <val>IMSIVP.IVP11,IMSIVP,DFLT,CYL,1</val>
000005 <var>IXUSPL2</var> <val>IMSIVP.IVP11,IMSIVP,DFLT,CYL,1</val>
```

   In the export data set:

   - The <ivpenv></ivpenv> tags indicate the IVP environment.
   - The <var></var> tags indicate the variable names.
   - The <val></val> tags indicate the variable values.
2. Use the ISPF editor to modify these variables.
3. Import the variables into the target IVP by using the import action command, **Imp**.

Related tasks

Exporting and importing IVP variables

Use the IVP Variable Export utility to export previously used IVP variables to a sequential data set so you can later import them.

Tailoring files

In the file-tailoring phase, the IVP uses variables that you specified during the variable-gathering phase to prepare a customized set of IVP JCL and tasks to be stored as members of the INSTALIB data set for use in the execution phase.

The ISPF file-tailoring facility creates this input by updating and building members in the INSTALIB data set based on the options you choose in this phase.

The IVP names the INSTALIB members according to the environment option that was chosen:

IV1ssnnt - DBB - Batch system
IV2ssnnt - DBC - DBCTL system
IV3ssnnt - DBT - DB/DC system
IV4ssnnt - XRF - XRF system
IV9ssnnt - DCC - DCCTL system

Where:

ss
Step number

nn
JOB/TASK/INDEX item number within the step.

The item numbers are not guaranteed to be in ascending sequence. Service changes might disrupt the apparent sequence.

t
J for job, T for task, N for Non - job (such as an example)

To perform file-tailoring:

1. In the IVP Phase Selection panel, select a phase. Each selection within a phase provides a different positioning option.

Option	Description
Select 3	to start or restart from the beginning of the phase
Select 4	to restart from the last known position within the phase
Select 5	to restart from the beginning of a selected step

   If you select 3, and you are performing file-tailoring for the first time, or you have selected a new IVP environment option, the File Tailoring All Request panel opens.

   a. Select option **1** to perform the File Tailoring ALL action. This action causes all items to be processed. The File Tailoring in Progress panel opens with the `Please do not interrupt this process` message and the keyboard is locked. When file-tailoring is complete, the FT Complete Verification panel opens.

   b. In the FT Complete Verification panel, press Enter to continue to the File Tailoring panel.

2. In the File Tailoring panel, use the various action commands to browse, display a description, or edit any of the members on this panel.
3. Press End or PF3 when you are finished.

The IVP Phase Selection panel opens and you can progress to the IVP execution phase.

Related reference

“File-tailoring dialog options” on page 137

Use the action commands and modes available for the file-tailoring phase.

Executing tailored jobs and tasks

You must process the jobs and tasks that were prepared by the file-tailoring phase individually through the execution phase.

To execute the IVP jobs and tasks:

1. In the IVP Phase Selection panel, select option **6**, **7**, or **8**.

   Each selection within a phase provides a different positioning option.

   The Execution panel opens and displays the list of the IVP jobs and tasks.
2. Open each job and task. To view the instructions for each job and task use the **ENT** action command. Use the appropriate display modes and action commands.
   - **For IVP jobs:** You can browse, edit, or submit the job. When you are ready to run a job, you can either submit the job using the **EXE** action or you can edit and submit the job. Each job has a scrollable description that is associated with it to assist you in running the job.

     Some items are nonexecutable examples. For these examples, the submit action is disabled, but the browse and edit actions are available. You can use ISPF split-screen mode to create an executable version of nonexecutable items.
   - **For IVP tasks:** You are provided a scrollable description to assist you in performing the task.
3. Press End or PF3 when you are done. Then press Enter again if you completed the execution of all jobs and tasks, or press End to save your work if you want to complete the execution phase later.

   You can return to the same location and run the jobs or tasks from a previous point by using the positioning options on the IVP Phase Selection panel.

Related reference

“Execution phase dialog options” on page 140
Use the action commands and modes available for the Execution phase.

Ending the IVP dialog session

You can end an IVP dialog session from any panel.

To end an IVP dialog session:

1. Press **End** repeatedly until you have backed out of the dialog. Each time you press **End**, you return to the last viewed panel.
2. Press **Return** to back out of the dialog completely.

Chapter 4. Sample applications provided by the IVP

The sample applications provided by the IVP verify the IMS product installation by using an IVP sample system.

Some of the sample applications are fully executed by running the IVP jobs and tasks while others can be executed outside the IVP dialog process. The IVP provides several sample applications that test various components of IMS. The sample applications consist of steps that can be executed using different methods. Most of the sample applications use a combination of batch and online processing. Sample applications executed using the batch method are performed by submitting jobs.

To execute an online sample, consult the IVP dialog's online help for step-by-step instructions for running the jobs and tasks for the sample. The exceptions are the sample applications provided for IMS solutions for Java development and IMS callout.

Related concepts

“IVP output ” on page 8
The IVP system provides a rich and diverse set of jobs and tasks that, after completion, provide a fully functional IMS system that is integrated into the z/OS environment and tested in the selected environment from the IVP panels.

IVP-executed sample applications

The IVP-executed sample applications include all the steps that are necessary to set up and run the sample applications in the IVP jobs and tasks.

IVP sample application

The IVP sample application is a simple telephone book application, also known as the phonebook application, that sends a transaction to request information from the IMS Telephone database.

The IVP phonebook application is demonstrated through the H series jobs and tasks of the IVP. After you complete the IVP jobs and tasks during the execution phase, the IVP sample application is fully executed, and the testing of the IMS components associated with the IVP sample application is complete. You can query the database to retrieve customer information such as first name, last name, telephone extension, and zip code. The Telephone database is loaded during the processing of the IVP jobs and tasks.

The functions and features that are tested by running the IVP sample application include several IMS databases and environments.

- The tested IMS databases consist of HIDAM/OSAM, HDAM/VSAM, DEDB/VSAM, MSDB, and GSAM databases.
- The tested IMS environments include non-conversational and conversational MPP, conversational JMP, non-conversational IFP (EMH), DB batch, DLI batch, BMP, JBP, message-driven WFI BMP, non-conversational and conversational message switch, IFP EMH message switch, WFI BMP GSAM, and BMP GSAM.

The IVP application program action is determined by a process code provided with the input data. The process codes are ADD, DELETE, UPDATE, DISPLAY, and TADD. Except for TADD, the process codes are self-explanatory. TADD causes the application program to add a record to the database and issue a WTOR request. Any character string could be used to reply to the WTOR issued by the TADD process. The database is changed, but the change is not committed. The TADD process code is used during the recovery portions of the IVP scripts.

For the EMH program that accesses the main storage database (MSDB), a TUPD process code is used instead of the TADD.

The online transactions are executed through an MFS block. For example, the DFSIVP1 program is executed by entering **/FOR IVTNO** at an IMS user terminal, and then entering a process code and data on the formatted screen.

When processing for the DFSIVP1 program is finished, press the Clear key and enter a new **FORMAT** command to execute a different application program.

The batch or BMP programs execute by using JCL. In the DCCTL environment, the IVP database is simulated through the use of a data area within program DFSIVAD (a message-driven WFI BMP). Programs DFSIVAE, DFSIVAF, and DFSIVAG perform message switches to send their transaction input to DFSIVAD for processing. DFSIVAD processes its input under the control of extended checkpoint/restart and returns its output to the originating terminal.

Two series of programs are included with the IVP sample application: The DFSIVA3 series and the DFSIVA6 series. The DFSIVA3 series programs test IMS components online. The DFSIVA6 series programs are executed through a batch process using JCL. These programs are provided in several different programming languages. The IVP assembles and tests several of these programs. If you do not want to use the assembled version, you must compile and bind the IVP execution outside of the IVP.

The DFSIVA3 series programs constitute a conversational MPP that accesses an HDAM/VSAM database. The Telephone database is accessed and queried through transaction input and output using an MFS screen format. Instructions on how to use the MFS screen are included in the IVP tasks. To display or delete a record, only the process code and the last name field are required input. To add or replace a record, all input fields are required.

The following figure shows the MFS screen format for the IVP sample application.

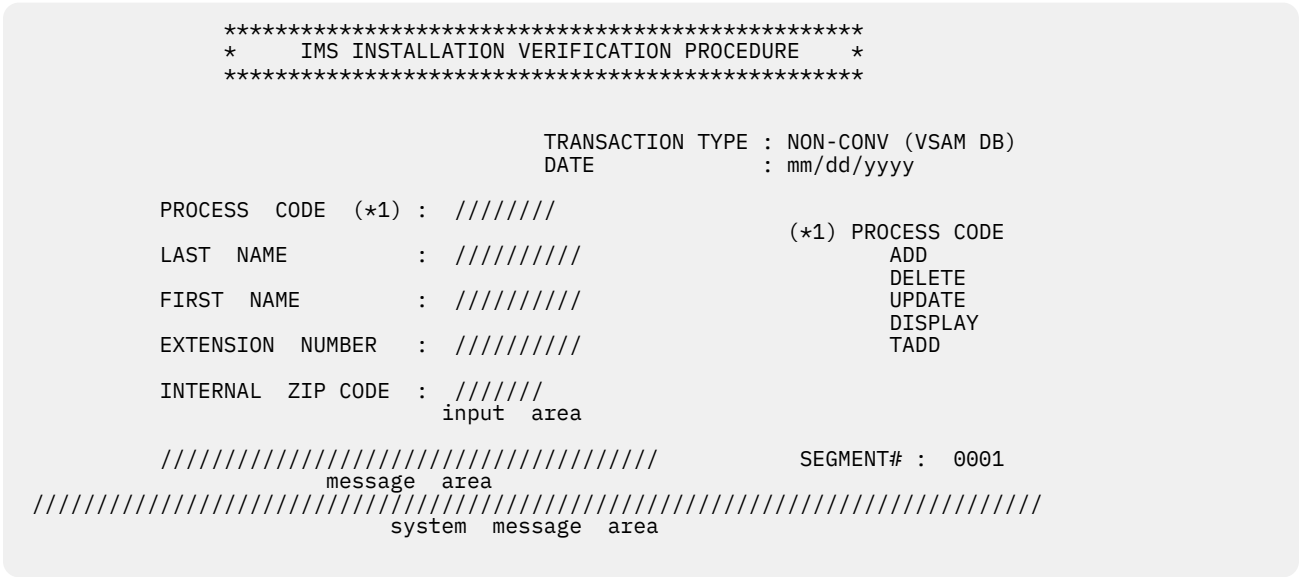


Figure 11. MFS screen format for the IVP sample application

The DFSIVA6 series programs are batch or BMP programs that access an HIDAM/OSAM database. The programs use GSAM to receive their transaction input and to display their transaction output. Instructions on how to run these jobs are included in the IVP online help of the IVP jobs and tasks.

Related reference

"IVP sample application table" on page 144
The SDFSISRC target library contains the source for programs, PSBs, DBDs, and MFSs, and other supporting materials that are used by the application.

"IVP sample application databases" on page 150
In the DCCTL environment, the IVP database is simulated through the use of a data area within program DFSIVAD.

"Steps Hx for IVP execution - DBT system (DB/DC)" on page 104

The H series of steps include the jobs and tasks that you must perform during the execution of the DBT (DB/DC) sample system.

Partitioning sample application

The Partitioning sample application provided by the IVP demonstrates the conversion of a non-partitioning database to a partitioned database.

After you complete the IVP jobs and tasks during the execution phase, the Partitioning sample application is fully executed, and the IMS components that are associated with the Partitioning sample application are tested.

This sample is based on the HIDAM database and applications of the IVP sample application but does not depend on it. This partitioning sample application is stand alone; that is, the IVP sample application does not need to be run.

This sample includes all the steps you need to perform to define, create and verify the partitioned database. These instructions are included in the online help of the IVP N series jobs and tasks. The partitioned database is set up and verified using an MFS screen format.

Defining the partitioned database includes the following processes:

1. Create and initialize a non-partitioned HIDAM database.
2. Unload the database by specifying Migrate = YES.
3. Delete the old database from the RECON data sets.
4. Define the partitioned database using %DFSHALDB.
5. Allocate the partitioned database.
6. Initialize the partitioned database.
7. Reload the partitioned database.
8. Create an image copy of the partitioned database.

Verifying that the partitioned database includes initializing IMS and running sample transactions through an MFS screen, terminating IMS, and performing clean-up activities.

Related concepts

"IVP sample application" on page 25
The IVP sample application is a simple telephone book application, also known as the phonebook application, that sends a transaction to request information from the IMS Telephone database.

Related reference

"Partitioning sample application table" on page 149
The parts used by the IVP sample partitioning application are identified in the following table.

"Partitioning sample application databases" on page 154
The following description about the database can assist you in becoming familiar with the sample database used by the partitioning sample application.

"Steps Nx for execution - partition database sample application" on page 110
The N series of steps include the jobs and tasks that you must perform during the execution of the IMS partition database sample application.

IMS Connect sample application

The IMS Connect sample application demonstrates that IMS Connect is operational by running a full-function MPP transaction using IMS Connect.

Execution of the IMS Connect sample application is performed through the *Q* series jobs and tasks of the IVP. The *Q* series jobs and tasks of the IVP contain all the necessary steps that you must perform to verify that IMS Connect is running properly.

The basic message processing flow for the IMS Connect sample application is as follows:

1. The client application sends a message request (input message) through a TCP/IP session to IMS Connect to obtain information about the part named AN960C10.
2. IMS Connect passes the message to the host IMS for processing.
3. The host IMS schedules a program to access the database named PART and to obtain information from the part AN960C10.
4. The host IMS sends the response message or output message to IMS Connect.
5. IMS Connect passes the response back to the application client.

Related reference

"IMS Connect sample application table" on page 149
The following table shows the parts used by the IMS Connect sample application. These parts are all installed by the IVP jobs and tasks.

"Steps Qx for execution - full-function MPP transaction using the IMS Connect sample application" on page 116
The Q series of steps include the jobs and tasks that you must perform during the execution of full-function MPP transaction using the IMS Connect sample application.

Dynamic resource definition sample application with RDDSs

The dynamic resource definition (DRD) sample application demonstrates how to set up and operate DRD with resource definition data sets (RDDSs).

The IVP provides the following steps to create an online IMS execution scenario that exercises DRD functions.

1. The IVP prepares the resources by creating DBD and PSBs for the new resources.

   This step is demonstrated in the E series jobs and tasks.
2. The IVP sets up the environment for DRD by:

   a. Creating the PROCLIB member DFSDF000. The IVP uses the following parameters:

Parameter	Description
MODBLKS=DYN	Enables DRD for MODBLKS resources.
AUTOEXPORT=AUTO	Exports resources automatically to an external data set at a checkpoint time.
AUTOIMPORT=AUTO	Imports resources automatically from an external data set during an IMS cold start.
RDDSDSN=(IMS.RDDS1,IMS.RDDS2,IMS.RDDS3)	Defines a set of three BSAM data sets that are used to save IMS resource definitions.

   b. Allocating three resource definition data sets (IMS.RDDS1, IMS.RDDS2, and IMS.RDDS3) that are used to save IMS resource definitions.

   This step is demonstrated in the E series jobs and tasks.
3. The IVP executes the IMS system online by:

   a. Allocating database data sets for the new DBD

   b. Starting the IMS system with a cold start.

   c. Creating new resources (database, programs, and transactions) by using DRD.

   d. Executing the new programs and transactions to verify that the new resources are in effect.

      - The program DFSIVPD1 accesses the existing phone book database to update a record and delete it.

- The program DFSIVPD2 is processed when the transaction IVTND is requested from a user terminal. The program also accesses the phone book database and responds to user terminal based on user action request to display, add, update, or delete a record.

e. Shutting down the IMS system normally.

This step is demonstrated in the O series jobs and tasks.

4. The IVP restarts the IMS system by:

   a. Starting the IMS system with a cold start.

   b. Executing the new programs and transactions again to verify that the new resources are still in effect.

   c. Shutting down the IMS system normally.

   This step is demonstrated in the O series jobs and tasks.

Dynamic resource definition sample application with the IMSRSC repository

The dynamic resource definition (DRD) sample application demonstrates how to set up and operate DRD with the IMSRSC repository.

The IVP provides the following steps to create an online IMS execution scenario that exercises DRD functions.

1. The IVP prepares the resources by creating a DBD and PSBs for the new resources.

   This step is demonstrated in the E series jobs and tasks.

2. The IVP sets up the environment for DRD by:

   a. Creating the PROCLIB member DFSDF000. The IVP uses the following parameters:

Parameter	Description
MODBLKS=DYN	Enables DRD for MODBLKS resources.
AUTOEXPORT=AUTO	Exports resources automatically to an external data set at a checkpoint time.
AUTOIMPORT=AUTO	Imports resources automatically from an external data set during an IMS cold start.
RDDSDSN=(IMS.RDDS1,IMS.RDDS2,IMS.RDDS3)	Defines a set of three BSAM data sets that are used to save IMS resource definitions.

   This step is demonstrated in the E series jobs and tasks.

3. The IVP executes the IMS system online by:

   a. Allocating database data sets for the new DBD.

   b. Allocating the IMSRSC repository data set and the RS catalog repository data set.

   c. Starting the Common Service Layer (CSL), consisting of an Operations Manager (OM), a Resource Manager (RM), and a Structured Call Interface (SCI).

   d. Starting the Repository Server (RS).

   e. Adding an IMSRSC repository to the RS catalog repository, then starting the IMSRSC repository.

   f. Listing the status information for all IMSRSC repositories in the RS catalog repository.

   g. Populating the IMSRSC repository.

   h. Stopping and renaming the IMSRSC repository in the RS catalog repository.

   i. Listing detailed information for a single IMSRSC repository.

   j. Modifying the resource definitions in the IMSRSC repository.

   k. Deleting an IMSRSC repository in the RS catalog repository.

l. Requesting the RS to start a previously deleted IMSRSC repository.

m. Shutting down the SCI, OM, RM, and RS.

n. Deleting the IMSRSC repository data sets and the RS catalog repository data sets.

This step is demonstrated in the U series jobs and tasks.

Related reference

“Steps Ux for the IMSRSC repository sample application” on page 120
The U series of steps include the jobs and tasks that you must perform during the execution of the IMSRSC repository sample application.

Other sample applications verified by the IVP

The IVP jobs and tasks also verify and demonstrate the setup of several IMS components, such as the Common Service Layer (CSL) and Common Queue Server (CQS) sample application, the type-2 command environment sample application, SPOC display of OM audit trail sample application, and the parallel RECON access sample application.

The sample applications to test IMS components include:

- CSL and CQS sample application

  This sample application demonstrates how to use the Operations manager (OM), Resource manager (RM), Structured Call Interface (SCI), TSO single point of control (SPOC), and CQS. Specifically, this sample application demonstrates:

  – Adding CSL members OM, RM, and SCI to the IMS PROCLIB data set to define an IMSplex
  – Adding CQS members to the IMS PROCLIB data set
  – Starting and stopping an IMSplex and CQS
  – Starting and using the TSO SPOC application, including how to issue IMS type-1 and type-2 commands

  The steps for these sample applications are described in the O series jobs and tasks of the IVP.
- Type-2 command environment sample application

  This sample application demonstrates how to use OM, SCI, and the TSO SPOC without RM. I

  n addition, you can inform OM whether to display information from the audit trail.

  Specifically, this sample application demonstrates:

  – Adding OM and SCI members to the IMS PROCLIB data set to define an environment in which RM is not required and type-2 commands can be issued
  – Using the TSO SPOC to issue commands to IMS

  The steps for this sample application are described in the P series jobs and tasks of the IVP.
- SPOC display of OM audit trail sample application

  This sample application demonstrates how to display the OM audit trail information from a TSO SPOC session. You can view an audit trail of command input, associated command response output, and unsolicited output messages by using the TSO SPOC menus.

  The steps for these sample applications are described in the O series jobs and tasks of the IVP.

  Support for the OM audit trail functions is available in the IVP. The OM audit trail records unsolicited messages as well as command input and command responses to a z/OS log stream. The z/OS system logger is required to provide storage for the audit trail. Two jobs, IV_E303J and IV_E307T, are part of the backend preparation for the audit trail functionality.

  The sample job that defines z/OS policies (IV_E307T) names the primary structure as IMSOM2Q01 for the audit trail purposes. This structure is associated with the AUDITLOG= parameter of CSLOI*xxx* member (the OM initialization IMS PROCLIB data set member) to specify a log stream data set name for storing log records.

For more information about IMS OM audit trail, see *IMS Version 15 System Administration*.

- Parallel RECON access sample application

  IMS IVP provides instructions on how to configure parallel access to RECON data sets in the sample IMS system and how to verify the new function setup after an IMS installation.

  The steps for these sample applications are described in the R series jobs and tasks of the IVP. For more information about IMS parallel RECON access, see *IMS Version 15 System Administration*.

Related reference

"Steps Ox for Common Service Layer and Common Queue Server sample application" on page 111
The O series of steps include the jobs and tasks that you must perform during the execution of the Common Service Layer sample application.

"Steps Px for type-2 command environment sample application" on page 115
The P series of steps include the jobs and tasks that you must perform during the execution of the enhanced command environment sample application.

Sample applications not tested by the IVP

Two sample applications are set up by the IVP but are not exercised by the IVP: the IMS sample application and the Fast Path sample application.

The IVP jobs and tasks only set up the environment in which you can run these sample applications. These applications are run outside the IVP.

- "IMS sample application" on page 31
- "Fast Path sample application" on page 37

IMS sample application

The IVP jobs and tasks create a basic structure to run the IMS sample application.

The structure includes:

- Building all the IMS blocks
- Creating the application programs
- Loading the database

The IMS sample application includes the creation, usage, and maintenance of the Parts database that is needed to run the sample application. You need to perform verification of this database through an IMS terminal user session.

The following figure shows the sample application's logical view of the Parts database.

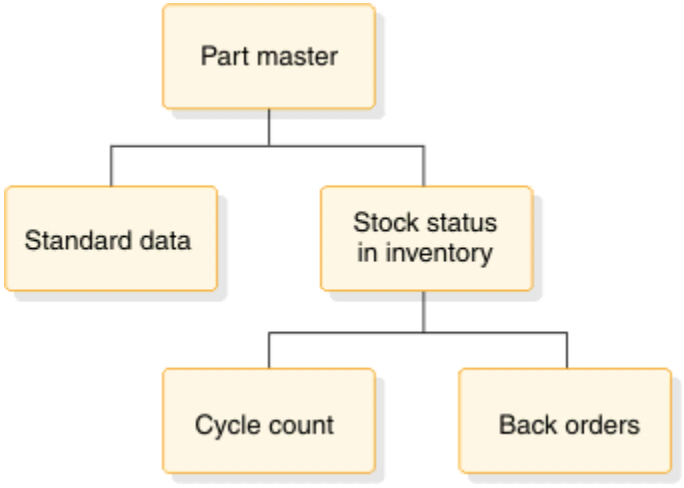


Figure 12. IMS sample application's logical view of the parts database

The application requires five segments of the Parts database:

- One part number description segment for each part within the database.
- A standard data segment for each part that provides additional information of a standard nature about the part.
- Inventory stock status segments for each part. The application is designed with multiple inventory locations permissible, and normally required, for any particular part.
- Cycle count segments (from 0 to *n*).
- Back-order segments for each inventory location of a particular part.

Related reference

"IVP sample application table" on page 144
The SDFSISRC target library contains the source for programs, PSBs, DBDs, and MFSs, and other supporting materials that are used by the application.

"Steps Hx for IVP execution - DBT system (DB/DC)" on page 104
The H series of steps include the jobs and tasks that you must perform during the execution of the DBT (DB/DC) sample system.

Running the IMS sample application

To verify the IMS sample application databases, you must run the IMS sample application using the six PSBs with their associated transactions and nine online functions.

- Run the IVP jobs and tasks through the H series to set up the basic infrastructure for the IMS sample application.
- Sign on to an IMS user terminal session.

To run the IMS sample application:

1. Run the PART transaction.

   The transaction PART queries the part number database for information from the part master and standard information segments of a specific part number. Execute the IMS sample application transactions using the following format:

   ```
   TRANSACTION_CODE OPERAND,OPERAND,OPERAND, . . .
   ```

   Separate the transaction code from the first operand with one blank. Blanks cannot be entered between one operand and another. Most of the transaction codes have been defined as multiple segment transactions and require an EOT (end-of-transmission, for 2740), or equivalent, to complete input.

   The input format is transaction code and part number as illustrated in the following figure.

   ```
   part AN960C10
   ```

   Figure 13. PART transaction - entry

   The output or response format is shown in the following figure.

   ```
            Part..........     AN960C10; Desc.......... WASHER
            Proc Code......           74; Inv Code.......        2
            Make Dept......        12-00; Plan Rev Num...
            Make Time......           63; Comm Code......       14
   ```

   Figure 14. PART transaction - output

   The available part numbers that you can use for message processing are:

   Part Numbers:

```
AN960C10                        7438995P002
3003806 *                       7618032P101 *
3007228                         922399-001
3013412                         82125-869
652799
```

The part numbers marked with an asterisk (*) have dependent back-order segments. All part numbers have at least one dependent inventory status segment.

2. Run the DSPALLI transaction.

   The DSPALLI transaction displays all inventory, cycle count, and back-order information for a specific part. The following figure illustrates the input format of transaction code and part number.

```
dspalli AN960C10
```

Figure 15. DSPALLI transaction - entry

The resulting terminal output is shown in the following figure.

```
   Part=AN960C10;   Desc=WASHER;   Proc Code=74

      Area  Inv  Proj  Div  Unit   Current   On     In     Total  Count Back
            Dept  CD        Price  Reqmts  Order  Stock  Disburse Taken Ordr
      ----  ----  ---  ---  -----  ------  -----  -----  -------- ----- ----
   1.        AA   165   11  0.000   146     20     126      104     No    0
   2.        AK   287   7F  0.000    88      0      88       37     No    0
   3.   2    80   091   26  0.000   630     15     680     1157     No    0
```

Figure 16. DSPALLI transaction - output

3. Run the DSPINV transaction.

   The DSPINV transaction displays inventory information from a specific inventory location. Assume that you want to display only the third inventory entry listed in Figure 16 on page 33. Obtain the inventory location key by concatenating AREA, INVDEPT, PROJCD, and DIV.

   The following figure illustrates the input format of transaction code followed by part number and inventory-location-key.

```
dspinv AN960C10,28009126
```

Figure 17. DSPINV transaction - entry

The resulting terminal output is shown in the following figure.

```
          Part...........     AN960C10; Desc........... WASHER
          Proc...........           74; Area...........        2
          Inv Dept.......           80; Prj............      091
          Div............           26; Price..........    0.000
          Stk Ct Date....          513; Unit...........     EACH
          Curr Reqmts....          630; On Order.......       15
          Total Stock....          680; Disb Planned...     1053
          Disb Unplanned.          104; Stk Ct Variance        0
```

Figure 18. DSPINV transaction - output

4. Run the ADDPART transaction.

   The ADDPART transaction adds a new part and its associated description and procurement code to the database. The input format for this transaction is transaction code, followed by part number, description, and procurement-code, entered as shown in the following figure.

```
addpart AB960C10,RIVET,74
```

Figure 19. ADDPART transaction - entry

The resulting terminal output is shown in the following figure.

```
 Part Number AB960C10 Added To Data Base
```

Figure 20. ADDPART transaction - output

5. Run the ADDINV transaction.

   The ADDINV transaction adds inventory location key information to an existing part in the database. The input format for this transaction is transaction code followed by part number and inventory-location-key, entered as shown in the following figure.

```
addinv AB960C10,80091260
```

Figure 21. ADDINV transaction - entry

The resulting terminal output is shown in the following figure.

```
 Inventory 80091260 Added To Part Number AB960C10
```

Figure 22. ADDINV transaction - output

If you want to display the part's updated inventory information, enter the command shown in the following figure.

```
dspinv AB960C10,80091260
```

Figure 23. DSPINV transaction - entry

The resulting terminal output is shown in the following figure.

```
          Part...........     AB960C10; Desc........... RIVET
          Proc...........           74; Area...........        8
          Inv Dept.......           00; Prj............      912
          Div............           60 ; Price..........    0.000
          Stk Ct Date....             ; Unit...........
          Curr Reqmts....            0; On Order.......        0
          Total Stock....            0; Disb Planned...        0
          Disb Unplanned.            0; Stk Ct Variance        0
```

Figure 24. DSPINV transaction - output

6. Run the DLETINV transaction.

   The DLETINV transaction code deletes a specific inventory item for a specific part. The input format for this transaction is transaction code, followed by part number and inventory-location-key, entered as shown in the following figure.

```
dletinv AB960C10,80091260
```

Figure 25. DLETINV transaction - entry

The resulting terminal output shown in the following figure.

```
 Inventory 80091260 Deleted From Part Number AB960C10
```

Figure 26. DLETINV Transaction - output

7. Run the DLETPART transaction.

   If all the inventory items are deleted, you can delete a particular part number from the database with the transaction code DLETPART. The input format is transaction code followed by part number, entered as shown in the following figure.

```
dletpart AB960C10
```

Figure 27. DLETPART transaction - entry

The resulting terminal output is shown in the following figure.

```
 Part Number AB960C10 Deleted From Data Base
```

Figure 28. DLETPART transaction - output

8. Run the CLOSE transaction.

   You can close an open order for a specific part in a specific inventory item by using the CLOSE transaction code. The input format is transaction code, part number, inventory-location-key, on-order-decrement, total-stock increment, entered as shown in the following figure.

```
close AN960C10,28009126,15,15
```

Figure 29. CLOSE transaction - entry

The resulting terminal output is shown in the following figure.

```
 17:43:38 PN= AN960C10 Invty Key=28009126 Excess Stock On Hand
```

Figure 30. CLOSE transaction - output

Other messages might follow, depending on the sample database update status. You might need to press PA1 first. An example is shown in the following figure.

```
 Update Complete
```

Figure 31. CLOSE transaction - output (additional)

To verify the operation of the CLOSE transaction, you can display inventory item 28009126 for part AN960C10. The input format is transaction code, part number, inventory-location-key, entered as shown in the following figure.

```
dspinv AN960C10,28009126
```

Figure 32. DSPINV transaction - entry

The resulting terminal output is shown in the following figure.

```
          Part..........     AN960C10; Desc.......... WASHER
          Proc..........           74; Area...........        2
          Inv Dept......           80; Prj............      091
          Div...........           26; Price..........    0.000
          Stk Ct Date....         513; Unit...........     EACH
          Curr Reqmts....         630; On Order.......        0
          Total Stock....         695; Disb Planned...     1053
          Disb Unplanned.         104; Stk Ct Variance        0
```

Figure 33. DSPINV transaction - output

Compare the input and output. Notice that the on-order quantity was reduced by 15 and the total stock quantity has been increased by 15 to 695.

9. Run the DISBURSE transaction.

   The DISBURSE transaction code allocates a quantity of a part from an inventory item on a planned or unplanned basis. The input format is transaction code, part number, inventory-location-key, planned or unplanned code, quantity. Enter the command as shown in the following figure.

```
disburse AN960C10,28009126,U,10
```

Figure 34. DISBURSE transaction - entry

The resulting terminal output is shown in the following figure.

```
 17:47:40 PN= AN960C10 Invty Key=28009126 Excess Stock On Hand
```

Figure 35. DISBURSE transaction - output

Other messages might follow, depending upon the sample database update status. You might need to press PA1 first. An example is shown in the following figure.

```
 Update Complete
```

Figure 36. DISBURSE transaction - output (additional)

If you want to display the inventory information for key 28009126 and part number AN960C10, enter the command as shown in the following figure. The input is transaction code, part number, inventory-location-key.

```
dspinv AN960C10,28009126
```

Figure 37. DSPINV transaction - entry

The resulting terminal output is shown in the following figure.

```
          Part..........     AN960C10; Desc.......... WASHER
          Proc..........           74; Area...........        2
          Inv Dept......           80; Prj............      091
          Div...........           26; Price..........    0.000
          Stk Ct Date....         513; Unit...........     EACH
          Curr Reqmts....         630; On Order.......        0
          Total Stock....         685; Disb Planned...     1053
          Disb Unplanned.         114; Stk Ct Variance        0
```

Figure 38. DSPINV transaction - output

Related reference

"IMS sample application PSBs" on page 147
These PSBs are provided to process the Parts database that is used by the IMS sample application.

Fast Path sample application

The Fast Path sample application demonstrates a banking application.

The IVP jobs and tasks set up the infrastructure that enables you to run the Fast Path sample application, but they do not run the sample application. The Fast Path sample application creates and uses four databases. Data is related to General Ledger (MSDB), Teller (MSDB), Loan (HDAM/VSAM), and Customer Account (DEDB) information for each account. DEDB and HDAM databases are loaded in the IVP offline using IMS-supplied utilities. All four databases are processed online using message processing regions (MPP) and Fast Path regions (IFP).

Two transaction codes are used in the Fast Path sample application:

- FPSAMP1 - Executes in an IFP region
- FPSAMP2 - Executes in an MPP region

The two transaction codes both execute the same application functions. The MOD name of the MFS format that is used by these transactions is DBFSMOUT. Use the IMS `/FORMAT DBFSMOUT` command to display this format.

The following figure shows the relationship of the four databases in the Fast Path sample application.

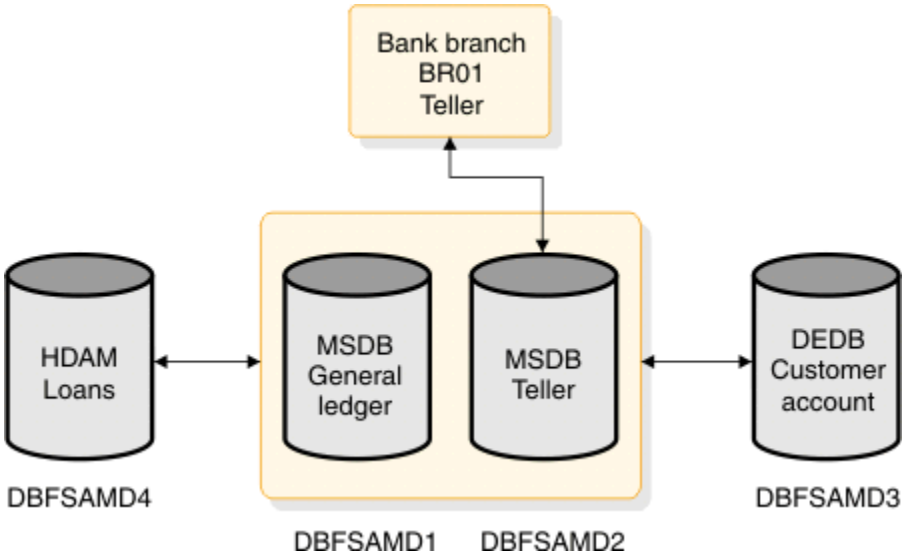


Figure 39. Relationship of the databases in the Fast Path sample application

The General Ledger database is a non-terminal-related MSDB. The DBD for the general ledger file contains a segment description that consists of the following items:

- General ledger account number
- General ledger account balance
- Transaction count
- Filler area

The Teller database is a terminal-related MSDB. The DBD for the teller file contains a segment description that consists of the following items:

- Withdrawal amount
- Deposit amount
- Loan payment amount

- Teller balance
- Transaction code
- Key to general ledger
- Filler area

The Customer Account database (a DEDB) includes nine segment types in a three-level hierarchy, as represented in the following figure. The segment types include a root segment type, a sequential dependent segment type, and seven types of direct dependent segments. In addition, subset pointers point to the three account segment types that are represented in the database. This configuration allows the application to demonstrate the use of multiple SSAs and the use of command codes (including subset pointer references) for a DEDB.

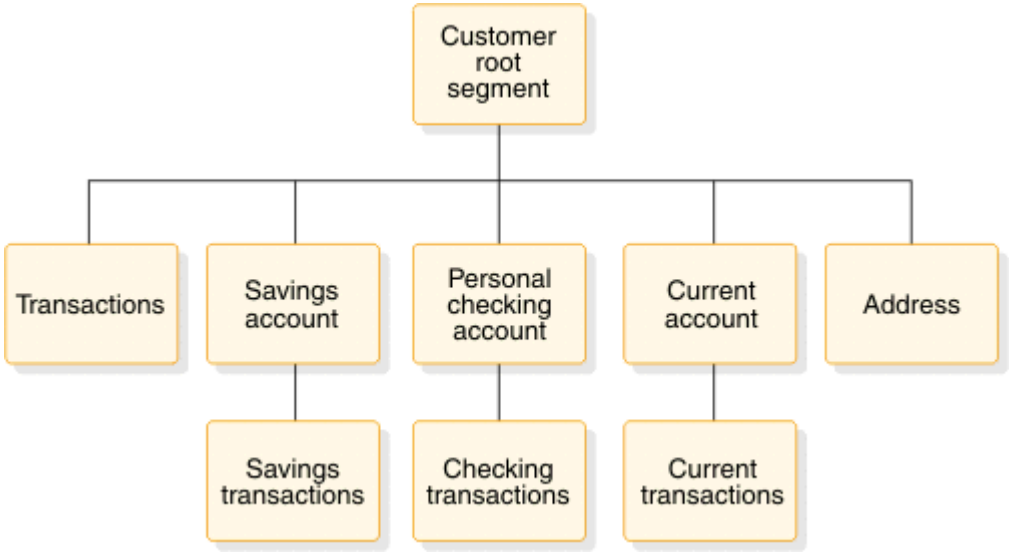


Figure 40. A hierarchical diagram of the Customer Account database (a DEDB)

The second-level transaction segment is sequential dependent; all other segments are direct dependents.

The Loan database (HDAM) contains customer identification and transaction information. Transaction information includes all aspects of a banking scenario, including loan information, account numbers, and dates and times of transactions.

The hierarchical diagram in the following figure displays the segments (customer root and loan) of the HDAM/VSAM Loan database.

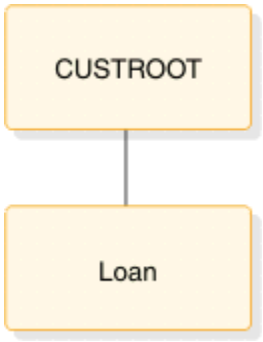


Figure 41. Segments of the HDAM/VSAM Loan database

Related reference

“Fast Path sample application table ” on page 148
The following table provides information about the parts used by the Fast Path sample application. It includes the language, PSB, MFS, transaction code, DBD, and description associated with those parts.

“Fast Path sample application databases” on page 153

The transactions that are described in the following tables can help you become familiar with the Fast Path sample application databases. Along with the following customer account information, the tables provide you the resources that you can use to prepare online training exercises for operators and programmers.

Running the Fast Path sample transactions

Run the Fast Path sample transactions from your terminal to verify the proper setup of your Fast Path infrastructure.

Run the IVP jobs and tasks through the H series to set up the infrastructure for the Fast Path sample application.

To run the sample transactions:

1. Sign on to an IMS user terminal session.
2. Press Clear and then press PA2. Repeat this sequence until a blank screen is returned. This sequence causes queued-up messages to be displayed.
3. Enter `/FORMAT DBFSMOUT` to display the MFS format.
4. Follow the transaction sequence that is described below.

   The terminal inputs have the following format:

   ```
   aaaaaaa bbbbbbbbcc def ggggggggg
   ```

   The general format of the input for these transactions is shown in the following table.

Table 1. Example input format for Fast Path sample application transactions

Field	Variables	Description
Transaction code	*aaaaaaa*	• FPSAMP1 - Execute transaction in FP message-driven region • FPSAMP2 - Execute transaction in IMS MPP region
Customer Account	*bbbbbbbbcc*	• *bbbbbbbb* - Eight-character customer number • *cc* - Two -character account type
Transaction type	*def*	• *d* - One of the following four characters: – L - Loan[a] – S - Savings account – C - Checking account – U - Current account • *e* - One of the following three characters: – W - Withdrawal – D - Deposit – P - Account statement • *f* - One of the following five characters: – P - Passbook[b] – 1 - Today[c] – 2 - This week[c] – 3 - This month[c] – 4 - This quarter[c]

Table 1. Example input format for Fast Path sample application transactions (continued)

Field	Variables	Description
Transaction amount	*ggggggggg*	Amount ($3000.00, for example, up to nine characters)

Note:

a. A transaction amount is not required on load transactions or account statement requests. A loan payment amount is predefined in the database.

b. For savings account deposits and withdrawals with a passbook. If no passbook exists, this character is left blank.

c. Valid combinations for statement requests are: SP3, SP4, CP2, CP3, CP4, UP1, UP2, UP3, and UP4.

a) Type the following terminal input: `FPSAMP1 BR01-H01M1 L`

The terminal output displayed is:

```
LOAN PAYMENT DETAILS:
BR01-H01M1 L     $482.77
   $60,000.00  $59,517.23  0001
```

The general format of the output for the Fast Path sample application transactions is listed in the following table.

Table 2. Example output format for Fast Path sample application transactions

Transaction	Return output	Syntax description

Customer Account	`CUST.ACCT TRANSACTION:`
	
	`BR`*`xxxxxxxxx yyy zzzzzzzzz`*
	`wwwwwwwww`
	
	`TRANS TO BE ENTERED IN PASSBK:`
	
	`YYDDD HHMM t aaaaaaaaa`
	`YYDDD HHMM t aaaaaaaaa`
	
	`YYDDD HHMM t aaaaaaaaa`
	`YYDDD HHMM t aaaaaaaaa`
	
	`END OF PASSBOOK TRANSACTIONS`
		Where:
		• *xxxxxxxx*: Customer account number
		• *yyy*: Transaction type
		• *zzzzzzzzz*: Transaction amount
		• *wwwwwwwww*: Account balance
		• *YYDDD*: Transaction date
		• *HHMM*: Transaction time
		• *t*: Transaction type (D or W)
		• *aaaaaaaaa*: Transaction amount

Account Statement	`CUST. ACCT  REQUEST   BALANCE:`
	
	`BR`*`xxxxxxxxx yyy wwwwwwwww`*
	
	`TRANSACTIONS THIS PERIOD:`
	
	`YYDDD HHMM t aaaaaaaaa`
	`YYDDD HHMM t aaaaaaaaa`
	
	`YYDDD HHMM t aaaaaaaaa`
	`YYDDD HHMM t aaaaaaaaa`
	
	`END OF TRANSACTIONS`
		• *xxxxxxxx*: Customer account number
		• *yyy*: Transaction type
		• *wwwwwwwww*: Account balance
		• *YYDDD*: Transaction date
		• *HHMM*: Transaction time
		• *t*: Transaction type (D or W)
		• *aaaaaaaaa*: Transaction amount

Table 2. Example output format for Fast Path sample application transactions (continued)

Transaction	Return output	Syntax description
Loan Payment	LOAN PAYMENT DETAILS: BR*xxxxxxxx L zzzzzzzzz* *wwwwwwwww uuuuuuuuu vvvv*	• *xxxxxxxx*: Customer account number • *L*: Transaction type (loan payment) • *zzzzzzzzz*: Loan payment amount • *wwwwwwwww*: Original loan balance • *uuuuuuuuu*: New loan balance • *vvvv*: Number of loan payments made on account

b) Type the following terminal input: `FPSAMP2 BR01-A01S1 SWP 1000.00`

Terminal output:

```
CUST. ACCT TRANSACTION:
BR01-A01S1 SWP        $1,000.00   $1000.00
TRANS TO BE ENTERED IN PASSBK:
YYDDD HHMM  W  $1000.00          END OF PASSBOOK TRANSACTIONS
```

Customer Account Transaction

c) Type the following terminal input: `FPSAMP1 BR02-T02C1 CD 1000.00`

Terminal output:

```
CUST. ACCT TRANSACTION:
BR02-T02C1 CD         $1,000.00   $1,900.00
```

d) Type the following terminal input: `FPSAMP2 BR01-F01C1 CW 900.00`

Terminal output:

```
REQUEST CAN NOT BE SERVICED:
PROCSG ERROR OD      BR01-F01C1    CW    $900.00
```

e) Type the following terminal input: `FPSAMP2 BR01-F01C1 CP2`

Terminal output:

```
CUST. ACCT  REQUEST        BALANCE:
BR01-F01C1 CP2               $800.00
NO TRANSACTIONS THIS PERIOD
```

f) Type the following terminal input: `FPSAMP1 BR01-A01S1 SW 500.00`

Terminal output:

```
CUST. ACCT TRANSACTION:
BR01-A01S1 SW           $500.00    $500.00
```

g) Type the following terminal input: `FPSAMP1 BR01-B01A1 L`

Terminal output:

```
LOAN PAYMENT DETAILS:
BR01-B01A1 L     $145.20   $4,500.00   $4,354.80   0001
```

h) Type the following terminal input: `FPSAMP1 BR01-A01S1 SDP 400.00`

Terminal output:

```
CUST. ACCT TRANSACTION:
BR01-A01S1 SDP          $400.00    $900.00
TRANS TO BE ENTERED IN PASSBK:
```

```
    YYDDD HHMM  W           $500.00  YYDDD HHMM  D  $400.00
    END OF PASSBOOK TRANSACTIONS
```

i) Type the following terminal input: `FPSAMP2 BR01-A01S1 SP3`

Terminal output:

```
    CUST. ACCT  REQUEST      BALANCE:
    BR01-A01S1 SP3            $900.00
    TRANSACTIONS THIS PERIOD:
    YYDDD HHMM  W         $1,000.00  YYDDD HHMM W        $500.00
    YYDDD HHMM  D           $400.00  END OF TRANSACTIONS
```

j) Type the following terminal input: `FPSAMP1 BR02-T01U1 UW 11500.00`

Terminal output:

```
    CUST. ACCT TRANSACTION:
    BR02-T01U1    UW     $11,500.00    $30,000.00
```

You have completed running the IMS Fast Path sample application and verified the proper setup of your Fast Path infrastructure.

Related reference

"Fast Path sample application error messages" on page 154

Use the following information to diagnose Fast Path sample application errors.

Samples for the IMS catalog

The IVP sets up the IMS catalog and provides sample application programs that verify that the IMS catalog is working as expected.

The IVP job IV_E319J performs the following actions to set up the IMS catalog:

1. Delete any existing IMS catalog data sets
2. Allocate new IMS catalog data sets
3. Load the IMS catalog database
4. Create an image copy of the IMS catalog data sets

During setup, you can define an alias name for the IMS catalog by specifying the IXUCATAL IVP variable, which sets the alias name on the ALIAS= parameter in the IMS catalog section of the DFSDF*xxx* member.

After set up is complete, the IVP job IV_O260J runs the DFSDDLT0 test application program, which verifies the content of the IMS catalog by retrieving the first 100 segments in the IMS catalog.

If the Java IVP feature is enabled, several O series jobs in the IVP provide four sample Java application programs that use the IMS catalog to read, insert, update, and delete data in the sample phone book database.

Two of the Java application programs, DFSCATS2 and DFSCATD2, use the type-2 IMS Universal drivers. DFSCATS2 uses SQL. DFSCATD2 uses DL/I.

The sample Java application programs CAT1GO and CAT2GO use the type-4 IMS Universal drivers. The application program CAT1GO uses SQL. The application program CAT2GO uses DL/I.

The following IVP jobs run the sample Java application programs:

- IV_O261J O2 JOB - Run Java Sample (SQL - Type 2)
- IV_O262J O2 JOB - Run Java Sample (DLI - Type 2)
- IV_O275J O2 JOB - Run Java Sample (SQL - Type 4)
- IV_O276J O2 JOB - Run Java Sample (DLI - Type 4)

Samples for IMS solutions for Java development and IMS callout

The IVP contains jobs and tasks that set up an environment in which sample applications that demonstrate the IMS solutions for Java development and IMS callout can be executed.

The IVP does not demonstrate these samples through its jobs and tasks.

Samples for the IMS solutions for Java development

The IVP performs several verifications for the IMS solutions for Java development.

- Runs a JMP application (IVP task IV_H216T - FF JMP Transactions) and a JBP application (IVP job IV_H213J - FF JBP) against the IVP Phone Book database. These applications ensure that the Java dependent regions have been correctly installed and set up, and that OMVS (UNIX System Services) is running correctly.
- In addition to the IVP Phone Book database, the IVP also sets up an environment for the Dealership database in its system definition. This database is used by many of the sample applications for the IMS solutions for Java development. After this environment is set up, the database is ready for use by the sample Java applications.

Sample applications for the IMS solutions for Java development, including sample applications for Open Database Type-2 support, are available under IMS Exchange on the developerWorks® website. Each sample package contains detailed instructions and example code to help you write applications that access and query IMS data.

Samples for the callout function

The IVP sets up an environment in its system definition and provides samples in assembly, Java, and COBOL languages, and two related XML converters, for the asynchronous and synchronous callout functions. These samples demonstrate how an IMS application can send outbound messages to request services or data from an external web service or Java applications, and, optionally, receive responses in either the same or a different transaction.

The samples are provided in the S series of jobs and tasks, with introduction information in IV_S001T.

Parts DFSACALO and DFSSCALO provide user-supplied IMS Connect client application samples in assembly language. Compile and bind the samples by running the job IV_E206J. Run job IV_S222J to start the asynchronous sample and job IV_S225J to start the synchronous sample.

Two Java samples, available in parts DFSJASMP and DFSJSSMP, provide sample user-supplied IMS Connect client applications in Java for the asynchronous and synchronous callout functions. These samples can run on either distributed or z/OS platforms.

Two COBOL samples, available in parts DFSASCBL and DFSSSCBL, demonstrate the asynchronous and synchronous callout functions as supported by the IMS TM resource adapter. DFSASCBL is located in the SDFSISRC data set. Compile and bind the samples before running the samples through job IV_S227J (for asynchronous) and job IV_S228J (for synchronous).

Two XML converters, DFSACCBL and DFSSCCBL, are provided when the callout requests are routed through the IMS Enterprise Suite SOAP Gateway. DFSACCBL is located in the SDFSISRC data set. Compile and bind the samples before running the samples through job IV_S230J (for asynchronous) and job IV_S231J (for synchronous).

The web service or Java applications that these samples call out to can be downloaded from the IMS TM Resource Adapter and IMS Enterprise Suite SOAP Gateway web pages. Each sample package contains detailed instructions.

Chapter 5. Reference information

This section provides reference information for the IVP.

IMS data sets

Use the data set information to troubleshooting and learning about the IMS.

The attribute values for each data set are provided and include:

DSORG
Data set organization

Restriction: IMS supports partitioned data sets extended (PDSEs) for only these libraries: MODBLKS, PGMLIB, SMPLTS, and SDFSJLIB.

Note: The External Subsystem Attach Facility (ESAF) supports PDSE load library data sets.

DSNTYPE
Data set name type

RECFM
Record format

LRECL
Logical record length

BLKSIZE
Block size

The DSNAME high-level qualifier for DLIB, SYSTEM, and EXECUTION data sets must be specified on the NODE parameter of the IMSGEN macro. TLIB data sets are included in the NODE parameter for SYSTEM data sets. The IMS online change function requires multiple copies of the system data sets IMS.ACBLIB, IMS.MODBLKS, and IMS.FORMAT. The base copies of these data sets are called staging libraries, and the copies form active and inactive libraries.

IVP dialog data sets

IVP dialog data sets are user data sets (not known to SMP/E) that are needed by the IVP dialog.

IMS.INSTALIB data set

The INSTALIB data set contains the IMS installation materials created by the file tailoring phase of the IVP dialog.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.INSTATBL data set

The INSTATBL data set contains the ISPF tables that are read and updated by the IVP dialog.

This data set has the following attributes:

DSORG
Partitioned

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.IVP.EXPORT data set

The IMS.IVP.EXPORT data set is the export data set that is used in the process for exporting and importing variables during the IVP variable-gathering phase. The data set can have any name. If the data set does not exist, you can create it during the export process.

DSORG
Sequential or partitioned

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

SMP/E data sets

SMP/E data sets establish the SMP/E environment for IMS. Do not share these data sets with other products.

One SMP/E Consolidated Software Inventory (CSI) can support multiple zones. Products that have the same SMP/E SREL (P115 for IMS) can share the same SMP/E CSI.

IMS.DLIBZONE.CSI data set

The DLIBZONE (distribution, or DLIB, zone) data set records information about the status and structure of the distribution libraries. You assign each distribution zone a one- to seven-character name when you create the zone. This name appears in the SET BDY command.

The DLIBZONE data set has the following attribute:

DSORG
VSAM KSDS

IMS.GLBLZONE.CSI data set

The GLBLZONE (global zone) data set contains information about SYSMODS and HOLDDATA that have been processed by the SMP/E RECEIVE command. The data set also contains information that allows SMP/E to access the DLIBZONE and TRGTZONE, and information that allows you to tailor parts of SMP/E processing.

The GLBLZONE data set has the following attribute:

DSORG
VSAM KSDS

IMS.SMPLTS data set

The SMPLTS data set is a target library that maintains the base version of a load module. The load module specifies a SYSLIB allocation to implicitly include modules. A base version of a load module includes only the explicitly defined modules for the load module. The data set is maintained in the SMPLTS if the load module is defined to SMP/E with a SYSLIB allocation (that is, its LMOD entry contains a CALLLIBS subentry list). SMP/E uses the load module in the SMPLTS as input when binding the load module into its specified target libraries.

Each target zone must have its own SMPLTS data set. The SMPLTS cannot be shared with any other target zone.

The SMPLTS data set has the following attributes:

DSORG
Partitioned data set extended (PDSE)

DSNTYPE
LIBRARY

RECFM
U

LRECL
0

BLKSIZE
Greater than or equal to 6144

IMS.SMPPTS data set

The SMPPTS data set is used as temporary storage for SYSMODs. It contains one member for each SYSMOD that is received.

This data set has the following attributes:

DSORG
Partitioned

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SMPSCDS data set

The SMPSCDS data set contains backup copies of target zone entries that are changed by inline JCLIN during APPLY processing.

Each target zone must have its own SMPSCDS data set. The SMPSCDS cannot be shared by any other target zone.

This data set has the following attributes:

DSORG
Partitioned

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SMPSTS data set

The SMPSTS data set is a temporary target source library for source modules that exist only in a distribution library.

Each target zone must have its own SMPSTS data set. The SMPSTS cannot be shared by any other target zone.

This data set has the following attributes:

DSORG
Partitioned

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.TRGTZONE.CSI data set

The TRGTZONE (target zone) data set records information about the status and structure of the target libraries. You assign each target zone a one to seven-character name when you create it. This name appears in the SET BDY command.

Each TRGTZONE data set must have its own SMPLTS, SMPMTS, SMPSTS, and SMPSCDS data sets. Each TRGTZONE can support only one release of a given product. Products having the same SMP/E SREL (P115 for IMS) are eligible for sharing the same SMP/E TRGTZONE. However, this practice is not recommended.

The TRGTZONE data set has the following attribute:

DSORG
VSAM KSDS

Related concepts
z/OS: SMP/E data sets and files

Distribution (DLIB) data sets

IMS distribution libraries (DLIBs) contain the master copy of elements in IMS. They can be used to restore SYSMODs in the target library or to rebuild a target environment. These data sets are maintained by SMP/E.

- "System services data sets" on page 49
- "Database Manager data sets" on page 54
- "Transaction Manager data sets" on page 54
- "IMS Extended Terminal Option Support data sets" on page 54
- "IMS Java On Demand features data sets" on page 54

System services data sets

The following DLIBs are used by the System Services component FMID:

IMS.ADFSBASE
IMS.ADFSCLST
IMS.ADFSDATA
IMS.ADFSEXEC
IMS.ADFSISRC
IMS.ADFSJJCL
IMS.ADFSLOAD
IMS.ADFSMAC
IMS.ADFSMLIB
IMS.ADFSPLIB
IMS.ADFSPROC
IMS.ADFSRTRM
IMS.ADFSSLIB
IMS.ADFSSMPL
IMS.ADFSSRC
IMS.ADFSTLIB

IMS.ADFSBASE: The ADFSBASE data set contains SMP/E sample jobs to install IMS.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.ADFSCLST: ADFSCLST contains TSO CLISTs.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.ADFSDATA: ADFSDATA contains data.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.ADFSEXEC: ADFSEXEC contains TSO REXX EXECs.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.ADFSISRC: ADFSISRC contains DBRC skeletal JCL members, a sample application, and miscellaneous source elements.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.ADFSLOAD: ADFSLOAD contains individually linked load modules.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE

RECFM
U

LRECL
0

BLKSIZE
Greater than or equal to 6144.

IMS.ADFSMAC: ADFSMAC contains system definition macros, utility macros, and the macros required for IMS module assembly.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80: the block size for this data set should be greater than or equal to the larger of the SYS1.AMACLIB and SYS1.AMODGEN block sizes.

To prevent DCB conflicts during IMS system definition and SMP/E processing, make the block sizes for ADFSMAC and OPTIONS the same.

IMS.ADFSMLIB: ADFSMLIB contains ISPF dialog message members.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.ADFSPLIB: ADFSPLIB contains ISPF dialog panels.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.ADFSPROC: ADFSPROC contains SMP/E sample jobs to install IMS.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.ADFSRTRM: ADFSRTRM contains description members that are used by the IVP dialog.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.ADFSSLIB: ADFSSLIB contains ISPF dialog file-tailoring skeletons.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.ADFSSMPL: ADFSSMPL contains sample jobs and exits.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.ADFSSRC: ADFSSRC contains source modules for the IMS DB licensed program, the System Services component, and the Transaction Manager licensed program.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.ADFSTLIB: ADFSTLIB contains ISPF dialog tables.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

Database Manager data sets

The following DLIBs are used by the Database Manager FMID:

IMS.ADFSCLST
IMS.ADFSLOAD
IMS.ADFSPLIB
IMS.ADFSPROC
IMS.ADFSSRC
IMS.ADFSSMPL

Transaction Manager data sets

The following DLIBs are used by the Transaction Manager FMID:

IMS.ADFSEXEC
IMS.ADFSLOAD
IMS.ADFSPLIB
IMS.ADFSPROC
IMS.ADFSSMPL
IMS.ADFSSRC

IMS Extended Terminal Option Support data sets

The IMS Extended Terminal Option Support FMID uses the IMS.ADFSLOAD DLIB.

IMS Java On Demand features data sets

IMS Java On Demand features use the following DLIB data sets:

IMS.ADFSJLIB
IMS.ADFSIC4J
IMS.ADFSJHFS
IMS.ADFSJRAR
IMS.ADFSJSAM
IMS.ADFSJCIC

IMS.ADFSJLIB: ADFSJLIB contains the C code library that is used by the classic type-2 driver.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE

RECFM
U

LRECL
0

BLKSIZE
32760

IMS.ADFSIC4J: ADFSIC4J contains the IMS TM resource adapter runtime component for the z/OS platform.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS or PDSE

RECFM
VB

LRECL
255

BLKSIZE
Greater than or equal to 259.

IMS.ADFSJHFS: ADFSJHFS contains the type-2 and type-4 IMS Universal driver Java class libraries used for IMS TM message queue and transaction processing and for IMS DB access through the JDBC and DLI for Java interfaces.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS or PDSE

RECFM
VB

LRECL
255

BLKSIZE
Greater than or equal to 259.

IMS.ADFSJRAR: ADFSJRAR contains the Universal type-2 and type-4 JCA drivers for direct IMS DB access from a Java EE runtime environment (such as WebSphere® Application Server for z/OS).

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS or PDSE

RECFM
VB

LRECL
255

BLKSIZE
Greater than or equal to 259.

IMS.ADFSJSAM: ADFSJSAM contains the Open Database sample (class files only).

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS or PDSE

RECFM
VB

LRECL
255

BLKSIZE
Greater than or equal to 259.

IMS.ADFSJCIC: ADFSJCIC contains code that is required to access IMS when using the IMS Database JDBC driver in an IMS Connect environment.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS or PDSE

RECFM
VB

LRECL
255

BLKSIZE
Greater than or equal to 259.

IMS.ADFSJJCL: ADFSJJCL contains side decks to link the C code.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
32720

Target (TLIB) data sets

The TLIB data sets are the IMS SMP/E target libraries (SYSLIBs), and are the libraries that are used to run and use IMS.

IMS data sets maintained by SMP/E

The following data sets are maintained by the SMP/E APPLY processing:

IMS.MODBLKS
IMS.SDFSBASE
IMS.SDFSCLST
IMS.SDFSDATA
IMS.SDFSEXEC
IMS.SDFSISRC
IMS.SDFSJLIB
IMS.SDFSJSID
IMS.SDFSMAC
IMS.SDFSMLIB
IMS.SDFSPLIB
IMS.SDFSPROC
IMS.SDFSRESL
IMS.SDFSRTRM
IMS.SDFSSLIB
IMS.SDFSSMPL
IMS.SDFSSRC
IMS.SDFSTLIB

The following data sets that reside in a UNIX System Services file system are also maintained by the SMP/E APPLY processing:

SDFSJCPS
SDFSJTOL
SDFSIC4J
SDFSJCIC
SDFSJCPI
SDFSJHFS
SDFSJRAR
SDFSJSAM

IMS system definition data sets

The following data sets are initially loaded or updated by Stage 2 of the IMS system definition process:

IMS.MODBLKS
IMS.SDFSRESL

IMS.MODBLKS

MODBLKS contains the control block modules that are created by IMS system definition. Its contents are copied by the Online Change utility to either IMS.MODBLKSA or IMS.MODBLKSB.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
U

LRECL
0

BLKSIZE
Less than or equal to 32 760. Default 32 760. IMS.SDFSRESL, MODBLKS, MODBLKSA, and MODBLKSB should have the same BLKSIZE.

IMS.SDFSBASE

SDFSBASE is the target library for ADFSBASE and contains sample jobs.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SDFSCLST

SDFSCLST is the target library for ADFSCLST and contains TSO CLISTs.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SDFSDATA

SDFSDATA is the target library for ADFSDATA and contains data.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SDFSEXEC

SDFSEXEC is the target library for ADFSEXEC and contains TSO REXX EXECs.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SDFSISRC

SDFSISRC is the target library for ADFSISRC and contains DBRC skeletal JCL members, and sample application and miscellaneous source modules.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SDFSJLIB

SDFSJLIB contains the bind output for the IMS Java On Demand Features load modules. It must be APF authorized.

This data set has the following attributes:

DSORG
Partitioned data set extended (PDSE)

DSNTYPE
LIBRARY

RECFM
U

LRECL
0

BLKSIZE
Less than or equal to 32 760. Default 32 760.

IMS.SDFSJSID

SDFSJSID is the target library for ADFSJJCL and contains side decks.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SDFSMAC

IMS.SDFSMAC is the target library for ADFSMAC, and it contains the IMS macros.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80: the BLKSIZE for this data set must be greater than or equal to the larger of the SYS1.AMACLIB and SYS1.AMODGEN BLKSIZEs.

The block sizes for SDFSMAC and OPTIONS should be the same to prevent DCB conflicts during IMS system definition and SMP/E processing.

IMS.SDFSMLIB

SDFSMLIB is the target library for ADFSMLIB and contains ISPF dialog message members.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SDFSPLIB

SDFSPLIB is the target library for ADFSPLIB and contains ISPF dialog panels.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SDFSPROC

SDFSPROC is the target library for ADFSPROC and contains sample jobs.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SDFSRESL

IMS.SDFSRESL contains the IMS nucleus and required action modules. This data set is built by a combination of system definition and SMP/E APPLY processing.

IMS.SDFSRESL must reside on DASD that supports a maximum record size of 18 KB or greater.

APF-authorize the IMS.SDFSRESL data set and any data set that is concatenated to it on JOBLIB or STEPLIB DD statements.

For IMS batch, APF-authorize the IMS.SDFSRESL data set and any data set concatenated to it on the DFSRESLB DD statement. This DD statement provides an authorized library for the IMS SVC modules. You do not need to authorize the JOBLIB or STEPLIB statement for IMS batch. If you omit the DFSRESLB DD

statement, the IMS SVC modules are loaded from JOBLIB or STEPLIB, and JOBLIB or STEPLIB data sets must be APF-authorized.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
U

LRECL
0

BLKSIZE
Less than or equal to 32 760. Default is 32 760. IMS.SDFSRESL, IMS.MODBLKS, IMS.MODBLKSA, and IMS.MODBLKSB must have the same BLKSIZE.

IMS.SDFSRTRM

SDFSRTRM is the target library for ADFSRTRM and contains description members used by the IVP dialog.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SDFSSLIB

SDFSSLIB is the target library for ADFSSLIB and contains ISPF dialog file tailoring skeletons.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SDFSSMPL

SDFSSMPL is the target library for ADFSSMPL and contains sample jobs and exits.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SDFSSRC

SDFSSRC is the target library for ADFSSRC and contains source programs.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SDFSTLIB

SDFSTLIB is the target library for ADFSTLIB and contains ISPF dialog tables.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDSE or PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80: INSTATBL and SDFSTLIB must have the same BLKSIZE.

System data sets

The system data sets are IMS system libraries. These data sets are user data sets (not known to SMP/E).

IMS file system path names

IMS uses the following paths that reside in a z/OS file system:

SDFSJCIC: Maps to PathPrefix/usr/lpp/ims/ims15/imsjava/cics/IBM/
SDFSJHFS: Maps to PathPrefix/usr/lpp/ims/ims15/imsjava/IBM/
SDFSJSAM: Maps to PathPrefix/usr/lpp/ims/ims15/imsjava/samples/IBM/
SDFSJRAR: Maps to PathPrefix/usr/lpp/ims/ims15/imsjava/IBM/
SDFSIC4J: Maps to PathPrefix/usr/lpp/ims/ims15/ico/IBM/

IMS SYSDEF data sets

The following data sets are initially loaded by Stage 2 of the IMS system definition (SYSDEF) process.

IMS.FORMAT
IMS.OBJDSET
IMS.PROCLIB
IMS.REFERAL
IMS.TFORMAT

JOBS data sets

JOBS data sets include various IMS jobs.

MODBLKS data sets

The IMS control region and the MSVERIFY utility use IMS.MODBLKS data sets that contain the IMS system definition output for the control block modules that are affected by online change. The MODBLKS data sets include:

IMS.MODBLKS
IMS.MODBLKSA
IMS.MODBLKSB

RDDS data sets

Resource definition data sets (RDDS) contain resource definitions and resource descriptor definitions for the MODBLKS data sets. The RDDS data sets include:

IMS.RDDS*nn*

A minimum of 2 RDDSs are required to support the export function.

TCFSLIB data sets

TCFSLIB data sets contain TCO SCRIPTS.

IMS.JOBS data set

JOBS contains job streams that are submitted for execution by either the IMS operator command: `/START REGION` or the z/OS command **`START IMSRDR,MBR=`**. You must customize any jobs that are stored in this data set with your installation job names, job statement parameters, and other pertinent specifications. This data set also contains the RACF password or user ID (on a job statement), and therefore must be read protected. You can assign a RACF password and user ID to this data set, and optionally code a RACF System Task Authorization exit routine to verify the use of protected data sets. Otherwise, system security cannot be assured.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.MODBLKSA, IMS.MODBLKSB data sets

The MODBLKSA and MODBLKSB data sets contain MODBLKS members. When one of these libraries is active (in use by the online system), the contents of IMS.MODBLKS are copied to the other, or inactive, library for use during the next online change process.

IMS.MODBLKSA or IMS.MODBLKSB can be brought online by a sequence of master terminal operator `/MODIFY` commands.

Before running online, you should APF-authorize these data sets to the z/OS system.

The MODBLKS data sets can be defined as either partitioned data sets (PDSs) or partitioned data sets extended (PDSEs). If a MODBLKS data set is defined as a PDS, the size of a member of the data set is limited to 16 M. Alternatively, if a MODBLKS data set is defined as a PDSE, the size of a member of the data set is up to 2G in size. Whether a MODBLKS data set is defined as a PDS or a PDSE depends on the required number and size of the following resources for your environment:

- Database, as defined by DDIRLEN in macro DFSDDIR
- Program, as defined by PDIRLEN in macro DFSPDIR
- Routing code length, as defined by RCTELEN in macro DBFRCTE
- Transaction, as defined by SMBTOTSZ in macro IAPS

To define your MODBLKS data sets as PDSEs, specify **DSNTYPE=LIBRARY** on your allocation job.

These data sets have the following attributes:

DSORG
Partitioned

DSNTYPE
PDS or PDSE

RECFM
U

LRECL
0

BLKSIZE
Greater than or equal to 32760. Default 32760. IMS.SDFSRESL, MODBLKS, MODBLKSA, and MODBLKSB should have the same BLKSIZE.

IMS.OBJDSET data set

The OBJDSET data set contains the assembler output that is created during IMS system definition Stage 2 execution. You specify the name of this data set in the IMSGEN macro.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80 less than or equal to 3200. This BLKSIZE limit of 3200 is a binder-imposed maximum for data sets containing object modules that are referenced by INCLUDE.

IMS PROCLIB data set

The IMS PROCLIB data set contains the cataloged procedure and control statement members. Some control statement members are created by IMS system definition. Other control statement members and procedures are generated by SMP processing and users might copy them from the IMS.SDFSPROC data set. It also contains user-created control statement members that are used to tailor IMS. After system definition, you might need to move some procedures to SYS1.PROCLIB.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS or PDSE

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.TCFSLIB data set

The TCFSLIB data set contains control statement members (scripts) used by IMS time-controlled operations (TCO).

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
F

LRECL
80

BLKSIZE
80

Execution data sets

The execution data sets are used during the execution of the IMS system and its related utilities. These data sets are user data sets which are not known to SMP/E.

ACBLIB data sets

The ACBLIB data sets contain the application description and database control blocks. The ACBLIB data sets include:

IMS.ACBLIB
IMS.ACBLIBA
IMS.ACBLIBB

The ACBLIB data sets require space for each PSB and all unique physical DBDs.

In systems that share data, the ACBLIBs in both systems must be identical, or the systems must share the same ACBLIB.

DBDLIB data set

The IMS.DBDLIB data set contains the database description blocks (DBDs) that are created by the Database Description Generation (DBDGEN) utility.

DBRC RECON data sets

The RECON data sets contain system restart and recovery information. The RECON data sets include:

IMS.RECON1
IMS.RECON2
IMS.RECON3

FORMAT data sets

The FORMAT data sets contain MFS definitions. The FORMAT data sets include:

IMS.FORMAT
IMS.FORMATA
IMS.FORMATB
IMS.REFERAL
IMS.TFORMAT

IMS.REFERAL,IMS.FORMAT, and IMS.TFORMAT are initialized during stage 2 of IMS system definition. IMS.FORMATA and IMS.FORMATB are created by copying the staging library, IMS.FORMAT. You must allocate one additional track for each user-defined format/message descriptor set for the IMS.FORMAT, IMS.REFERAL, and IMS.TFORMAT data sets.

Log data sets

The log data sets include:

IMS.DFSOLPnn
IMS.DFSOLSnn
IMS.DFSWADSn
IMS.IEFRDER

IMS.IEFRDER2
IMS.IMSMON
IMS.MSDBCP1
IMS.MSDBCP2
IMS.MSDBCP3
IMS.MSDBCP4
IMS.RDS
IMS.RDS2

Message queue data sets

The message queue data sets are used for message queuing. The message queue data sets include:

IMS.LGMSG
IMS.LGMSG1-LGMSG9
IMS.LGMSGL
IMS.MODSTAT
IMS.QBLKS
IMS.QBLKSL
IMS.SHMSG
IMS.SHMSG1-SHMSG9
IMS.SHMSGL

MSDB data sets

MSDB data sets contain information associated with MSDB databases. The MSDB data sets include:

- IMS.MSDBCP1
- IMS.MSDBCP2
- IMS.MSDBCP3
- IMS.MSDBCP4
- IMS.MSDBDUMP
- IMS.MSDBINIT

Online change data sets

The online change data sets include:

- IMS.MODSTAT
- IMS.MODSTAT2
- IMSPLEX.OLCSTAT

PGMLIB data sets

The IMS.PGMLIB data set contains user-written application programs. This dataset can be a PDS or a PDSE.

Restriction: COBOL5 requires all load modules to be in a PDSE.

PSBLIB data sets

The IMS.PSBLIB data set contains the program specification blocks (PSBs) created by the Program Specification Block Generation (PSBGEN) utility.

SYSOUT data sets

SYSOUT data sets include:

- IMS.SYSO*nnn* data set
- Direct output data sets

Trace data sets

Trace data sets contain output from IMS internal tracing. The trace data sets include:

- IMS.DFSTRA01
- IMS.DFSTRA02
- IMS.DFSTRA0T

IMS.ACBLIB

ACBLIB contains the application control blocks (ACBs) that are created by the Application Control Blocks Maintenance (ACBGEN) utility. Its contents are copied by the Online Change Copy utility to either the IMS.ACBLIBA data set or the IMS.ACBLIBB data set.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
U

LRECL
0

BLKSIZE
User choice. The default is 6144.

IMS.ACBLIBA, IMS.ACBLIBB

ACBLIBA and ACBLIBB contain ACBLIB members. When one of these libraries is active (in use by the online system), the contents of IMS.ACBLIB are copied to the other, or inactive, library for use in the next online change run.

IMS.ACBLIBA or IMS.ACBLIBB can be brought online by a sequence of master terminal operator `/MODIFY` commands.

If you specify DOPT in the APPLCTN macro, concatenate the library that contains these PSBs after the library that contains the non-DOPT PSBs (that is, after the library pointed to by the IMS.ACBLIBA or IMS.ACBLIBB DD statements). The order of concatenation must be the same for the IMS.ACBLIBA and IMS.ACBLIBB data sets.

These data sets have the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
U

LRECL
0

BLKSIZE
User choice. The default is 6144.

IMS.DBDLIB

DBDLIB contains the database description blocks (DBDs) that are created by the Database Description Generation (DBDGEN) utility. Each DBD (one per database) requires approximately 1500 bytes to 2500 bytes of direct-access storage. Exact requirements depend on the number of data set groups, segments, fields, and hierarchic levels.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
U

LRECL
0

BLKSIZE
User choice. The default is 6144.

IMS.DFSOLP*nn*, v.DFSOLS*nn*

DFSOLP*nn* and DFSOLS*nn* are the online log data sets (OLDS) that are used by the IMS online systems. OLDSs can occur singly (SNGL) or in pairs (DUAL). DFSOLP*nn* is the primary (or SNGL) OLDS. DFSOLS*nn* is the secondary OLDS. The *nn* suffix can range from 00 to 99. A minimum of 3 OLDSs (SNGL or DUAL) must be available to start IMS. The OLDSs do not support compression.

These data sets have the following attributes:

DSORG
Sequential

RECFM
VB

LRECL
BLKSIZE-4

BLKSIZE
Multiple of 2048 greater than or equal to 6144. These data sets must be allocated as a single extent (contiguous tracks). Secondary allocation must not be specified. Choose a block size that results in from 1 to 4 blocks per track.

IMS.DFSTRA01, IMS.DFSTRA02

DFSTRA01 and DFSTRA02 are the external trace data sets that are used by the IMS online systems. The two data sets are used when the trace table OUT parameter is used in the DFSVSMxx OPTIONS statement or when the `/TRACE SET ON TABLE nnn OPTION LOG` command is used. When DFSTRA01 fills, DFSTRA02 is used; when DFSTRA02 fills, DFSTRA01 is used.

These data sets have the following attributes:

DSORG
Sequential

RECFM
VB

LRECL
4016

BLKSIZE
(LRECL x n) + 4. The block size must be a multiple of the LRECL (4016), with an additional 4 bytes for the block descriptor word.

Recommendation: Use a BLKSIZE of 20 084, which is 5 logical records (4016 x 5) plus the block descriptor word (4). The BLKSIZE of 20 084 is recommended for current DASD, because it is 1/2 track.

These data sets must be allocated as a single extent (contiguous tracks). Secondary allocation must not be specified.

IMS.DFSTRA0T

If you use tape for the external trace data set, you must use DFSTRA0T instead of DFSTRA01 and DFSTRA02.

DFSTRA0T must be dynamically allocated.

This data set has the following attributes:

DSORG
Sequential

RECFM
VB

LRECL
4016

BLKSIZE
(LRECL x n) + 4

IMS.DFSWADS*n*

DFSWADSn are the write-ahead data sets (WADS) used by the IMS online systems. WADS can occur singly (SNGL) or in pairs (DUAL), but primary or secondary concepts do not apply as they do with OLDS. The *n* suffix can range from 0 to 9. To start IMS, a minimum of 1 WADS must be available.

These data sets have the following attributes:

DSORG
VSAM linear

RECFM
F

LRECL
4096

BLKSIZE
4096

These data sets must be allocated as a single extent (contiguous tracks). Secondary allocation must not be specified.

IMS.FORMAT

FORMAT contains the Message Format Service (MFS) blocks created by the MFS Language utility. Its contents are copied by the Online Change Utility to either IMS.FORMATA or IMS.FORMATB.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
U

LRECL
0

BLKSIZE
User choice. Default 6144. The FORMAT, FORMATA, FORMATB, and TFORMAT data sets must all have the same BLKSIZE.

IMS.FORMATA, IMS.FORMATB

FORMATA and FORMATB contain FORMAT members. When one of these libraries is active (in use by the online system), the contents of IMS.FORMAT are copied to the other, or inactive, library for use in the next online change run.

IMS.FORMATA or IMS.FORMATB can be brought online by a sequence of master terminal operator `/MODIFY` commands.

These data sets have the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
U

LRECL
0

BLKSIZE
User choice. The default is 6144. The FORMAT, FORMATA, FORMATB, and TFORMAT data sets must all have the same BLKSIZE.

IMS.IEFRDER, IMS.IEFRDER2

IEFRDER typically refers to the primary IMS batch log. IEFRDER2 typically refers to the secondary IMS batch log. They can also refer to the input data set in the IMSRDR procedure.

In batch logging, these data sets have the following attributes:

DSORG
Sequential

RECFM
VB

LRECL
BLKSIZE-4

BLKSIZE
User choice between 4 KB and 32 KB

Recommendation: Choose a block size that is a 2 KB multiple greater than or equal to 6 KB.

IMS.IMSDALIB

IMSDALIB defines the non-authorized partitioned data set (PDS) in which members to be dynamically allocated can be stored. Examples include RECON definitions, database definitions, and anything that is in a DFSMDA member.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
U

LRECL
0

BLKSIZE
User choice

Note: The default is 6144.

IMS.IMSMON

IMSMON contains the trace records for either the DB Monitor or IMS (System) Monitor if the trace records are not routed to the IMS log.

This data set has the following attributes:

DSORG
Sequential

RECFM
VB

LRECL
BLKSIZE-4

BLKSIZE
User choice

Recommendation: Choose a block size that is a 2 KB multiple greater than or equal to 6 KB.

IMS.MODSTAT

MODSTAT contains information to indicate which of the following suffixed data sets the IMS online system must use at initialization time. MODSTAT must be the ddname for these data sets.

ACBLIBA or ACBLIBB
FORMATA or FORMATB
MODBLKSA or MODBLKSB

This data set has the following attributes:

DSORG
Sequential

RECFM
F

LRECL
80

BLKSIZE
80

This data set is a single-record BSAM data set and requires one track of storage.

Before the IMS system can be run, you need to initialize IMS.MODSTAT.

IMS.MODSTAT2

MODSTAT2 is used only in an XRF complex; this data set is identical in function to IMS.MODSTAT. Its ddname must be MODSTAT2. As with MODSTAT, you need to initialize this data set before the IMS system can run.

This data set has the following attributes:

DSORG
Sequential

RECFM
F

LRECL
80

BLKSIZE
80

This data set is a single-record BSAM data set and requires one track of storage.

IMS.MSDBCP1, IMS.MSDBCP2

MSDBCP1 and MSDBCP2 are required if MSDBs are defined to the system. During each IMS checkpoint, a control record followed by the contents of the contiguous block of virtual storage occupied by the MSDBs is written to one of these data sets. The data sets are used alternately by successive checkpoints, with each subsequent checkpoint overlaying a previous one.

These data sets have the following attributes:

DSORG
Sequential

RECFM
Determined by IMS

LRECL
Determined by IMS

BLKSIZE
Determined by IMS

These data sets must be allocated as a single extent (contiguous tracks). Secondary allocation must not be specified.

IMS.MSDBCP3, IMS.MSDBCP4

MSDBCP3 and MSDBCP4 are used only in an XRF complex; these data sets are identical in function to MSDBCP1 and MSDBCP2. With XRF, any two of the four data sets can contain the latest MSDB checkpoint.

Although an active subsystem can select the data set containing the latest MSDB checkpoint and any other, the alternate subsystem must select the two data sets not used by the active subsystem.

These data sets have the following attributes:

DSORG
Sequential

RECFM
Determined by IMS

LRECL
Determined by IMS

BLKSIZE
Determined by IMS

These data sets must be allocated as a single extent (contiguous tracks). Secondary allocation must not be specified.

IMS.MSDBDUMP

MSDBDUMP is required when the command /DBDUMP specifies database MSDB. This command causes a dump of all MSDBs to be written to this data set. The contents are identical to that of MSDBCPx. Successive executions of the command cause the previous contents to be overlaid.

This data set has the following attributes:

DSORG
Sequential

RECFM
Determined by IMS

LRECL
Determined by IMS

BLKSIZE
Determined by IMS

This data set must be allocated as a single extent (contiguous tracks). Secondary allocation must not be specified.

IMS.MSDBINIT

MSDBINIT is required for an IMS system that includes MSDBs. This data set contains a record for each MSDB segment. It is read during all cold starts and during a normal restart if the MSDBLOAD parameter is specified for the /NRESTART command. It is produced by executing the MSDB Dump Recovery or MSDB Maintenance utility. MSDBINIT can contain one, several, or all MSDBs defined.

This data set has the following attributes:

DSORG
Sequential

RECFM
VBT

LRECL
BLKSIZE-4

BLKSIZE
User choice

This data set must be allocated as a single extent (contiguous tracks). Secondary allocation must not be specified.

IMS.PGMLIB

PGMLIB contains user-written application programs.

This data set has the following attributes:

DSORG
Partitioned or Partitioned Extended (PDSE)

RECFM
U

LRECL
0

BLKSIZE
User choice. Default 6144.

IMSPLEX.OLCSTAT

OLCSTAT is an optional data set that contains global online change information and status. OLCSTAT is a global data set that is dynamically allocated by IMS. The MODSTAT and MODSTAT2 data sets do not need to be defined in the IMS control region JCL when OLCSTAT is used.

To enable global online change, OLCSTAT must be defined instead of the local MODSTAT data set. All IMS systems in an IMSplex must define the same physical OLCSTAT data set. Otherwise, IMS initialization fails. OLCSTAT is required if OLC=GLOBAL is defined.

To initialize the OLCSTAT data set, run the Global Online Change utility, DFSUOLC0.

This data set has the following attributes:

DSORG
Sequential

RECFM
V

LRECL
5204

BLKSIZE
Default 5208

IMS.PSBLIB

PSBLIB contains the program specification blocks (PSBs) created by the PSBGEN utility. Each PSB (one per program) requires approximately 250 bytes to 500 bytes of direct-access storage. Exact requirements depend on the number of I/O PCBs and databases (PCBs) in the PSB and the number of sensitive segments. This data set is required in DB and DB/DC systems.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
U

LRECL
0

BLKSIZE
User choice. Default 6144.

IMS.QBLKS, IMS.SHMSG/1-9, IMS.LGMSG/1-9

QBLKS, SHMSG, and LGMSG are required by the IMS DB/DC system for message queuing. Space requirements for message queue data sets vary with the system environment.

These data sets have the following attributes:

DSORG
Sequential

RECFM
Determined by IMS

LRECL
Determined by IMS

BLKSIZE
Determined by IMS

These data sets must be allocated as a single extent (contiguous tracks). Secondary allocation must not be specified.

For SHMSG and LGMSG, up to 10 data sets can be provided for each. Multiple message queue data sets provide for configuration flexibility and performance.

If you use multiple data sets, you must do the following:

- Add the data sets in sequence, with SHMSG or LGMSG specified first.
- Specify the same space allocation for all data sets. Even if you allocate different amounts for multiple data sets, the smallest amount specified is the amount used for all data sets. For example, if four data sets are allocated with 600, 600, 500, and 400 cylinders respectively, the actual total available space is 1600 cylinders (4 X 400), rather than 2100 cylinders (the sum of the allocated amounts). Records are assigned to the data sets cyclically; thus, the smallest space allocated controls the amount of space for all, which in turn determines the total space available and the highest valid record number.

The DD names for the data sets must be:

- For SHMSG:
  - SHMSG
  - SHMSG1
  - SHMSG2
  - SHMSG3
  - SHMSG4
  - SHMSG5
  - SHMSG6
  - SHMSG7
  - SHMSG8
  - SHMSG9
- For LGMSG:
  - LGMSG
  - LGMSG1
  - LGMSG2
  - LGMSG3
  - LGMSG4

- LGMSG5
- LGMSG6
- LGMSG7
- LGMSG8
- LGMSG9

IMS.QBLKSL, IMS.SHMSGL, IMS.LGMSGL

QBLKSL, SHMSGL, and LGMSGL are used only in an XRF complex; these data sets are similar in function to the regular message queue data sets. These data sets are always cold started and used as local message queues on an XRF alternate subsystem, from startup until completion of takeover, when the regular message queues become available. The DCB specification for the local message queue data sets must match the regular message queue data sets. However, the local message queues can be much smaller. The local message queues must be large enough to hold the shutdown message margin, plus primary and secondary IMS master terminal messages until they are dequeued.

These data sets have the following attributes:

DSORG
Sequential

RECFM
Determined by IMS

LRECL
Determined by IMS

BLKSIZE
Determined by IMS

These data sets must be allocated as a single extent (contiguous tracks). Secondary allocation must not be specified.

IMS.RDS

The restart data set (RDS) contains information required for recovery, including the checkpoint ID table required for restarting IMS. The RDS also contains OSAM and VSAM buffer pool definition changes made by the **UPDATE POOL TYPE(DBAS)** command to be retained across emergency restart. However, RDS does not contain any log records.

This data set has the following attributes:

DSORG
Sequential

RECFM
Determined by IMS

LRECL
Determined by IMS

BLKSIZE
Determined by IMS

The BLKSIZE value is set to the buffer size as specified in the **RDS=** parameter in the DFSPB*xxx* member of the IMS PROCLIB data set or in JCL. The default is 4 KB.

You should allocate a minimum of one cylinder of contiguous tracks to this data set. The data set must be allocated large enough to contain at least 90 blocks reserved for RDS1 components BCPT, RRE, SIDX, and LCRE.

IMS.RDS2

RDS2 is used only in an XRF complex; this data set is identical in function to IMS.RDS.

This data set has the following attributes:

DSORG
Sequential

RECFM
Determined by IMS

LRECL
Determined by IMS

BLKSIZE
Determined by IMS

The BLKSIZE value is set to the buffer size as specified in the **RDS=** parameter in the DFSPB*xxx* member of the IMS PROCLIB data set or in JCL. The default is 4 KB.

You should allocate a minimum of one cylinder of contiguous tracks to this data set. The data set must be allocated large enough to contain at least 90 blocks reserved for RDS1 components BCPT, RRE, SIDX, and LCRE.

Do not manage either RDS with a migration or recall system that might recall the data set to a volume other than the one to which it was originally allocated. If you do so, IMS might be unable to warm start or emergency start the system.

IMS.RECON1, IMS.RECON2, IMS.RECON3

RECON1, RECON2, and RECON3 data sets contain system restart and recovery information managed by the Database Recovery Control (DBRC) function.

These data sets have the following attribute:

DSORG
VSAM KSDS

IMS.REFERAL

REFERAL contains intermediate text copies of descriptions supplied to the MFS Language utility.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IMS.SYSOnnn

The SYSOnnn DASD data sets are used to store spool SYSOUT data. The contents of these data sets can be printed using the SPOOL SYSOUT Print utility. This utility is either scheduled automatically or must be submitted manually, depending upon an option in the LINEGRP system definition macro. *nnn* is a one- to three-digit suffix assigned sequentially by IMS during system definition.

This data set has the following attributes:

DSORG
Sequential

RECFM
UM

These data sets must be allocated as a single extent (contiguous tracks). Secondary allocation must not be specified.

These data sets must be initialized before they are used by IMS. For example, these data sets can be allocated on the SYSUT2 DD statement for the IEBGENER utility. Use DD DUMMY for SYSUT1. Specify DCB attributes for both SYSUT1 and SYSUT2.

IMS.TFORMAT

TFORMAT contains the online MFS descriptors, created by the MFS Language utility, for MFSTEST (test mode) online execution.

This data set must be concatenated in front of FORMATA or FORMATB in the IMSTFMTA or IMSTFMTB DD statements in the IMS execution procedure.

If you change MFS formats online, two DD statements must point to this TFORMAT data set, or the DD statements can point to two separate TFORMAT data sets.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
U

LRECL
0

BLKSIZE
User Choice. Default 6144. The FORMAT, FORMATA, FORMATB, and TFORMAT data sets must all have the same BLKSIZE.

This data set must be allocated as a single extent (contiguous tracks). Secondary allocation must not be specified.

IRLM data sets

The IRLM data sets are the distribution and target libraries associated with the IRLM.

IMS.ADXRLOAD

ADXRLOAD is the IRLM distribution library that contains object modules.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
U

LRECL
0

BLKSIZE
User choice. Greater than or equal to 6144.

IMS.ADXRSAMP

ADXRSAMP is the IRLM distribution library that contains JCL.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.SDXRSAMP

SDXRSAMP is the IRLM target library that contains load modules.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80.

IMS.SDXRRESL

IMS.SDXRRESL is the IRLM target library that contains load modules.

Prior to running online, you should APF authorize IMS.SDXRRESL to the z/OS system.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
U

LRECL
0

BLKSIZE
User choice. Greater than or equal to 32760.

User data set (USER.ISPTABL)

These data sets can be allocated by the user.

Some IMS features use ISPF as a dialog manager and might require the use of a user table data set. The user data set might be required to use some of the features of DFSSPOC, DFSHALDB, and Syntax Checker. The USER.ISPTABL data set must be the only data set allocated to file ISPTABL and must also be in the ISPTLIB concatenation before the IMS.SDFSTLIB data set.

Multiple users cannot use the same USER.ISPTABL data set at the same time. A user can have more than one USER.ISPTABL data set, but can use only one data set at a time.

This data set has the following attributes:

DSORG
Partitioned

DSNTYPE
PDS

RECFM
FB

LRECL
80

BLKSIZE
Multiple of 80

IVP variables

The IVP dialog uses user modifiable variables when creating the JOBs and supporting materials used for the IVP process.

The variables that are presented by the IVP dialog are determined by the environment options you select during the Initialization phase. You can print additional documentation for the IVP variables using the DOC action during the variable-gathering phase of the IVP Dialog.

Use the IVP dialog to obtain current information regarding IVP variables. In the lists in these topics, the variables are presented in the same sequence in which they are used by the IVP dialog.

- “General variables” on page 82
- “Data set allocation variables” on page 94

General variables

The IVP dialog uses these variables for jobs and tasks other than data set allocation.

Name
Title

IXUIVPHQ
IVP - High-level DSNAME qualifier for IVP (IVP) data sets

IXURLMHQ
IVP - High-level DSNAME qualifier for IRLM (RLM) data sets

IXUSMPHQ
IVP - High-level DSNAME qualifier for SMP/E (SMP) data sets

IXUDLBHQ
IVP - High-level DSNAME qualifier for IMS DLIB (DLB) data sets

IXUSYSHQ
IVP - High-level DSNAME qualifier for IMS System (SYS) data sets

IXUEXEHQ
IVP - High-level DSNAME qualifier for Execution (EXE) data sets

IXUUTLHQ
IVP - High-level DSNAME qualifier for utility (UTL) data sets

IXUVSMHQ
IVP - High-level DSNAME qualifier for VSAM (VSM) data sets

IXUSSCLS
SMS - Storage Class

IXUSSCL2
SMS - Storage Class for PRA RECON data sets

IXUSMCLS
SMS - Management Class

IXUSMCL2
SMS - Management Class for PRA RECON data sets

IXUSDCL2
SMS - Data Class for PRA RECON data sets

IXUIVPVS
IVP - VOLSER for IVP (IVP) data sets

IXUDLBVS
IVP - VOLSER for IMS distribution, DLIB, (DLB) data sets

IXUSYSVS
IVP - VOLSER for IMS System, (SYS) data sets

IXUEX1VS
IVP - VOLSER for IMS Execution (EX1) data sets - group 1

IXUEX2VS
IVP - VOLSER for IMS Execution (EX2) data sets - group 2

IXUUTLVS
IVP - VOLSER for utility (UTL) data sets - non-VSAM

IXUUTVVS
IVP - VOLSER for utility (UTL) data sets - VSAM

IXUIVPDT
IVP - Device type for IVP (IVP) data sets

IXUDLBDT
IVP - Device type for IMS Distribution (DLB) data sets

IXUSYSDT
IVP - Device type for IMS System (SYS) data sets

IXUEX1DT
IVP - Device type for IMS Execution (EX1) data sets

IXUEX2DT
IVP - Device type for IMS Execution (EX2) data sets

IXUUTLDT
IVP - Device type for utility (UTL) data sets - non-VSAM

IXUUTVDT
IVP - Device type for utility (UTL) data sets - VSAM

IXUTEMPU
IVP - Device type for temporary data sets

IXUPDSFB
IVP - BLKSIZE for PDSs with RECFM=FB and LRECL=80 - (PFB)

IXUPDSU0
IVP - BLKSIZE for PDSs with RECFM=U and LRECL=0 - (PU0)

IXUSEQVB
IVP - BLKSIZE for RECFM=VB sequential data sets - (SVB)

IXUOBJFB
IVP - BLKSIZE for OBJDSET (STAGE2 assembly output) (OBJ)

IXURESU0
IVP - BLKSIZE for IMS SDFSRESL (RESLIB)

IXUOLDVB
IVP - BLKSIZE for IMS OLDS (Online Log Data Set) (OLD)

IXULOGVB
IVP - BLKSIZE for IMS MONITOR and Batch Logs data sets (LOG)

IXUTRCVB
IVP - BLKSIZE for IMS External Trace data sets (TRC)

IXUVSAMD
IVP - BLKSIZE for VSAM data CIs (VSD)

IXUGZDSN
SMP - Fully Qualified DSNAME - IMS SMP/E Global Zone

IXUTZONE
SMP - Zone ID - IMS SMP/E Target Zone

IXUSPROC
IVP - Fully qualified DSNAME - SYS1.PROCLIB

IXUSMACL
SMP - Fully qualified DSNAME - SYS1.MACLIB (or AMACLIB)

IXUSAMOD
SMP - Fully qualified DSNAME - SYS1.MODGEN (or AMODGEN)

IXUSMACT
SMP - Fully qualified DSNAME - High Level Assembler Toolkit Feature MACLIB

IXUUMAC1
SMP - Fully qualified DSNAME - User Macro Library #1 >>> See description

IXUUMAC2
SMP - Fully qualified DSNAME - User Macro Library #2 >>> See description

IXUUMAC3
SMP - Fully qualified DSNAME - User Macro Library #3 >>> See description

IXULELKD
SMP - Fully Qualified DSNAME - IBM Language Environment® for z/OS Data Set

IXULESPC
SMP - Fully Qualified DSNAME - IBM Language Environment for z/OS Resident DS

IXULECRL
JCL - Fully Qualified DSNAME - System C Runtime Library

IIXUTCPDS
JCL - Fully Qualified DSNAME - TCP/IP Data Set

IXUTCPHN
JCL - TCP/IP Hostname

IXUJESTY
JCL - JES VERSION. (JES2 OR JES3)

IXUUPROC
JCL - User PROCLIB ddname (JES2) or ddname suffix (JES3)

IXUJOBNM
JCL - JOBNAME - USE IVP JOBNAME (Y) OR TSO USERID (N)

IXUJACT1
JCL - JOB statement accounting information - Part 1 of 5

IXUJACT2
JCL - JOB statement accounting information - Part 2 of 5

IXUJACT3
JCL - JOB statement accounting information - Part 3 of 5

IXUJACT4
JCL - JOB statement accounting information - Part 4 of 5

IXUJACT5
JCL - JOB statement accounting information - Part 5 of 5

IXUPGMNM
JCL - JOB statement programmer name

IXUJCLAS
JCL - JOB statement CLASS parameter - IVP JOBs

IXUJCLS2
JCL - JOB statement CLASS parameter - SYSDEF STAGE2 JOBs

IXUMCLAS
JCL - JOB statement MSGCLASS parameter

IXUGROUP
JCL - JOB statement GROUP parameter

IXUUSRID
JCL - JOB statement USER parameter

IXUPASWD
JCL - JOB statement PASSWORD parameter

IXUNOTFY
JCL - JOB statement NOTIFY parameter

IXURGNSZ
JCL - JOB statement REGION parameter (4M or larger)

IXUJTIME
JCL - JOB statement TIME parameter

IXUSTIM1
JCL - EXEC statement TIME parameter for SMP/E, STAGE1, STAGE2

IXUSTIM2
JCL - EXEC statement TIME parameter for DL/I Batch and BMP Jobs

IXUSTIM3
JCL - EXEC statement TIME parameter for MPPs, IFPs, and other programs

IXUJESC1
JCL - JESx statement - 1 of 5

IXUJESC2
JCL - JESx statement - 2 of 5

IXUJESC3
JCL - JESx statement - 3 of 5

IXUJESC4
JCL - JESx statement - 4 of 5

IXUJESC5
JCL - JESx statement - 5 of 5

IXUJPATH
JCL - Path prefix for Java native code

IXUSPATH
JCL - Path prefix for Java sample applications

IXUJHOME
JCL - Path name for JDK installation

IXUJOUT
JCL - Standard output file and path name

IXUJERR
JCL - Standard error file and path name

IXUIMIDB
GEN - IMSID for Batch >>> See description

IXUIMID1
GEN - IMSID for DB/DC (and DB/DC with XRF) >>> See description

IXUIMID2
GEN - IMSID for DB/DC with XRF >>> See description

IXUIMID3
GEN - IMSID for DBCTL >>> See description

IXUIMID4
GEN - IMSID for DCCTL >>> See description

IXUCRC1
GEN - Command Recognition Character (CRC) for CCTL - IVP1

IXUCRC2
GEN - Command Recognition Character (CRC) for CCTL - IVP2

IXUCRC3
GEN - Command Recognition Character (CRC) for CCTL - IVP3

IXUSVCT2
GEN - IMS Type 2 SVC

IXUSVCT4
GEN - IMS Type 4 SVC (for DBRC)

IXURLSS
IVP - IRLM Subsystem Names

IXURLNM1
IVP - IRLM #1 JOBNAME

IXUIMNM1
IVP - IMS DB/DC JOBNAME and PROC name for system IVP1

IXUIMNM2
IVP - IMS DB/DC JOBNAME and PROC name for system IVP2

IXUIMNM3
IVP - IMS DBCTL JOBNAME and PROC name for system IVP3

IXUIMNM4
IVP - IMS DCCTL JOBNAME and PROC name for system IVP4

IXURCNM1
GEN - DBRC procedure name for system IVP1

IXURCNM2
IVP - DBRC procedure name for system IVP2

IXURCNM3
GEN - DBRC procedure name for system IVP3

IXURCNM4
GEN - DBRC procedure name for system IVP4

IXUDLNM1
GEN - DLISAS procedure name for system IVP1

IXUDLNM2
IVP - DLISAS procedure name for system IVP2

IXUDLNM3
GEN - DLISAS procedure name for system IVP3

IXUPRDR1
GEN - IMSRDR procedure name for system IVP1

IXUPRDR2
IVP - IMSRDR procedure name for system IVP2

IXUPRDR3
IVP - IMSRDR procedure name for system IVP3

IXUPRDR4
IVP - IMSRDR procedure name for system IVP4

IXUCQSN1
IVP - CQS JOBNAME for system IVP1

IXUSCIN1
IVP - SCI JOBNAME for system IVP1

IXUOMNM1
IVP - OM JOBNAME for system IVP1

IXURMNM1
IVP - RM JOBNAME for system IVP1

IXUMPP11
IVP - MPP #1 - JOBNAME and JOBS member name - IVP1

IXUMPP21
IVP - MPP #1 - JOBNAME and JOBS member name - IVP2

IXUMPP41
IVP - MPP #1 - JOBNAME and JOBS member name - IVP4

IXUIFP11
IVP - IFP #1 - JOBNAME and JOBS member name - IVP1

IXUIFP21
IVP - IFP #1 - JOBNAME and JOBS member name - IVP2

IXUIFP41
IVP - IFP #1 - JOBNAME and JOBS member name - IVP4

IXUIFP12
IVP - IFP #2 - JOBNAME and JOBS member name - IVP1

IXUIFP22
IVP - IFP #2 - JOBNAME and JOBS member name - IVP2

IXUIFP13
IVP - IFP #3 - JOBNAME and JOBS member name - IVP1

IXUJMP11
IVP - JMP #1 - JOBNAME and JOBS member name - IVP1

IXUJMP21
IVP - JMP #1 - JOBNAME and JOBS member name - IVP2

IXUHWSN1
IVP - IMS Connect JOBNAME for system IVP1

IXUTCPJN
IVP - TCP/IP JOBNAME for IMS Connect

IXUTCPPI
IVP - TCP/IP Port ID

IXUCICSP
IVP - Local TCP/IP port to communicate with CICS

IXUKEEPI
IVP - Number of seconds for a TCP/IP KeepAlive interval

IXRHSTNM
JCL - TCP/IP hostname of the remote CICS subsystem

IXRMTCNM
IVP - Remote CICS subsystem name

IXRMTPNB
IVP - Remote CICS subsystem port number

IXUISCID
IVP - Unique ID for this ISC statement

IXUISCND
IVP - ISC node name defined to the local IMS

IXUISCPX
IVP - IMS Connect name that identifies it in the IMSplex

IXUCICSA
IVP - APPLID of the remote CICS subsystem

IXUCICSN
IVP - NETWORK ID of the remote CICS subsystem

IXUXCFGN
IVP - XCF GROUP NAME for IMS OTMA

IXUVAPL1
GEN - VTAM APPLID for system IVP1

IXUVAPL2
GEN - VTAM APPLID for system IVP2

IXUVAPL4
GEN - VTAM APPLID for system IVP4

IXUVPWD1
GEN - VTAM PASSWORD for system IVP1

IXUVPWD2
GEN - VTAM PASSWORD for system IVP2

IXUVPWD4
GEN - VTAM PASSWORD for system IVP4

IXUVNDP1
GEN - VTAM node name for the Master Terminal - IVP1

IXUVNDP2
GEN - VTAM node name for the Master Terminal - IVP2

IXUVNDP4
GEN - VTAM node name for the Master Terminal - IVP4

IXULTNP1
GEN - LTERM name for the Master Terminal

IXULTNS1
GEN - LTERM name for the Secondary Master Terminal

IXUVNDU1
GEN - VTAM node name for IMS User Terminal #1

IXULTNU1
GEN - LTERM name for IMS User Terminal #1

IXUVNDU2
GEN - VTAM node name for IMS User Terminal #2

IXULTNU2
GEN - LTERM name for IMS User Terminal #2

IXUSUFIX
GEN - Character to be assigned as the IMS Nucleus suffix

IXURSENM
IVP - IMS RSE name for XRF

IXURAID1
IVP - RACF defined user ID #1

IXURAID2
IVP - RACF defined user ID #2

IXURAID3
IVP - RACF defined user ID #3

IXURAPSW
IVP - RACF user logon password

IXURAGRP
IVP - RACF defined group

IXUDBRGR
IVP - DBRC Group ID for shared RECONs

IXUIPLEX
IVP - IMSplex name

IXUOBJD
OBJDSET allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXULGNI
LGENIN allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXULGNO
LGENOUT allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUPROC
PROCLIB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUMBKS
MODBLKS allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUMBKA
MODBLKSA allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUMBKB
MODBLKSB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUMTRX
MATRIX allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUMRXA
MATRIXA allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUMRXB
MATRIXB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUPGML
PGMLIB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUPSBL
PSBLIB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUDBDL
DBDLIB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUACBL
ACBLIB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUACBA
ACBLIBA allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUACBB
ACBLIBB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUFMTL
FORMAT allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUFMTA
FORMATA allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUFMTB
FORMATB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUTFMT
TFORMAT allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

```
       00206000  DFSIXV48   N   EXE  SYS  PU0  N
```

IXURFRL
REFERAL allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUMST1
MODSTAT allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUMST2
MODSTAT2 allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXUMON1
IMSMON allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC

IXUMON2
IMSMON2 allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC - IVP2

IXUTRC1
DFSTRA01 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUTRC2
DFSTRA02 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUTRC3
DFSTRA01 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUTRC4
DFSTRA02 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXURDS1
IMSRDS allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXURDS2
IMSRDS2 allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXURCN1
RECON1 allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC

IXURCN2
RECON2 allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC

IXURCN3
RECON3 allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC

IXUOLP0
DFSOLP00 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLP1
DFSOLP01 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLP2
DFSOLP02 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLP3
DFSOLP03 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLP4
DFSOLP04 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLP5
DFSOLP05 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLP9
DFSOLP99 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLS0
DFSOLS00 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLS1
DFSOLS01 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLS2
DFSOLS02 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLS3
DFSOLS03 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLS4
DFSOLS04 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLS5
DFSOLS05 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLS9
DFSOLS99 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUWAD0
DFSWADS0 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUWAD1
DFSWADS1 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUWAD8
DFSWADS8 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUWAD9
DFSWADS9 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUJOB1
JOBS allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUJOB2
JOBS allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR - IVP

IXUTCFS
TCFSLIB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUQBK1
QBLKS allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUQBK2
QBLKS allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUQBL1
QBLKSL allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXUQBL2
QBLKSL allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUSHM1
SHMSG allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUSHM2
SHMSG allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUSHM3
SHMSG1 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUSHM4
SHMSG1 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUSHL1
SHMSGL allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXUSHL2
SHMSGL allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXULGM1
LGMSG allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXULGM2
LGMSG allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXULGM3
LGMSG1 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXULGM4
LGMSG1 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXULGM5
LGMSG2 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXULGM6
LGMSG2 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXULGL1
LGMSGL allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXULGL2
LGMSGL allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUSPL1
SYSO1 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUSP12
SYSO1 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUSPL2
SYSO2 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUSP22
SYSO2 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUSPL3
SYSO3 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUSP32
SYSO3 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUMCP1
MSDBCP1 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUMCP2
MSDBCP2 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUMCP3
MSDBCP3 allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXUMCP4
MSDBCP4 allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXUMDM1
MSDBDUMP allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUMDM2
MSDBDMP2 allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXUODBM
IVP - ODBM JOBNAME

IXUODNM
IVP - ODBM NAME

IXUCSLD
CSLD suffix

IXUBPEC
BPE configuration file used for OPDB

IXUDRDA
DRDA Port ID

IXUCSLC
The suffix of the member CSLDCXXX in the PROCLIB

IXUMCRE
The maximum communication retry

IXUCORM
The maximum amount, in kilobytes, of in-core storage

IXUSAFC
The SAF security class name

IXUVBFN
The number of VSAM buffers in the VSAM LSR pool

IXUVBFS
The size in kilobytes of the VSAM LSR pool buffer

IXUXCFT
The number of XCF listener threads available

IXURCFN
The name of the configuration file for the repository server

IXURSPN
IVP - The name of the repository server

IXUBPER
IVP - The BPE configuration name used for the repository server

IXURCGN
IVP - The XCF group name for repository

IXUIMRN
IVP - The IMS repository name

IXUAUDI
IVP - The audit

IXUAUID
IVP - The audit ID

IXUAFAI
IVP - Start or cancel RS if log stream is unavailable

IXUALOG
IVP - The MVS™ log stream name

IXUALVL
IVP - The audit level

IXUADFL
IVP - The audit default

IXURPNM
IVP - IMSplex group name for the repository

IXURLRI
IVP - The CSLRIxxx suffix used by RM in the repository

IXURRMN_
IVP - The name of the RM used in the repository

IXURLSI
IVP - The CSLSIxxx suffix used by SCI in the repository

IXURCSN
IVP - The name of the SCI used in the repository

IXURLOI
IVP - The CSLOIxxx suffix used by OM in the repository

IXUACVTN
IVP - The asynchronous callout converter name

IXUSCVTN
IVP - The synchronous callout converter name

IXUCATAL
IVP - The alias name to use for the IMS catalog

Related reference
FRPCFG member of the IMS PROCLIB data set (System Definition)

Data set allocation variables

The IVP uses these variables for data set allocation.

Name
Title

IXUOBJD
OBJDSET allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXULGNI
LGENIN allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXULGNO
LGENOUT allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUPROC
PROCLIB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUMBKS
MODBLKS allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUMBKA
MODBLKSA allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUMBKB
MODBLKSB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUMTRX
MATRIX allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUMRXA
MATRIXA allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUMRXB
MATRIXB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUPGML
PGMLIB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUPSBL
PSBLIB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUDBDL
DBDLIB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUACBL
ACBLIB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUACBA
ACBLIBA allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUACBB
ACBLIBB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUFMTL
FORMAT allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUFMTA
FORMATA allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUFMTB
FORMATB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUTFMT
TFORMAT allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXURFRL
REFERAL allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUMST1
MODSTAT allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUMST2
MODSTAT2 allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXUMON1
IMSMON allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC

IXUMON2
IMSMON2 allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC - IVP2

IXUTRC1
DFSTRA01 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUTRC2
DFSTRA02 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUTRC3
DFSTRA01 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUTRC4
DFSTRA02 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXURDS1
IMSRDS allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXURDS2
IMSRDS2 allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXURCN1
RECON1 allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC

IXURCN2
RECON2 allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC

IXURCN3
RECON3 allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC

IXUOLP0
DFSOLP00 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLP1
DFSOLP01 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLP2
DFSOLP02 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLP3
DFSOLP03 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLP4
DFSOLP04 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLP5
DFSOLP05 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLP9
DFSOLP99 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLS0
DFSOLS00 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLS1
DFSOLS01 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLS2
DFSOLS02 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLS3
DFSOLS03 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLS4
DFSOLS04 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLS5
DFSOLS05 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUOLS9
DFSOLS99 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUWAD0
DFSWADS0 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUWAD1
DFSWADS1 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUWAD8
DFSWADS8 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUWAD9
DFSWADS9 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUJOB1
JOBS allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUJOB2
JOBS allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR - IVP

IXUTCFS
TCFSLIB allocation parameters - HLQ,VOL,BLK,TYP,PRM,SEC,DIR

IXUQBK1
QBLKS allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUQBK2
QBLKS allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUQBL1
QBLKSL allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXUQBL2
QBLKSL allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUSHM1
SHMSG allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUSHM2
SHMSG allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUSHM3
SHMSG1 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUSHM4
SHMSG1 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUSHL1
SHMSGL allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXUSHL2
SHMSGL allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXULGM1
LGMSG allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXULGM2
LGMSG allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXULGM3
LGMSG1 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXULGM4
LGMSG1 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXULGM5
LGMSG2 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXULGM6
LGMSG2 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXULGL1
LGMSGL allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXULGL2
LGMSGL allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUSPL1
SYSO1 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUSP12
SYSO1 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUSPL2
SYSO2 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUSP22
SYSO2 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUSPL3
SYSO3 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUSP32
SYSO3 allocation parameters - HLQ,VOL,BLK,TYP,PRM - IVP2

IXUMCP1
MSDBCP1 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUMCP2
MSDBCP2 allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUMCP3
MSDBCP3 allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXUMCP4
MSDBCP4 allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IXUMDM1
MSDBDUMP allocation parameters - HLQ,VOL,BLK,TYP,PRM

IXUMDM2
MSDBDMP2 allocation parameters - HLQ,VOL,BLK,TYP,PRM - XRF

IVP jobs and tasks

This section identifies all the jobs and tasks that can be used during the IVP process.

The jobs and tasks that are presented by the IVP dialog are determined by your choice of environment options and IVP suboptions. The final list in this group, Steps Zx for index of additional PDS members does not identify jobs or tasks in the IVP process. It identifies members of SDFSSLIB and SDFSISRC that support the IVP process. Additional documentation for the IVP jobs and tasks can be printed using the DOC action during either the File Tailoring phase or the Execution phase of the IVP dialog.

Use the IVP dialog to obtain current information regarding IVP jobs and tasks.

In these lists, the jobs and tasks are presented in the same sequence that is used by the IVP dialog. The naming convention used for jobs and tasks is:

IV_*ssnnt*

Where:

- _ - (underscore) identifies the selected environment option:

1 - DBB - Batch
2 - DBC - DBCTL
3 - DBT - DB/DC
4 - XRF - DB/DC with XRF
9 - DCC - DCCTL

- *ss* - identifies the IVP step
- *nn* - a number assigned by IVP that provides a unique name
- *t* - identifies the item type:
  - J - JOB

    A PDS member with the same name is placed into INSTALIB during the file-tailoring phase. Items of type J are intended to be submitted for execution.
  - T - task

    Tasks represent items of work that must be prepared by the user. For some tasks, an example is provided in the INSTALIB data set. These examples are not intended for execution.
  - N - Supporting materials

    The INSTALIB data set can also contain members that support other jobs (such as CLISTs and control statements).

Related concepts

“IVP output ” on page 8
The IVP system provides a rich and diverse set of jobs and tasks that, after completion, provide a fully functional IMS system that is integrated into the z/OS environment and tested in the selected environment from the IVP panels.

Steps Ax for IVP preparation

The items within the A series of steps are used to perform initialization for the IVP dialog.

Name
Title

IV_A001T
NOTE - Introduction - Dialog Set-up

IV_A301N
CLIST - Offline Formatted Dump - IVP1/2/3/4

IV_A302N
CLIST - Offline Dump Formatter - BATCH

IV_A303N
CNTRL - MSDB Load Cntrl Stmts - DBFSAMD1/DBFSAMD2

Steps Cx for system definition (SYSDEF)

The C series of steps include the jobs and tasks that are necessary to perform IMS system definition.

Name
Title

IV_C001T
NOTE - Introduction - System Definition

IV_C101J
JOB - Allocate SYSDEF Data Sets

IV_C105J
JOB- ASM/BIND RACF Security Exits

IV_C201T
TASK - Browse the STAGE1 Source Deck

IV_C202J
JOB - Run SYSDEF Preprocessor

IV_C203J
JOB - Run SYSDEF STAGE1

IV_C301J
JOB - Run SYSDEF STAGE2

IV_C302J
JOB - Assembly/Bind Install. Default Block

IV_C401J
JOB - Run SMP/E JCLIN

IV_C405T
TASK - Edit IMS PROCLIB Members

Steps Dx for interface IMS to z/OS and VTAM

The D series of steps identify the jobs and tasks which you must perform in order to establish the interfaces between IMS and z/OS, and between IMS and VTAM.

Name
Title

IV_D001T
NOTE - Introduction - z/OS and VTAM Interface

IV_D101T
TASK - XMPL - Allocate Interface Data Sets

IV_D102T
TASK - XMPL - Allocate Sharing Control Data Sets (SHCDS)

IV_D200T
TASK - XMPL - Update JESx Procedure

IV_D201T
TASK - XMPL - Update IEAAPFxx or PROGxx - Authorized DSN

IV_D202T
TASK - XMPL - Update IEALPAxx - MLPA Modules

IV_D203T
TASK - XMPL - Update IFAPRDxx - DFSMStvs enablement - PRA

IV_D204T
TASK - XMPL - Update IGDSMSxx - SMS initialization - PRA

IV_D205T
TASK - XMPL - Update IEFSSNxx - SMS & z/OS Resource Recovery Services Subsystem Names

IV_D206T
TASK - XMPL - Update IEFSSNxx - RLM Subsystem Names

IV_D207T
TASK - XMPL - Update IEASVCxx - SVC Numbers

IV_D209T
TASK - XMPL - Install TYPE 2 SVC

IV_D210T
TASK - XMPL - Bind TYPE 4 SVC

IV_D215T
TASK - XMPL - Update BLSCECTX IPCS exits

IV_D216T
TASK - XMPL - IPCS ISPF Data Set Concatenation

IV_D217T
TASK - XMPL - Define z/OS Dump Options

IV_D218T
TASK - XMPL - Define RACF Security Profile

IV_D301T
TASK - XMPL - Define VTAM Application Nodes

IV_D302T
TASK - XMPL - Define VTAM Network Nodes

IV_D303T
TASK - XMPL - Define VTAM Logon Mode Tables

IV_D304T
TASK - XMPL - Define VTAM Interpret Tables

IV_D305T
TASK - XMPL - Define VTAM USS Definition Tables

IV_D306T
TASK - XMPL - Define VTAM Configuration List (ATCCONxx)

IV_D307T
TASK - XMPL - Define VTAM Start Option List (ATCSTRxx)

IV_D308T
TASK - XMPL - Copy VTAM Procedure to SYS1.PROCLIB

IV_D401T
TASK - Restart z/OS with MLPA or CLPA Option

Steps Ex for preparing IVP applications and system

The E series of steps include the jobs and tasks that you must perform to prepare the sample applications and the sample IMS system for execution.

Name
Title

IV_E001T
NOTE - Introduction - Build IVP Appl / System

IV_E101J
JOB - Allocate Data Sets

IV_E102J
JOB - Copy procs and started tasks

IV_E104T
TASK - Modify IMS PROCS

IV_E201J
JOB - DBDGENs

IV_E202J
JOB - PSBGENs

IV_E203J
JOB - ACBGEN

IV_E204J
JOB - MFS Language utility

IV_E206J
JOB - Assembly/Bind Applications

IV_E301J
JOB - Create Dynamic Allocation Members

IV_E302J
JOB - Add Control Statements to IMS.PROCLIB

IV_E303J
JOB - Add SCI/OM/RM Members to IMS.PROCLIB

IV_E304J
JOB - Add CQS Members to IMS.PROCLIB

IV_E305J
JOB - Define EXEC PARM Defaults

IV_E306T
TASK - Syntax Checker Sample

IV_E307T
TASK - Define z/OS Policies

IV_E308J
JOB - Define DRA Start-up Table

IV_E309J
JOB - Verify TCO Scripts

IV_E310J
JOB - Create XRF Procedures

IV_E311T
TASK - Modify IMS PROCs >>> SEE DESCRIPTION

IV_E312J
JOB - Copy STC Procedures to SYS1.PROCLIB

IV_E313J
JOB - Copy Jobs to IMS JOBS

IV_E314J
JOB - Copy DBRC Skeletons to IMS.PROCLIB

IV_E315J
JOB - Assembly/Bind User Exits

IV_E317J
JOB - Initialize MODSTAT

IV_E318J
JOB - Copy Staging Libraries

IV_E319J
JOB - Allocate and load the IMS catalog database

IV_E401T
TASK - Back up System

Steps Fx for IVP execution - DBB system (batch)

The F series of steps include the jobs and tasks that you must perform during the execution of the DBB batch sample system.

Name
Title

IV_F001T
NOTE - Introduction - IVP Execution - DBB

IV_F101J
JOB - Allocate Data Sets

IV_F102J
JOB - Initialize RECON / Register Data Bases

IV_F103J
JOB - Data Base Initial Load

IV_F104J
JOB - Batch Image Copy

IV_F105T
z/OS - Clear z/OS DUMPxx Data Sets

IV_F106J
JOB - Dump Data Base Using DFSDDLT0 (DBBBATCH)

IV_F201J
JOB - Start IRLM

IV_F204J
JOB - FF HIDAM Update

IV_F205J
JOB - FF HDAM Update

IV_F206J
JOB - FF HIDAM Update

IV_F207J
JOB - FF HDAM Update

IV_F208T
TASK - z/OS - Cancel jobs with dump

IV_F209J
JOB - Batch Backout utility - HIDAM Updates

IV_F210J
JOB - Batch Backout utility - HDAM Updates

IV_F211J
JOB - FF HIDAM Update

IV_F212J
JOB - FF HDAM Update

IV_F213T
z/OS - Stop IRLM

IV_F301J
JOB - List RECON data set

IV_F302J
JOB - Print a log with DFDSS

IV_F303J
JOB - Print DB Monitor Report

IV_F305J
JOB - File Select and Print utility

IV_F401J
JOB - Scratch Data Sets

Steps Gx for IVP execution - DBC system (DBCTL)

The G series of steps include the jobs and tasks that you must perform during the execution of the DBC (DBCTL) sample system.

Name
Title

IV_G001T
NOTE - Introduction - IVP Execution - DBC

IV_G101J
JOB - Allocate Data Sets

IV_G102J
JOB - Initialize RECON data set / Register Data Bases

IV_G103J
JOB - Data Base Initial Load

IV_G104J
JOB - Batch Image Copy

IV_G105T
z/OS - Clear z/OS DUMPxx Data Sets

IV_G201J
JOB - Start IRLM

IV_G203J
JOB - Start DBCTL Region - IVP3

IV_G204T
z/OS - Cold Start DBCTL

IV_G205T
z/OS - Review DBCTL Operator Commands

IV_G206J
JOB - FP BMP - DEDB Load

IV_G207J
JOB - FF BMP - Online Image Copy

IV_G208J
JOB - Concurrent Image Copy

IV_G209J
JOB - FF BMP - HIDAM Update

IV_G210J
JOB - FF BMP - HDAM Update

IV_G211J
JOB - FP BMP - DEDB Update

IV_G212J
JOB - FF JBP - HDAM Update

IV_G213T
TASK - MVS - Stop DBCTL with a /CHE FREEZE

IV_G214J
JOB - Start DBCTL Region - IVP3

IV_G215T
TASK - MVS - Warm Start DBCTL

IV_G216J
JOB - FF BMP - HIDAM Update

IV_G217J
JOB - FF BMP - HDAM Update

IV_G218J
JOB - FP BMP - DEDB Update

IV_G219T
TASK - MVS - Stop BMP Regions with a /STO REGION ABDUMP

IV_G220J
JOB - FF BMP - HIDAM Update

IV_G221J
JOB - FF BMP - HDAM Update

IV_G222J
JOB - FP BMP - DEDB Update

IV_G223T
TASK - MVS - Stop DBCTL with a MODIFY IMS,DUMP

IV_G224J
JOB - Log Recovery Utility - CLS / WADS

IV_G225J
JOB - Start DBCTL Region - IVP3

IV_G226T
TASK - MVS - Emergency Restart DBCTL

IV_G227J
JOB - FF BMP - HIDAM Update

IV_G228J
JOB - FF BMP - HDAM Update

IV_G229J
JOB - FP BMP - DEDB Update

IV_G230T
TASK - MVS - Stop DBCTL with /CHE FREEZE

IV_G231T
TASK - MVS - Stop IRLM

IV_G301J
JOB - List RECON data set

IV_G302J
JOB - Print an OLDS with DFDSS

IV_G303J
JOB - Print DC Monitor Reports

IV_G305J
JOB - Print Fast Path Log Analysis

IV_G306J
JOB - Log Recovery utility - PSB Mode

IV_G307J
JOB - File Select and Print utility

IV_G308J
JOB - Program Isolation (PI) Trace Report

IV_G309T
TASK - IPCS Dump Sample for DBCTL

IV_G401J
JOB - Scratch Data Sets

Steps Hx for IVP execution - DBT system (DB/DC)

The H series of steps include the jobs and tasks that you must perform during the execution of the DBT (DB/DC) sample system.

Name
Title

IV_H001T
NOTE - Introduction - IVP Execution - DBT

IV_H101J
JOB - Allocate Data Sets

IV_H102J
JOB - Initialize RECON

IV_H103J
JOB - Register Data Bases

IV_H104J
JOB - Data Base Initial Load

IV_H105J
JOB - Batch Image Copy

IV_H106T
z/OS - Clear z/OS DUMPxx Data Sets

IV_H201J
JOB - Start IRLM

IV_H203J
JOB - Start DB/DC Region IVP1

IV_H204T
TASK - IVP1 - Cold Start IMS

IV_H205T
TASK - IVP1 - Review MTO Operator Commands

IV_H206T
TASK - USER - Review User Operator Commands

IV_H207J
JOB - FP BMP - DEDB Load

IV_H208J
JOB - FF BMP - Online Image Copy

IV_H209J
JOB - Concurrent Image Copy

IV_H210J
JOB - FF BMP - HIDAM Update

IV_H211J
JOB - FF BMP - HDAM Update

IV_H212J
JOB - FP BMP - DEDB Update

IV_H213J
JOB - FF JBP - HDAM Update

IV_H214T
USER - FP MPP Transactions

IV_H215T
USER - FP IFP Transactions

IV_H216T
USER - FF JMP Transactions

IV_H225T
IVP1 - Stop IMS with a /CHE DUMPQ

IV_H226J
JOB - Start DB/DC Region - IVP1

IV_H227T
IVP1 - Warm Start IMS

IV_H228J
JOB - FF BMP - HIDAM Update

IV_H229J
JOB - FP BMP - HDAM Update

IV_H230J
JOB - FP BMP DEDB Update

IV_H231T
USER - FF MPP Transaction

IV_H232T
IVP1 - Stop Dependent Region /STO REGION ABDUMP

IV_H233J
JOB - FF BMP - HIDAM Update

IV_H234J
JOB - FF BMP - HDAM Update

IV_H235J
JOB - FF BMP - DEDB Update

IV_H236T
TASK - USER - FF MPP Transaction

IV_H237J
JOB - FP BMP - HIDAM Update

IV_H238J
JOB - FF BMP - HDAM Update

IV_H239J
JOB - FP BMP - DEDB Update

IV_H240T
TASK - USER - FF MPP Transaction

IV_H241T
z/OS - Stop IMS with a MODIFY IMS,DUMP

IV_H242J
JOB - Log Recovery Utility - CLS/WADS

IV_H243J
JOB - Start DB/DC Region - IVP1

IV_H244T
IVP1 - Emergency Restart IMS

IV_H245J
JOB - FF BMP - HIDAM Update

IV_H246J
JOB - FF BMP - HDAM Update

IV_H247J
JOB - FP BMP - DEDB Update

IV_H248T
USER - FF MPP Transaction

IV_H249T
IVP1 - Stop IMS with a /CHE FREEZE

IV_H250T
z/OS - Stop IRLM

IV_H301J
JOB - List RECON data set

IV_H302J
JOB - Print an OLDS with DFDSS

IV_H303J
JOB - Print DC Monitor Reports

IV_H305J
JOB - Print Log Statistics

IV_H306J
JOB - Print Log Transaction Analysis

IV_H307J
JOB - Print Fast Path Log Analysis

IV_H308J
JOB - Log Recovery utility - PSB Mode

IV_H309J
JOB - File Select and Print utility

IV_H310J
JOB - PI Trace Report utility

IV_H311T
TASK - IPCS Dump Sample for DB/DC and XRF

IV_H401J
JOB - Scratch Data Sets

Steps Ix for IVP execution - XRF system (DB/DC with XRF)

The I series of steps include the jobs and tasks that you must perform during the execution of the XRF (DB/DC with XRF) sample system.

Name
Title

IV_I001T
NOTE - Introduction - IVP Execution - XRF

IV_I101J
JOB - Allocate Data Sets

IV_I102J
JOB - Initialize RECON

IV_I103J
JOB - Register Data Bases

IV_I104J
JOB - Data Base Initial Load

IV_I105J
JOB - Batch Image Copy

IV_I106T
TASK - MVS - Clear MVS DUMPxx Data Sets

IV_I201J
JOB - Start IRLM

IV_I203J
JOB - Start DB/DC Region - IVP1

IV_I204T
TASK - IVP1 - Cold Start IVP1 as Active

IV_I205J
JOB - Start DB/DC Region - IVP2

IV_I206T
TASK - IVP2 - Emergency Restart IVP2 as Alternate

IV_I207J
JOB - FP BMP - DEDB Load

IV_I208J
JOB - FF JBP - HDAM Update

IV_I209T
USER - FF MPP Transactions

IV_I210T
USER - FP IFP Transactions

IV_I211T
USER - FF JMP Transactions

IV_I212T
IVP1 - /DIS HSB

IV_I213T
IVP2 - /DIS HSB

IV_I214T
z/OS - TKOVR IVP1 to IVP2 with a MODIFY IVP1,STOP

IV_I215T
IVP2 - /UNLOCK SYSTEM

IV_I216J
JOB - Start DB/DC Region - IVP1

IV_I217T
IVP1 - Emergency Restart IVP1 as Alternate

IV_I218T
TASK - USER - FF MPP Transactions

IV_I219T
TASK - USER - FP IFP Transactions

IV_I220T
TASK - USER - FF JMP Transactions

IV_I221T
TASK - IVP1 - TKOVER IVP2 to IVP1 with /SWI SYSTEM FORCE

IV_I222T
IVP1 - /UNLOCK SYSTEM

IV_I223J
JOB - Start DB/DC Region - IVP2

IV_I224T
IVP2 - Emergency Restart IVP2 as Alternate

IV_I225T
USER - FF MPP Transactions

IV_I226T
USER - FP IFP Transactions

IV_I227T
USER - FF JMP Transactions

IV_I228T
IVP2 - Stop IVP2 with a /STO BACKUP

IV_I229T
IVP1 - Stop IVP1 with a /CHE FREEZE

IV_I230T
z/OS - Stop IRLM

IV_I301J
JOB - List RECON data set

IV_I401J
JOB - Scratch Data Sets

Steps Jx for IVP execution - DCC system (DCCTL)

The J series of steps include the jobs and tasks that you must perform during the execution of the DCC (DCCTL) sample system.

Name
Title

IV_J001T
NOTE - Introduction - IVP Execution - DCC

IV_J101J
JOB - Allocate Data Sets

IV_J102J
JOB - Initialize RECON data set

IV_J103T
z/OS - Clear z/OS DUMPxx Data Sets

IV_J201J
JOB - Start DCCTL Region IVP4

IV_J202T
IVP4 - Cold Start IMS

IV_J203T
IVP4 - Review MTO Operator Commands

IV_J204T
USER - Review User Operator Commands

IV_J205J
JOB - Start the WFI BMP

IV_J206T
USER - BMP/MPP/IFP Transactions

IV_J208T
IVP4 - Stop IMS with a /CHE DUMPQ

IV_J209J
JOB - Start DCCTL Region IVP4

IV_J210T
IVP4 - Warm Start IMS

IV_J211J
JOB - Start the WFI BMP

IV_J212T
USER - BMP TADD Transaction

IV_J214T
IVP4 - Abend the WFI BMP - /STO REGION ABDUMP

IV_J215J
JOB - Restart (XRST) the WFI BMP

IV_J216T
USER - BMP TADD Transaction

IV_J218T
z/OS - Stop IMS with a MODIFY IMS,DUMP

IV_J219J
JOB - Log Recovery utility - CLS/WADS

IV_J220J
JOB - Start DCCTL Region IVP4

IV_J221T
IVP4 - Emergency Restart IMS

IV_J222J
JOB - Restart (XRST) the WFI BMP

IV_J223T
USER - WFI BMP TADD Transaction

IV_J224T
USER - BMP/MPP/IFP Transactions

IV_J225T
IVP4 - Stop IMS with a /CHE FREEZE

IV_J301J
JOB - List RECON data set

IV_J302J
JOB - Print an OLDS with DFDSS

IV_J303J
JOB - Print DC Monitor Reports

IV_J305J
JOB - Print Log Statistics

IV_J306J
JOB - Print Log Transaction Analysis

IV_J307J
JOB - Log Recovery utility - PSB Mode

IV_J308J
JOB - File Select and Print utility

IV_J309T
TASK - IPCS Dump Sample for DCCTL

IV_J401J
JOB - Scratch Data Sets

Steps Nx for execution - partition database sample application

The N series of steps include the jobs and tasks that you must perform during the execution of the IMS partition database sample application.

Name
Title

IV_N001T
NOTE - Introduction - Partition Data Base Sample

IV_N101J
JOB - Allocate Data Sets

IV_N102J
JOB - Initialize RECON data set / Register Data Bases

IV_N103J
JOB - Data Base Initial Load

IV_N202J
JOB - Unload Data Base - Migrate = YES

IV_N203J
JOB - Delete DBD

IV_N204J
JOB - DBDGEN

IV_N205J
JOB - ACBGEN

IV_N206T
TASK - HALDB Partition Definition Utility

IV_N207J
JOB - Allocate Partitioned Data Bases

IV_N208J
JOB - Initialize Partitioned Data Bases

IV_N209J
JOB - Reload Data Bases

IV_N210J
JOB - Batch Image Copy

IV_N211J
JOB - Copy Staging Library

IV_N301J
JOB - Start IRLM

IV_N303J
JOB - Start DB/DC Region - IVP1

IV_N304T
z/OS - Cold Start IMS - IVP1

IV_N305T
USER - Sample Transactions

IV_N306T
IVP1 - Stop IMS with a /CHE FREEZE

IV_N307T
z/OS - Stop IRLM

IV_N313J
JOB - Start DBCTL Region - IVP3

IV_N314T
z/OS - Cold Start DBCTL

IV_N315J
JOB - FF BMP - HIDAM Update

IV_N316T
z/OS - Stop DBCTL with a /CHE FREEZE

IV_N317T
z/OS- Stop IRLM

IV_N401J
JOB - Cleanup

IV_N402J
JOB - Scratch Data Sets

Steps Ox for Common Service Layer and Common Queue Server sample application

The O series of steps include the jobs and tasks that you must perform during the execution of the Common Service Layer sample application.

Name
Title

IV_O001T
NOTE - Introduction - CSL, DRD, OM Audit Trail, IMS Catalog

IV_O002T
NOTE - Introduction - CSL, CQS, DRD, OM Audit Trail, IMS Catalog

IV_O004T
NOTE - Introduction - CSL, CQS, DRD, OM Audit Trail, IMS Catalog

IV_O101J
JOB - Allocate Data Sets

IV_O102J
JOB - Allocate CQS Execution Data Sets

IV_O103J
JOB - Initialize RECON data set/Register Data Bases

IV_O104J
JOB - Data Base Initial Load

IV_O105J
JOB - Batch Image Copy

IV_O201J
JOB - Start SCI

IV_O202J
JOB - Start OM

IV_O203J
JOB - Start CQS

IV_O204J
JOB - Start RM

IV_O205T
TASK - SPOC Sample

IV_O210J
JOB - Start IRLM

IV_O214J
JOB - Start DCCTL Region IVP4

IV_O215J
JOB - Start DB/DC Region IVP1

IV_O216J
JOB - Start DBCTL Region - IVP3

IV_O217T
TASK - IVP1 - Cold Start IMS

IV_O218T
TASK - IVP3 - Cold Start DBCTL

IV_O219T
TASK - IVP4 - Cold Start DCCTL

IV_O220T
TASK - Type-2 commands sample

IV_O221T
TASK - Dynamic Resource Definition sample

IV_O222T
TASK - OM Audit Trail sample

IV_O231J
JOB - Start the WFI BMP

IV_O232J
JOB - FF BMP - HDAM Update

IV_O233J
JOB - FF BMP - HDAM Update

IV_O234T
USER - FF MPP Transactions

IV_0235T
USER - BMP Transaction

IV_0236T
IVP1 - Stop IMS with a /CHE FREEZE NOCQSSHUT

IV_0237T
z/OS - Stop DBCTL with a /CHE FREEZE

IV_0238T
IVP1 - Stop DCCTL with a /CHE FREEZE

IV_0239J
JOB - Start DB/DC Region - IVP1

IV_0240J
JOB - Start DBCTL Region - IVP3

IV_0241J
JOB - Start DCCTL Region - IVP4

IV_0242T
IVP1 - Cold Start IMS

IV_0243T
IVP3 - Cold Start DBCTL

IV_0244T
IVP4 - Cold Start DCCTL

IV_0245J
JOB - Start the WFI BMP

IV_0246J
JOB - FF BMP - HDAM Update

IV_0247J
JOB - FF BMP - HDAM Update

IV_0248T
USER - FF MPP Transactions

IV_0249T
USER - BMP Transaction

IV_0250T
IVP1 - Stop DCCTL with a /CHE FREEZE

IV_0251T
IVP1 - Stop IMS and CQS with a /CHE FREEZE

IV_0252T
z/OS - Stop DBCTL with a /CHE FREEZE

IV_0253T
z/OS - Shut Down SCI/OM/RM

IV_0254T
TASK - Introduction to the IMS Catalog

IV_0255J
JOB - Start DB/DC region with XRF

IV_0256J
JOB - Start DB/DC region

IV_0257J
JOB - Start DBCTL region

IV_0258T
TASK - Cold start IMS

IV_0259T
TASK - Cold Start DBCTL

IV_O260J
JOB - Retrieve data from the IMS catalog database

IV_O261J
JOB - Run Java sample (SQL with type-2 connectivity)

IV_O262J
JOB - Run Java sample (DL/I with type-2 connectivity)

IV_O263T
TASK - Stop IMS with /CHE FREEZE

IV_O264T
TASK - Start TCP/IP and RRS

IV_O265J
JOB - Start SCI

IV_O266J
JOB - Start OM

IV_O267J
JOB - Start RM

IV_O268J
JOB - Start DB/DC with XRF

IV_O269J
JOB - Start DB/DC

IV_O270J
JOB - Start DBCTL

IV_O271T
TASK - Cold start IMS DB/DC

IV_O272T
TASK - Cold start IMS DBCTL

IV_O273J
JOB - Start ODBM

IV_O274J
JOB - Start IMS Connect

IV_O275J
JOB - Run Java sample (SQL with type-4 connectivity)

IV_O276J
JOB - Run Java sample (DL/I with type-4 connectivity)

IV_O277T
TASK - Shut down CSL components

IV_O278T
TASK - Stop TCP/IP and RRS

IV_O279T
TASK - Stop IMS Connect region

IV_O280T
TASK - Stop IMS with /CHE FREEZE

IV_O282T
TASK - Stop IRLM

IV_O401J
JOB - Scratch Data Sets

Steps Px for type-2 command environment sample application

The P series of steps include the jobs and tasks that you must perform during the execution of the enhanced command environment sample application.

IV_P001T
NOTE - Introduction - Type-2 Command Environment Sample

IV_P101J
JOB - Allocate Data Sets

IV_P102J
JOB - Initialize RECON data set/Register Databases

IV_P103J
JOB - Database Initial Load

IV_P104J
JOB - Batch Image Copy

IV_P210J
JOB - Start IRLM

IV_P214J
JOB - Start DCCTL Region IVP4

IV_P215J
JOB - Start DB/DC Region IVP1

IV_P216J
JOB - Start DBCTL Region IVP3

IV_P217T
TASK - SPOC Sample I

IV_P218T
IVP1 - Cold Start IMS

IV_P219T
IVP3 - Cold Start DBCTL

IV_P220T
IVP4 - Cold Start DCCTL

IV_P221T
TASK - SPOC Sample II

IV_P229T
IVP4 - Stop DCCTL with a `/CHE FREEZE`

IV_P230T
IVP1 - Stop IMS with a `/CHE FREEZE`

IV_P231T
IVP3 - Stop DBCTL with a `/CHE FREEZE`

IV_P232T
z/OS - Shut Down SCI/OM

IV_P233T
z/OS - Stop IRLM

IV_P401J
JOB - Scratch data sets

Steps Qx for execution - full-function MPP transaction using the IMS Connect sample application

The Q series of steps include the jobs and tasks that you must perform during the execution of full-function MPP transaction using the IMS Connect sample application.

Name
Title

IV_Q001T
NOTE - Introduction to the IMS Connect Sample

IV_Q101J
JOB - Allocate Data sets

IV_Q102J
JOB - Initialize RECON

IV_Q103J
JOB - Register Data Bases

IV_Q104J
JOB - Data Base Initial Load

IV_Q105J
JOB - Batch Image Copy

IV_Q201J
JOB - Start IRLM

IV_Q202JT
TASK - Start TCPIP

IV_Q203J
JOB - Start DB/DC Region - IVP1

IV_Q204T
TASK - Cold Start IMS - IVP1

IV_Q205J
JOB - Start IMS Connect Region

IV_Q206J
JOB - IMS Connect client application

IV_Q207T
TASK - IMS Connect command

IV_Q210T
TASK - IVP1 - Stop IMS Connect Region

IV_Q211T
TASK - Stop IMS with a /CHE FREEZE

IV_Q212T
z/OS - Stop IRLM

IV_Q401J
JOB - Scratch Data Sets

Steps Rx for the parallel RECON access sample

The R series of steps include the jobs and tasks that you must perform during the execution of the parallel RECON access samples.

Name
Title

IV_R001T
NOTE - Intro - Parallel RECON Access sample

IV_R002T
TASK - z/OS Resource Recovery Services (RRS)

IV_R003T
TASK - Start Transactional VSAM (TVS)

IV_R004J
JOB - Start SCI

IV_R005J
JOB - Assemble/Bind DBRC SCI Registration exit

IV_R101J
JOB - Allocate Data Sets

IV_R102J
JOB - Initialize RECON

IV_R103J
JOB - Register Data Bases

IV_R104J
JOB - Data Base Initial Load

IV_R105J
JOB - Batch Image Copy

IV_R201J
JOB - Start IRLM

IV_R203J
JOB - Start DB/DC Region - IVP1

IV_R204T
IVP1 - Cold Start IMS

IV_R220J
JOB - FF BMP - HIDAM Update

IV_R221T
USER - FF MPP Transactions

IV_R222J
USER - JOB - Start the WFI BMP

IV_R223T
USER - BMP/MPP/IFP Transactions

IV_R224J
JOB - List RECON

IV_R230T
TASK - Stop DB/DC

IV_R231T
TASK - Stop DBCTL

IV_R232T
TASK - Stop DCCTL

IV_R233T
z/OS - Stop IRLM

IV_R234T
z/OS - Stop SCI

IV_R235T
z/OS - Stop TVS

IV_R236T
z/OS - Stop RRS

IV_R401J
JOB - Scratch Data Sets

Steps Sx for callout samples

The S series of steps include the jobs and tasks that you must perform to set up the environment for running the synchronous and asynchronous callout IVP samples that use the IMS TM resource adapter, IMS Enterprise Suite SOAP Gateway, or a user-supplied IMS Connect client application.

Name
Title

IV_S001T
NOTE - Introduction - Callout sample

IV_S101J
JOB - Allocate Data Sets

IV_S102J
JOB - Initialize RECON

IV_S103J
JOB - Register Databases

IV_S104J
JOB - Database Initial Load

IV_S105J
JOB - Batch Image Copy

IV_S201J
JOB - Start IRLM

IV_S202T
TASK - Start TCP/IP

IV_S203J
JOB - Start DB/DC Region

IV_S205T
TASK - Cold Start IMS - IVP1

IV_S207J
JOB - Start IMS Connect Region

IV_S220J
JOB - Host program sends an Async Callout message

IV_S221T
TASK - Display transaction pipe status

IV_S222J
JOB - Client program receives Async Callout request

IV_S223J
JOB - IMS Program Sends a Sync Callout Message

IV_S224T
TASK - Display TPIPE Status for Sync

IV_S225J
JOB - User App Receives Sync Request; Sends Response

IV_S226T
TASK - NOTE - Callout Sample - IMS TM Resource Adapter

IV_S227J
JOB - IMS Program Sends an Async Callout Message

IV_S228J
JOB - IMS Program Sends a Sync Callout Message

IV_S229T
TASK - Callout Sample - IMS Enterprise Suite SOAP Gateway

IV_S230J
JOB - IMS Program Sends an Async Callout Message

IV_S231J
JOB - IMS Program Sends a Sync Callout Message

IV_S301T
TASK - Stop IMS Connect region

IV_S302T
TASK - Stop IMS with /CHE FREEZE

IV_S304T
TASK - Stop IRLM

IV_S401J
JOB - Scratch Data Sets

Steps Tx for Open Database sample application

The T series of steps include the jobs and tasks that you must perform during the execution of the Open Database sample application.

Name
Title

IV_T101T
NOTE - Introduction - Open Database sample

IV_T101J
JOB - Allocate Data Sets

IV_T102J
JOB - Initialize RECON

IV_T103J
JOB - Register Data Bases

IV_T104J
JOB - Data Base Initial Load

IV_T105J
JOB - Batch Image Copy

IV_T201T
TASK - Start TCP/IP and z/OS Resource Recovery Services

IV_T201J
JOB - Start SCI

IV_T202J
JOB - Start OM

IV_T203J
JOB - Start RM

IV_T204J
JOB - Start IRLM

IV_T205J
JOB - Start DB/DC with XRF

IV_T206J
JOB - Start DB/DC

IV_T207J
JOB - Start DBCTL

IV_T208T
TASK - Cold Start IMS DB/DC

IV_T209T
TASK - Cold Start IMS DBCTL

IV_T210J
JOB - Start ODBM

IV_T211J
JOB - Start IMS Connect

IV_T220J
JOB - Create a UNIX Script to run the application

IV_T230J
JOB - Run the sample and copy the output to job log

IV_T301T
TASK - Stop OM, RM, SCI, ODBM, IMS Connect

IV_T302T
TASK - Stop IMS with /CHE FREEZE

IV_T303T
TASK - Stop IRLM

IV_T401J
JOB - Scratch Data Sets

Steps Ux for the IMSRSC repository sample application

The U series of steps include the jobs and tasks that you must perform during the execution of the IMSRSC repository sample application.

Name
Title

IV_U101T
NOTE - Introduction to IMSRSC repository Usage For DRD resources

IV_U101J
JOB - Allocate data sets

IV_U102J
JOB - Start SCI

IV_U103J
JOB - Start OM

IV_U104J
JOB - Start the Repository Server (RS)

IV_U105J
JOB - Add an IMSRSC repository to the RS catalog repository and then start the repository

IV_U106J
JOB - Start RM

IV_U202J
JOB - List status information for all repositories

IV_U204J
JOB - Populate the IMSRSC repository

IV_U205J
JOB - Stop and rename the IMSRSC repository in the RS catalog repository

IV_U206J
JOB - List detail information for a single repository

IV_U207J
JOB - Modify / update definitions for a repository

IV_U208J
JOB - Delete a repository in the RS catalog repository

IV_U209J
JOB - Request the RS to start an already deleted repository

IV_U401T
z/OS - Shut down SCI, OM, RM, and RS

IV_U402J
z/OS - Scratch data sets

Related concepts
“Dynamic resource definition sample application with the IMSRSC repository” on page 29
The dynamic resource definition (DRD) sample application demonstrates how to set up and operate DRD with the IMSRSC repository.

Steps Zx for index of additional PDS members

The Z series of steps include the jobs and tasks for index of additional PDS members.

Name
Title

IV_Z001T
NOTE - Introduction - INDEX to PDS Members

DFSAAAS0
Copyright Statement - //*

DFSAAAS1
Copyright Statement - *

DFSAAAS2
Copyright Statement - /* */

DFSIXS00
IMBED - Copyright Statement - //*

DFSIXS01
IMBED - Copyright Statement - *

DFSIXS02
IMBED - Copyright Statement - /* */

DFSIXS03
IMBED - EXAMPLE Block

DFSIXS04
IMBED - IMBED - JMP #1 Execution JCL for IVP1

DFSIXS05
IMBED - Standard JOB Statement

DFSIXS06
IMBED - SMP/E Cataloged Procedure

DFSIXS07
IMBED - IRLM DXRJPROC In-line Procedure

DFSIXS08
IMBED - JMP #1 Execution JCL for IVP2

DFSIXS09
IMBED - DFSPBIV3 for DBCTL

DFSIXS10
IMBED - DFSPBIV1 for DB/DC

DFSIXS11
IMBED - DFSPBIV2 for XRF - System IVP2

DFSIXS12
IMBED - DFSPBIV1 for XRF - System IVP1

DFSIXS13
IMBED - DFSPBIV4 for DCCTL - System IVP4

DFSIXS14
IMBED - IRLM #1 Execution JCL

DFSIXS15
IMBED - IRLM #2 Execution JCL

DFSIXS16
IMBED - DBCTL Execution Step for IVP3

DFSIXS17
IMBED - DB/DC Execution Step for IVP1

DFSIXS18
IMBED - DB/DC Execution Step for IVP2

DFSIXS19
IMBED - MPP #1 Execution JCL for IVP1

DFSIXS20
IMBED - IFP #1 Execution JCL for IVP1

DFSIXS21
IMBED - IFP #2 Execution JCL for IVP1

DFSIXS22
IMBED - IFP #3 FP Sample Application for IVP1

DFSIXS23
IMBED - MPP #1 Execution JCL for IVP2

DFSIXS24
IMBED - IFP #1 Execution JCL for IVP2

DFSIXS25
IMBED - IFP #2 Execution JCL for IVP2

DFSIXS26
IMBED - HIDAM BMP Execution Step for IVP3

DFSIXS27
IMBED - HDAM BMP Execution Step for IVP3

DFSIXS28
IMBED - DEDB BMP LOAD Execution Step for IVP3

DFSIXS29
IMBED - DEDB BMP Execution Step for IVP3

DFSIXS30
IMBED - DBRC in-line procedure

DFSIXS32
IMBED - DBRC Skeletal JCL - JOBJCL2

DFSIXS33
IMBED - DBRC Skeletal JCL - ARCHJCL

DFSIXS34
IMBED - DBRC Skeletal JCL - CAJCL

DFSIXS35
IMBED - DBRC Skeletal JCL - ICJCL

DFSIXS36
IMBED - DBRC Skeletal JCL - JOBJCL

DFSIXS37
IMBED - DBRC Skeletal JCL - LOGCLJCL

DFSIXS38
IMBED - DBRC Skeletal JCL - OICJCL

DFSIXS39
IMBED - DBRC Skeletal JCL - RECOVJCL

DFSIXS40
IMBED - Stage 1 Source - IVP Sample Application

DFSIXS41
IMBED - Stage 1 Source - IMS Sample Application

DFSIXS42
IMBED - Stage 1 Source - FP Sample Application

DFSIXS43
IMBED - Stage 1 Source - Communications

DFSIXS46
IMBED - JES Control Statement for User PROCLIB

DFSIXS47
IMBED - User Supplied JES Control Statements

DFSIXS48
IMBED - IVPB HIDAM Batch execution step

DFSIXS49
IMBED - IVPB HDAM Batch execution step

DFSIXS50
IMBED - IVP1 HIDAM BMP execution step

DFSIXS51
IMBED - IVP1 HDAM BMP execution step

DFSIXS52
IMBED - IVP1 DEDB BMP LOAD execution step

DFSIXS5A
IMBED - IVP1 HDAM JBP execution step

DFSIXS53
IMBED - IMSWT000 - IVP1/IVP4

DFSIXS54
IMBED - IMSWT000 - IVP2

DFSIXS56
IMBED - Statistical Analysis utility In-line Proc

DFSIXS57
IMBED - Log Transact Analysis utility In-line Proc

DFSIXS58
IMBED - FP Log Analysis utility In-line Proc

DFSIXS59
IMBED - DEDB BMP Execution Step for IVP1

DFSIXS60
IMBED - DFSIVD1 - HIDAM/OSAM - DB Load JOB Step

DFSIXS61
IMBED - DFSIVD2 - HDAM/VSAM - DB Load JOB Step

DFSIXS62
IMBED - DFSIVD3 - DEDB/VSAM - DB Load JOB Step

DFSIXS63
IMBED - DI21PART - HISAM/VSAM - DB Load JOB Step

DFSIXS64
IMBED - DBFSAMD3 - DEDB/VSAM - DB Load JOB Step

DFSIXS65
IMBED - DBFSAMD4 - HDAM/VSAM - DB Load JOB Step

DFSIXS66
IMBED - MSDBs - DB Load JOB Step - IVP & FP Sample

DFSIXS67
IMBED - INIT.RECON data set Control Statement

DFSIXS68
IMBED - INIT.DB/DBDS/ADS - DFSIVD1 - HIDAM/OSAM

DFSIXS69
IMBED - INIT.DB/DBDS/ADS - DFSIVD2 - HDAM/VSAM

DFSIXS70
IMBED - INIT.DB/DBDS/ADS - DFSIVD3 - DEDB/VSAM

DFSIXS71
IMBED - INIT.DB/DBDS/ADS - DI21PART - HISAM/VSAM

DFSIXS72
IMBED - INIT.DB/DBDS/ADS - DBFSAMD3 - DEDB/VSAM

DFSIXS73
IMBED - INIT.DB/DBDS/ADS - DBFSAMD4 - HDAM/VSAM

DFSIXS74
IMBED - Allocation JCL - DFSIVD1 - HIDAM/OSAM

DFSIXS75
IMBED - Allocation JCL - DFSIVD2 - HDAM/VSAM

DFSIXS76
IMBED - Allocation JCL - DFSIVD3 - DEDB/VSAM

DFSIXS77
IMBED - Allocation JCL - DI21PART - HISAM/VSAM

DFSIXS78
IMBED - Allocation JCL - DBFSAMD3 - DEDB/VSAM

DFSIXS79
IMBED - Allocation JCL - DBFSAMD4 - HDAM/VSAM

DFSIXS80
IMBED - Allocation JCL - MSDBINIT - IVP & FP SMPL

DFSIXS81
IMBED - Allocation JCL - DBRC RECON data set Data Sets

DFSIXS82
IMBED - Scratch JCL - DFSIVD1 - HIDAM/OSAM

DFSIXS83
IMBED - Scratch JCL - DFSIVD2 - HDAM/VSAM

DFSIXS84
IMBED - Scratch JCL - DFSIVD3 - DEDB/VSAM

DFSIXS85
IMBED - Scratch JCL - DI21PART - HISAM/VSAM

DFSIXS86
IMBED - Scratch JCL - DBFSAMD3 - DEDB/VSAM

DFSIXS87
IMBED - Scratch JCL - DBFSAMD4 - HDAM/VSAM

DFSIXS88
IMBED - Scratch JCL - MSDBINIT - IVP & FP SMPL

DFSIXS89
IMBED - Scratch JCL - DBRC RECON data set Data Sets

DFSIXS90
IMBED - DCCTL execution step for system IVP4

DFSIXS91
IMBED - Execution step for BMP - IVP4

DFSIXS92
IMBED - Execution JCL for MPP #1 - IVP4

DFSIXS93
IMBED - Execution JCL for IFP #1 - IVP4

DFSIXS94
IMBED - XRST Execution JCL for WFI BMP - IVP4

DFSIXS95
IMBED - PI Trace Report utility - In-line Proc

DFSIXS96
IMBED - Scratch JCL - CQS Execution data sets

DFSIXS97
IMBED - Scratch JCL - Java data sets

DFSIXS98
IMBED - Allocation JCL - Java data sets

DFSIXS99
IMBED - Java AUTO and EMPL - DB Load JOB Step

DFSIVC04
IVP - CPY - HD DB Load control statements

DFSIVC05
IVP - CPY - HD DB DLI/DBB/BMP control statements

DFSIVC06
IVP - CPY - MSDB Load control statements

DFSIVC07
IVP - CPY - WFI BMP (DCCTL) load statements

DFSIVD1
IVP - DBD - HIDAM/OSAM

DFSIVD1I
IVP - DBD - HIDAM Index/VSAM

DFSIVD2
IVP - DBD - HDAM/VSAM

DFSIVD3
IVP - DBD - DEDB/VSAM

DFSIVD4
IVP - DBD - MSDB

DFSIVD5
IVP - DBD - GSAM/BSAM

DFSIVP1
IVP - PSB - Non-conv HIDAM

DFSIVP2
IVP - PSB - Non-conv HDAM

DFSIVP3
IVP - PSB - Conv HDAM

DFSIVP31
IVP - PSB - Conv HDAM - PASCAL Version

DFSIVP32
IVP - PSB - Conv HDAM - C Version

DFSIVP33
IVP - PSB - Conv HDAM - Java Version

DFSIVP34
IVP - PSB - Conv HDAM - COBOL Version

DFSIVP35
IVP - PSB - Conv HDAM - REXX Version

DFSIVP37
IVP - PSB - Conv HDAM - Java Version

DFSIVP4
IVP - PSB - IFP DEDB

DFSIVP5
IVP - PSB - IFP MSDB

DFSIVP6
IVP - PSB - DLI/DBB/BMP HIDAM

DFSIVP61
IVP - PSB - DLI/DBB/BMP HIDAM - PASCAL Version

DFSIVP62
IVP - PSB - DLI/DBB/BMP HIDAM - C Version

DFSIVP64
IVP - PSB - DLI/DBB/BMP HIDAM - COBOL Version

DFSIVP65
IVP - PSB - DLI/DBB/BMP HIDAM - REXX Version

DFSIVP67
IVP - PSB - DLI/DBB/JBP HDAM - Java Version

DFSIVP7
IVP - PSB - DLI/DBB/BMP HDAM

DFSIVP8
IVP - PSB - BMP DEDB

DFSIVP9
IVP - PSB - OLIC BMP - HIDAM/OSAM

DFSIVPA
IVP - PSB - HIDAM Load

DFSIVPB
IVP - PSB - HDAM Load

DFSIVPC
IVP - PSB - DEDB Load

DFSIVPD
IVP - PSB - DCCTL WFI BMP

DFSIVPE
IVP - PSB - DCCTL Non-Conversational MPP

DFSIVPF
IVP - PSB - DCCTL Conversational MPP

DFSIVPG
IVP - PSB - DCCTL IFP

DFSIVF1
IVP - MFS - Non-conv HIDAM

DFSIVF2
IVP - MFS - Non-conv HDAM

DFSIVF3
IVP - MFS - Conv HDAM

DFSIVF31
IVP - MFS - Conv HDAM - PASCAL Version

DFSIVF32
IVP - MFS - Conv HDAM - C Version

DFSIVF33
IVP - MFS - Conv HDAM - Java Version

DFSIVF34
IVP - MFS - Conv HDAM - COBOL Version

DFSIVF35
IVP - MFS - Conv HDAM - REXX Version

DFSIVF37
IVP - MFS - Conv HDAM - Java Version

DFSIVF4
IVP - MFS - IFP DEDB

DFSIVF5
IVP - MFS - IFP MSDB

DFSIVFD
IVP - MFS - WFI BMP

DFSIVFE
IVP - MFS - Non-Conversational MPP

DFSIVFF
IVP - MFS - Conversational MPP

DFSIVFG
IVP - MFS - IFP

DFSIVA1
IVP - PGM - Non-conv HIDAM

DFSIVA2
IVP - PGM - Non-conv HDAM

DFSIVA3
IVP - PGM - Conv HDAM

DFSIVA31
IVP - PGM - Conv HDAM - PASCAL Version

DFSIVA32
IVP - PGM - Conv HDAM - C Version

DFSIVA34
IVP - PGM - Conv HDAM - COBOL Version

DFSIVA35
IVP - PGM - Conv HDAM - REXX Version

DFSIVA4
IVP - PGM - IFP DEDB

DFSIVA5
IVP - PGM - IFP MSDB

DFSIVA6
IVP - PGM - DLI/DBB/BMP HIDAM

DFSIVA61
IVP - PGM - DLI/DBB/BMP HIDAM - PASCAL Version

DFSIVA62
IVP - PGM - DLI/DBB/BMP HIDAM - C Version

DFSIVA64
IVP - PGM - DLI/DBB/BMP HIDAM - COBOL Version

DFSIVA65
IVP - PGM - DLI/DBB/BMP HIDAM - REXX Version

DFSIVA7
IVP - PGM - DLI/DBB/BMP HDAM

DFSIVA8
IVP - PGM - BMP DEDB

DFSIVA9
IVP - PGM - IMS Connect client

DFSIVAC
IVP - PGM - DEDB Load

DFSIVAD
IVP - PGM - WFI BMP for DCCTL

DFSIVAE
IVP - PGM - Non-Conversational MPP for DCCTL

DFSIVAF
IVP - PGM - Conversational MPP for DCCTL

DFSIVAG
IVP - PGM - IFP for DCCTL

IV_REXX
IVP - PGM - IVPREXX Generic EXEC Driver

DFSIVG20
IVP - PGM - WTOR Subroutine for PASCAL

DFSIVG30
IVP - PGM - WTOR Subroutine for C

DFSIVJG2
IVP - JCL - Sample Assemble and Link for DFSIVG20

DFSIVJG3
IVP - JCL - Sample Assemble and Link for DFSIVG30

DFSIVJP3
IVP - JCL - Sample Compile and Link for DFSIVA31

DFSIVJP6
IVP - JCL - Sample Compile and Link for DFSIVA61

DFSIVJC3
IVP - JCL - Sample Compile and Link for DFSIVA32

DFSIVJC6
IVP - JCL - Sample Compile and Link for DFSIVA62

DFSIVJB3
IVP - JCL - Sample Compile and Link for DFSIVA34

DFSIVJB6
IVP - JCL - Sample Compile and Link for DFSIVA64

DI21PART
IMS - DBD - HISAM/VSAM

DFSSAMC1
IMS - CPY - DI21PART Dump Control Statements

DFSSAM11
IMS - PSB - DB Load

DFSSAM12
IMS - PSB - PART Tran

DFSSAM13
IMS - PSB - DSPINV Tran

DFSSAM14
IMS - PSB - ADDPART/ADDINV/DLETPART/DLETINV Tran

DFSSAM15
IMS - PSB - CLOSE Tran

DFSSAM16
IMS - PSB - DISBURSE Tran

DFSSAM17
IMS - PSB - DSPALLI Tran

DFSSAM18
IMS - PSB - DB Dump

DFSSAM19
IMS - PSB - Batch/BMP Misc

DFSSAM01
IMS - PGM - DB Load

DFSSAM02
IMS - PGM - PART Tran

DFSSAM03
IMS - PGM - DSPINV Tran

DFSSAM04
IMS - PGM - ADDPART/ADDINV/DLETPART/DLETINV Tran

DFSSAM05
IMS - PGM - CLOSE Tran

DFSSAM06
IMS - PGM - DISBURSE Tran

DFSSAM07
IMS - PGM - DSPALLI Tran

DFSSAM08
IMS - PGM - DB Dump

DFSSUT04
IMS - PGM - Unexpected Status exit

MFDFSYSN
IMS - CPY - DB Load input

DFSIVPC1
IMS - PSB - IBM CICS Transaction Server for z/OS IVP DFHSAM04

DFSIVPC2
IMS - PSB - CICS IVP DFHSAM05

DFSIVPC3
IMS - PSB - CICS IVP DFHSAM14

DFSIVPC4
IMS - PSB - CICS IVP DFHSAM24

DFSIVPC5
IMS - PSB - CICS IVP DFHSAM15

DFSIVPC6
IMS - PSB - CICS IVP DFHSAM25

DBFSAMD1
FP - DBD - MSDB

DBFSAMD2
FP - DBD - MSDB

DBFSAMD3
FP - DBD - DEDB

DBFSAMD4
FP - DBD - HDAM/VSAM

DBFSAMP1
FP - PSB - DEDB Load

DBFSAMP2
FP - PSB - HDAM Load

DBFSAMP3
FP - PSB - FPSAMP1

DBFSAMP4
FP - PSB - FPSAMP2

DBFSAMP5
FP - PSB - HDAM MISC

DBFSAMP6
FP - PSB - DEDB MISC

DBFSAMF1
FP - MFS - FPSAMP1/FPSAMP2

DBFSAMA1
FP - PGM - DEDB Load

DBFSAMA2
FP - PGM - HDAM Load

DBFSAMA3
FP - PGM - FPSAMP1/FPSAMP2

DFSIVJ01
JOB - Dialog init - Define ICF User Cat. / ALIASs

DFSIVJ02
JOB - Dialog init - Alloc INSTALIB / Copy Tape

DFSIVJ03
JOB - Dialog init - Alloc SYSLIBS / Copy from tape

ARCHJCL
SKEL - ARCHJCL

CAJCL
SKEL - CAJCL

ICJCL
SKEL - ICJCL

JOBJCL
SKEL - JOBJCL

LOGCLJCL
SKEL - LOGCLJCL

OICJCL
SKEL - OICJCL

RECOVJCL
SKEL - RECOVJCL

IVP system definition stage 1 input streams

Use the sample source deck or input source for the IVP as an educational tool. This information is important during problem determination and troubleshooting.

The IMS system definition stage 1 input streams in this section are generated by the IVP dialog. The samples are not members of SDFSISRC.

DBB - DB Batch (batch) stage 1

This topic contains the stage 1 source for a DB batch system.

DBC - Database Control (DBCTL) stage 1

This topic contains the stage 1 source for a DBCTL system.

DBT - Database/Transaction Manager (DB/DC) stage 1

This topic contains the stage 1 source for a DBT (DB/DC) system.

XRF - Database/Transaction Manager with Extended Recovery Facility (DB/DC with XRF) stage 1

This topic contains the stage 1 source for a DB/DC system with XRF.

DCC - Transaction Manager Control (DCCTL) stage 1

This topic contains the stage 1 source for a DCCTL system.

IVP environment options

During the initialization phase of the IVP process, you select the environment options specific to the needs of your business.

Each environment option uses different IMS components and facilities. The following table shows the IVP environment option followed by the IMS components and facilities that are specific to that environment.

IVP environment option	IMS components and facilities
DBB (DB)	GSAM DBs (HISAM, HIDAM, and HDAM) Logging Database Recovery Control (DBRC) Internal Resource Lock Manager (IRLM) (optional) Batch applications
DBC (DBCTL)	GSAM DBs (HISAM, HIDAM, HDAM, PHIDAM, DEDB) Database Recovery Control (DBRC) Internal Resource Lock Manager (IRLM) (optional) Resource Access Control Facility (RACF) (optional) Batch applications Batch-oriented BMP applications HALDB sample Common Service Layer sample Parallel RECON Access (optional) Open Database (optional) Dynamic resource definition (DRD)

IVP environment option	IMS components and facilities
DBT (DB/DC)	GSAM DBs (HISAM, HIDAM, HDAM, PHIDAM, DEDB, MSDB) Logging Database Recovery Control (DBRC) Java sample applications Logging IMS Connect sample application Internal Resource Lock Manager (IRLM) (optional) IMS Transaction Manager (IMS TM) Time Control Option (TCO) file Batch applications Batch-oriented BMP applications Non-conversational message processing program (MPP) applications Conversational MPP applications IFP applications (optional) HALDB sample Common Service Layer sample Resource Access Control Facility (RACF) (optional) Parallel RECON Access (optional) Open Database (optional) Dynamic resource definition (DRD)
XRF (DB/DC with XRF)	GSAM DBs (HISAM, HIDAM, HDAM, PHIDAM, DEDB, MSDB) Logging Database Recovery Control (DBRC) Java sample applications IMS Connect sample application Internal Resource Lock Manager (IRLM) (optional) IMS Transaction Manager (IMS TM) Time Control Option (TCO) file Batch applications Batch-oriented BMP applications Non-conversational MPP applications Conversational MPP applications IFP applications (optional) HALDB sample Common Service Layer sample Resource Access Control Facility (RACF) (optional) Parallel RECON Access (optional) Open Database (optional) Dynamic resource definition (DRD)

IVP environment option	IMS components and facilities
DCC (DCCTL)	GSAM Logging Database Recovery Control (DBRC) IMS Transaction Manager (IMS TM) Time Control Option (TCO) file Transaction-driven WFI BMP applications Non-conversational MPP applications Conversational MPP applications IFP applications (optional) Common Service Layer sample Resource Access Control Facility (RACF) (optional) Parallel RECON Access (optional) Dynamic resource definition (DRD)

Related tasks

"Selecting the environment options" on page 12
Select the options that apply to your environment. The IVP provides suboptions and tasks based on your choices to build a sample IMS system for installation verification.

Variable gathering dialog options

Use the action commands and modes available for the variable-gathering phase.

Variable-gathering action commands

Two modes are used to display the variables:

LST
Variables are presented as a scrollable list of items. One or more items can be modified at a time, but minimal information is displayed for each item. LST is the default.

ENT
Variables are presented one at a time. Scrollable descriptive information is provided for each variable.

Action commands are provided to support the IVP dialog during the variable-gathering phase. Action commands are also referred to as action verbs.

The following table contains the action commands, accepted modes, and command descriptions. Mode indicates whether the commands are accepted in:

LST
LST mode

ENT
ENT mode

Both
LST mode and ENT mode

In the following table, the capitalized letters in the action column indicate the shortest allowable abbreviation for each command.

Table 3. Variable-gathering action commands

Action	Mode	Description
Chg	Both	Changes information in an item. The dialog performs basic validity checking for the new value. If an item is modified, CHG is the default. It is not necessary to use CHG in the action field.

Table 3. Variable-gathering action commands (continued)

Action	Mode	Description
Doc	Both	Prints variables documentation to the ISPF LIST data set. The DOC action prints all variables, not only the variable where the DOC action is requested.
e**N**t	LST	Switches to ENT mode. ENT mode presents items one at a time on a formatted screen and provides a description of the variable along with its usage.
Lst	ENT	Switches to LST mode. LST mode presents a scrollable list of items. Within ENT Mode, the ISPF END command is also interpreted as the LST action.
Nxt	ENT	Moves forward to the next item.
Prv	ENT	Moves backward to the previous item.
Rfr	Both	Refreshes a variable value from the IVP master table.
Imp	Both	Imports the IVP variables.
Exp	Both	Exports the IVP variables.

In LST mode, you can either change one item at a time or make changes to many items before pressing **ENTER**. Whenever two or more changes are made before pressing **Enter**, the dialog attempts to process all change requests before returning control.

If errors occur during the variable-gathering phase, the item is updated with the supplied information and the character string "ERR" is placed into the action field. If errors exist after all requests have been processed, a single error message is displayed and the screen is positioned at the first item containing the string "ERR" in the action field. Correct all errors before the dialog allows you to enter the file-tailoring or execution phases. If you cannot tell what is wrong with a given item, type **CHG** in the action field for that item, and change one item at a time. When you press **Enter**, the dialog reruns variable edit for that item and produces an appropriate error message.

Variable gathering—LST mode

When you enter the variable-gathering phase, you are in LST mode. The following figure depicts the LST mode panel of the variable gathering phase. This mode provides the greatest visibility of the variables available for the selected option.

```
 Help
--------------------------------------------------------------------------------
IVP          Variable Gathering (LST Mode)- XRF        IMS 15.1 ROW 1 to 8 of 154
COMMAND ===>                                             SCROLL ===> PAGE

Action Codes: Chg Doc eNt Rfr Imp Exp - CHG is the default for a modified
              item
      Variable = Value.....................................
          Var-Title.....................................
 *       IXUIVPHQ = IVPIVP11
               IVP - High-level DSNAME qualifier for IVP (IVP) data sets
 *       IXURLMHQ = IVPRLM11
               IVP - High-level DSNAME qualifier for the IRLM (RLM) data sets
 *       IXUDLBHQ = IVPIVP11
               IVP - High-level DSNAME qualifier for IMS DLIB (DLB) data sets
 *       IXUSYSHQ = IVPIVP11
               IVP - High-level DSNAME qualifier for IMS System (SYS) data sets
 *       IXUEXEHQ = IVPIVP11
               IVP - High-level DSNAME qualifier for Execution (EXE) data sets
 *       IXUUTLHQ = IVPIVP11
               IVP - High-level DSNAME qualifier for Utility (UTL) data sets
 !       IXUSSCLS =
               SMS - Storage Class
 !       IXUSMCLS =
               SMS - Management Class
 !       IXUTAPEU = 3480
               IVP - Tape device type
```

Figure 42. Variable gathering (LST mode) panel

If you are not sure what a variable is, you can use the ENT action to switch to ENT mode, read the variable description, and use the LST action to return to LST Mode. You can also use the ENT action to switch to ENT mode, and then progress through the variables one at a time.

Special characters in the action field indicate changes to variables:

!
: Indicates that either a variable was added to the table (due to service) or that the RFR action has restored the master table default value. You can remove the ! indicator by running the table-merge process.

*****
: Indicates that the variable was changed, either by the CHG action or by the copy-startup variables process.

@
: Indicates that the variable has been changed, either by the CHG action or by the copy-startup variables process, as the result of propagating the change of a global variable to the affected data set allocation variables.

Try the **ENT** action for the first item. Type ENT (or N) into the action field next to the item you want. The command line cannot be used for action commands on the LST mode panels.

To switch modes, press **Enter**.

Variable gathering—ENT mode

The following figure depicts the ENT mode panel corresponding to the item you selected in the LST mode panel.

```
  Help
-------------------------------------------------------------
IVP         Variable Gathering (ENT Mode)-XRF           IMS 15.1
COMMAND ===>                                             SCROLL ===> PAGE
DFSIX001: DFSIXX08 - "ENT" action complete
Action...... *   Select one of: Chg  Doc  Lst  Nxt  Prv  Rfr Imp Exp
Name........: IXUIVPHQ
User Value..: IVPIVP11
Title.......: IVP - High-level DSNAME qualifier for IVP (IVP) data sets
Blank-OK....: N  HLQ-Group...    VOL-Group...    BLK-Group...
------------------------------------------------------------------------------
**************** Top of Data ******************************
This variable specifies the high level DSNAME qualifier (HLQ) to be
used when allocating and referencing IVP data sets.

This grouping of data sets includes those data sets which are specific
to the IVP process:
  o INSTALIB, INSTATBL, ...

Data sets associated with this HLQ variable belong to the “IVP”
group. If this variable is changed, then the HLQ variables for ALL
data sets belonging to this group will be changed accordingly.

NOTE:
Press HELP for additional information on the following:
   o  Global Variables
   o  Data Set Allocation Variables
       (including the overriding of global VOLSER and BLKSIZE values)

JCL (and VSAM, if applicable) coding rules apply.

*************** BOTTOM OF DATA *****************
```

Figure 43. Variable gathering (ENT mode) panel

ENT mode provides more information for each variable:

- Whether the variable can be blank.
- Membership in the global variable groups. These fields are blank for all variables except those variables that used for data set allocation.
- A scrollable description of the variable.

You can view all the items in the variable table by using the NXT and PRV actions.

Use the DOC action to print a copy of the online help. Type DOC into the action field and press **Enter**.

Variable gathering—DOC action

The following figure shows the DOC action panel for the variable-gathering phase.

```
 Help
-------------------------------------------------------------
IVP        Variables Documentation - XRF       IMS 15.1
COMMAND ===>

   Select (1 or 2) the type of output:
_  1 - LST Mode equivalent containing names, titles, and current values
   2 - ENT Mode equivalent containing full descriptions

   Select (/) the types of variables to be documented:
_  General variables (HLQ, VOL, BLK, JOB, SMP, SYSDEF, etc.)
_  Data set allocation variables



   Press END to return to Variable Gathering.
   Press ENTER to initiate the documentation request.

   NOTE:  Output will be printed to the ISPF List data set.
   Maximum output (for XRF/ISD) is about 2K lines for Type 1
   and 12K lines for Type 2.
```

Figure 44. Variable gathering (DOC action) panel

In this panel, select the type of output listing you want:

1. LST mode equivalent
2. ENT mode equivalent

Then select the variables you want to print and press **Enter**. The requested documentation prints to the ISPF LIST data set. Even though the DOC action is entered against a single item, the resulting documentation is for all the selected types of items.

For the example in the above figure, no documentation is printed. Press **End** twice to return to LST mode, and then press **End** again to exit from the variable-gathering phase.

Related tasks

“Gathering variables” on page 18
Gathering variables involves changes to prepare the JCL and other materials that are necessary for further customization in the file-tailoring phase.

File-tailoring dialog options

Use the action commands and modes available for the file-tailoring phase.

File-tailoring action commands

Action commands are provided to support the IVP dialog during the file-tailoring phase. The following table contains the action commands, accepted modes, and command descriptions.

In the table, the capital letters in the Action column indicate the shortest allowable abbreviation for each command. The entries in the Mode column indicate whether the commands are accepted in:

LST
LST mode

ENT
ENT mode

Both
LST mode and ENT mode

Table 4. File-tailoring commands, accepted modes, and command descriptions

Action	Mode	Description
All	Both	Perform the file-tailoring phase for INSTALIB members, starting with the item for which the request is made.
brM	Both	Browse an INSTALIB member.
brS	Both	Browse an SDFSSLIB or SDFSISRC member.
Doc	Both	Print JOB/TASK/INDEX documentation to the ISPF LIST data set. The DOC action prints all items, not just the item where the DOC action is requested.
Edm	Both	Edit an INSTALIB member.
eNt	LST	Switch to ENT mode. Provides a detailed description of the item, action to be taken, expected results, and error recovery.
Ftl	Both	Perform the file-tailoring phase for a single INSTALIB member.
Lst	ENT	Switch to LST mode. LST mode presents a scrollable list of items. Within ENT Mode, the ISPF END command is also interpreted as the LST action.
Nxt	ENT	Move forward to the next item.
Prv	ENT	Move backward to the previous item.

File-tailoring—LST mode

The following figure shows the LST mode panel of the file-tailoring phase. When you enter the file-tailoring phase, you are placed in LST Mode. This mode provides the greatest visibility of the jobs and tasks available for the selected option. Except when you are browsing index items, this mode is the only mode that you need for the file-tailoring phase, because the item descriptions that are displayed in ENT mode are intended for use during the execution phase.

```
  Help
------------------------------------------------------------------
IVP  -  File Tailoring (LST Mode)  - XRF     ROW 1 OF 17 of 369
COMMAND ===>                                  SCROLL ===> PAGE

Action Codes :  All brM brS Doc Edm eNt Ftl
     Member..    Skeleton      Step
Title.....................................................................
!  IV4A001T   IVPA001T   A0    NOTE   - Step Introduction-Dialog Set-up
*  IV4A301N   DFSIXSA4   A3    CLIST  - Offline Formatted Dump - IVP1/2/3/4
*  IV4A302N   DFSIXSA5   A3    CLIST  - Offline Dump Formatter - BATCH
*  IV4A303N   DFSIXSA6   A3    CNTRL  - MSDB Load Cntrl Stmts - DBFSAMD1/DBFSA
!  IV4C001T   IVPC001T   C0    NOTE   - Step Introduction - System Definition
*  IV4C101J   DFSIXSC0   C1    JOB    - Alloc SYSDEF Data Sets
*  IV4C201T   DFSIXSC1   C2    TASK   - Browse the STAGE1 Source Deck
*  IV4C202J   DFSIXSC2   C2    JOB    - Run SYSDEF Preprocessor
*  IV4C203J   DFSIXSC3   C2    JOB    - Run SYSDEF STAGE1
*  IV4C301J   DFSIXSC4   C3    JOB    - Run SYSDEF STAGE2     >>> SEE DESCRIPT
*  IV4C401J   DFSIXSC5   C4    JOB    - Run SMP/E JCLIN
!  IV4C405T   IVPC405T   C4    TASK   - Edit IMS PROCLIB Members
!  IV4D001T   IVPD001T   D0    NOTE   - Step Introduction -  and VTAM Interf
*  IV4D101T   DFSIXSD0   D1    XMPL   - Allocate Interface Data Sets
*  IV4D201T   DFSIXSD1   D2    XMPL   - Update JESx Procedure
*  IV4D202T   DFSIXSD2   D2    XMPL   - Update BLSCECTX - DFSOFMD0 / DXRRLM50
*  IV4D203T   DFSIXSD3   D2    XMPL   - Udpate IEAAPFxx or PROGxx - Authorized
```

Figure 45. File-tailoring (LST mode) panel

If you scroll towards the bottom of the list, you can see file-tailoring items that belong to the Z series of steps. These are INDEX entries. The "Z1" items are members of SDFSSLIB (file-tailoring skeletons), which

are embedded by the earlier items. The "Z2" items are members of SDFSISRC (DBDs, PSBs, MFSs, PGMs, for example). Use the BRS action to browse these members.

Special characters are used in the action field as described below:

!
Indicates that an item was added to the table (due to service). You can remove the indicator by running the table-merge process again.

*****
Indicates that the item was processed by either the ALL action or the FTL action.

Try the ENT action for the first item. You must type ENT (or N) into the action field next to the item you want to select. The command line cannot be used for action commands on LST mode panels.

Press **Enter** to switch modes.

File-tailoring—ENT mode

The follow table shows the ENT mode panel that corresponds to the item you selected on the LST mode panel.

```
  Help
-------------------------------------------------------------------
IVP      File Tailoring(ENT Mode) - XRF      IMS 15.1  ROW 1 OF 22
COMMAND ===>                                    SCROLL ===> CSR
DFSIX001: DFSIXX09 - "ENT" action complete
Action......> DOC  <----- All brM brS Doc Edm Ftl Lst Nxt Prv
Member......: IV4A001T
Skeleton....: IVPA001T
Step........: A0
Title.......: NOTE - Step Introduction - Dialog Set-up
-------------------------------------------------------------------
o Item Type:

  NOTE - Information only

o Action Required:

  Read the description below.

  Use the "NXT" action to proceed to the next item.

o Description:

  The items within the "Ax" series of steps are used to perform
  initialization for the IVP Dialog.  There are no user
  executable JOBs within these steps.  Please skip to the first
  item for Step C0 (This is the default starting position for
  the Execution Phase of the IVP Dialog).
```

Figure 46. File-tailoring (ENT mode) panel

The only additional information provided by ENT Mode is the scrollable item description. Except for the INDEX items, these descriptions are intended for the execution phase and have no meaning for the file-tailoring phase.

The NXT and PRV actions can be used to view all the items in the file-tailoring table.

Use the DOC action to print a copy of the online help. Type DOC into the action field and press **Enter**.

File-tailoring—DOC action

The following table depicts the DOC action panel for the file-tailoring phase.

```
Help
-------------------------------------------------------------
IVP       JOB/TASK/INDEX Documentation - XRF     IMS 15.1
COMMAND ===>

  Select (1 or 2) the type of output and press Enter.
  1 - LST Mode equivalent containing names and titles
  2 - ENT Mode equivalent containing full descriptions

  Select (/) the types of JOBs/TASKs to be documented:
   SETUP  - IVP Preparation (CLISTs, Control statements)
   IVP        - IMS System Definition
   IVP        - /VTAM Interface
   IVP        - IVP System and Application Build
   IVP        - IVP Execution
   INDEX  - DFSSLIB (IMBEDs) and DFSISRC members


   NOTE:  Output will be printed to the ISPF List data set.
   Maximum output (for XRF/ISD) is about 3K lines for Type 1
   and 22K lines for Type 2.
```

Figure 47. File-tailoring (DOC action) panel

In this panel, select the type of output listing you want:

1. LST mode equivalent
2. ENT mode equivalent

Then select which items you want to print and press **Enter**. The requested documentation prints to the ISPF LIST data set.

Even though the DOC action is entered against a single item, the resulting documentation is for all the selected types of items.

You can print the documentation for jobs and tasks during either the file-tailoring phase or the execution phase. The documentation for index items prints only from the file-tailoring phase.

For the example in the file-tailoring (DOC action) panel, no documentation is printed. Press **End** twice to return to LST mode and then press END again to exit from the file-tailoring phase.

Related tasks

"Tailoring files" on page 22
In the file-tailoring phase, the IVP uses variables that you specified during the variable-gathering phase to prepare a customized set of IVP JCL and tasks to be stored as members of the INSTALIB data set for use in the execution phase.

Execution phase dialog options

Use the action commands and modes available for the Execution phase.

Execution action commands

Action commands are provided to support the IVP dialog during the execution phase. The table below contains the action commands, accepted modes, and command descriptions.

Two modes are used to display the jobs and tasks:

LST
: The items are presented in a scrollable list. Each item represents one job or task. LST is the default.

ENT
: The job and task members are presented one at a time in sequence. Scrollable information is provided to describe each item.

In the following table, the capital letters in the action column indicate the shortest allowable abbreviation for each command. The entries in the mode column indicate whether the commands are accepted in:

LST
LST mode

ENT
ENT mode

Both
Both LST mode and ENT mode

Table 5. Execution action commands

Action	Mode (LST or ENT)	Description
br**M**	Both	Browse an INSTALIB member.
Doc	Both	Print job or task documentation to the ISPF LIST data set. The DOC action prints all items, not only the item where the DOC action is requested.
Edm	Both	Edit an INSTALIB member.
e**N**t	LST	Switch to ENT mode. ENT mode presents items one at a time on a formatted screen.
e**X**e	Both	Use the TSO SUBMIT command to submit an INSTALIB job for execution. Alternatively, you can issue the TSO SUBMIT command directly while editing an INSTALIB member through the EDM action (see EDM in this table).
Ftl	LST	File tailor an individual member.
Lst	ENT	Switch to LST mode. LST mode presents a scrollable list of items. Provides a detailed description of the item, actions to be taken, expected results, and error recovery.
Nxt	ENT	Move forward to the next item.
Prv	ENT	Move backward to the previous item.
sp**R**	Both	Execute a special processing routine that has been provided to assist with the performance of a task.

Execution phase—LST mode

When you enter the execution phase, you are in LST mode. The following depicts the LST mode panel of the execution phase.

```
  Help
-------------------------------------------------------------
IVP              Execution (LST Mode) - XRF               ROW 11 to 26 OF 177
COMMAND ===>                                                SCROLL ===> PAGE

Action Codes :  Brm Doc Edm eNt eXe spR
     JOB/Task  Step         Title.............................
!    IV4C001T  C0   NOTE  - Step Introduction - System Definition
!    IV4C101J  C1   JOB   - Allocate SYSDEF Data Sets
!    IV4C201T  C2   TASK  - Browse the STAGE1 Source Deck
!    IV4C202J  C2   JOB   - Run SYSDEF Preprocessor
!    IV4C203J  C2   JOB   - Run SYSDEF STAGE1
*    IV3C301J  C3   JOB   - Run SYSDEF STAGE2  >>> SEE DESCRIPTION
!    IV3C401J  C4   JOB   - Run SMP/E JCLIN
!    IV3C405T  C4   TASK  - Edit IMS PROCLIB Members
!    IV3D001T  D0   NOTE  - Step Introduction -  and VTAM Interface
!    IV3D101T  D1   XMPL  - Allocate Interface Data Sets
!    IV3D201T  D2   XMPL  - Update JESx Procedure
!    IV3D202T  D2   XMPL  - Update BLSCECTX - DFSOFMD0 / DXRRLM50
!    IV3D203T  D2   XMPL  - Update IEAAPFxx or PROGxx - Authorized DSN
!    IV3D204T  D2   XMPL  - Update IEALPAxx - MLPA Modules
!    IV3D207T  D2   XMPL  - Update IEASVCxx - SVC Numbers
!    IV3D208T  D2   XMPL  - Update SCHEDxx - PPT Entries
```

Figure 48. Execution phase (LST mode) panel

LST Mode provides the greatest visibility of the items that make up the IVP process. However, only minimal descriptive information is provided. Do not use LST mode for the execution phase until you are familiar with the requirements for each job and task. Use ENT mode instead.

Special characters in the action field indicate changes to variables:

!
: Indicates that an item was added to the table (due to service).

*****
: Indicates that the item was processed by either the SUB action or EDM action. (The dialog assumes that if you edited an item, you have also submitted that item.)

You can remove the ! and * indicators by running the table-merge process again.

Try the ENT action for the second item. Type ENT (or N) in the action field.

To switch modes, press **Enter**.

Execution phase—ENT mode

The following figure shows the ENT mode panel of the execution phase. The item displayed is the item for which you requested the ENT action on the LST mode panel.

```
  Help
-------------------------------------------------------------------
IVP           Execution (ENT Mode) - XRF               IMS 15.1
COMMAND ===>                                  SCROLL ===> PAGE
DFSIX001: DFSIXX11 - "ENT" action complete
Action...... !    Select one of: Brm Doc Edm eXe Lst Nxt Prv spR
JOB or Task....: IV4C101J
Step........: C1
Title.......: JOB - Alloc SYSDEF Data Sets
-------------------------------------------------------------------
*********** Top of Data ********************
o Item Type:

  JOB - The batch JOB provided for this item must be run.

o Action Required:

  1. Review “Description” below.
  2. If desired (or required by the description below), use the
     “EDM” action to edit the supplied JOB. (The “BRM” action
     can be used to browse the JOB.)
  3. Submit the JOB to be run. Use the TSO SUBMIT command
     from within edit or use the “EXE” action from the Execution
     Phase panels.
  4. When the JOB completes execution, review all step completion
     for successful completion. See “Completion Codes and Messages”
     below for acceptable completion codes for this JOB.

  It might be necessary to review the printed output generated by
  this JOB in order to verify successful completion.

  If the JOB does not complete successfully, see “Error Recovery” below.

  5. When you are satisfied that the JOB completed successfully, use
     the “NXT” action to proceed to the next item.

o Description:
  This JOB scratches and reallocates the data sets needed, in addition to
  those allocated for SMP/E processing, for IMS system definition.
```

Figure 49. Execution phase (ENT mode) panel

The NXT and PRV actions can be used to progress through the items in the execution table.

When you are familiar with the requirements for each job and task (possibly by browsing the entire process before you actually begin submitting jobs), you can switch back to LST mode.

For this example, press **End** to return to LST mode and then press **End** again to exit from the execution phase.

Related tasks

“Executing tailored jobs and tasks” on page 23
You must process the jobs and tasks that were prepared by the file-tailoring phase individually through the execution phase.

Sample application parts tables and PSBs

The sample application parts tables describe the parts of the sample applications.

For example, the parts tables tell you the language in which a particular part of the sample application is written, the name of the databases that use that part, and a description of the database. In some cases, program specification blocks (PSBs) are provided to process the database that is used by the sample application.

IVP sample application table

The SDFSISRC target library contains the source for programs, PSBs, DBDs, and MFSs, and other supporting materials that are used by the application.

The following table provides information about the parts that are used by the IVP sample application. It includes the language, PSB, MFS, transaction code, DBD, JCL, and description that are associated with those parts (where applicable).

Table 6. IVP sample application parts

Part name	Language	PSB	MFS	TRANCD	DBD	Compile and BIND JCL	Description
DFSIVD1	assembler	n/a	n/a	n/a	DFSIVD1	n/a[1]	HIDAM/OSAM database
DFSIVD1I	assembler	n/a	n/a	n/a	DFSIVD1I	n/a[1]	HIDAM/OSAM primary index database
DFSIVD2	assembler	n/a	n/a	n/a	DFSIVD2	n/a[1]	HDAM/VSAM database
DFSIVD3	assembler	n/a	n/a	n/a	DFSIVD3	n/a[1]	DEDB database
DFSIVD4	assembler	n/a	n/a	n/a	DFSIVD4	n/a[1]	MSDB database
DFSIVD5	assembler	n/a	n/a	n/a	DFSIVD5	n/a[1]	GSAM/BSAM database
DFSIVA1	assembler	DFSIVP1	DFSIVF1	IVTNO	DFSIVD1	n/a[1]	Non-conversational MPP
	assembler	DFSIVPD2	DFSIVF6	IVTND	DFSIVD1	n/a	Non-conversational MPP
DFSIVA2	assembler	DFSIVP2	DFSIVF2	IVTNV	DFSIVD2	n/a[1]	Non-conversational MPP
DFSIVA3	assembler	DFSIVP3	DFSIVF3	IVTCV	DFSIVD2	n/a[1]	Conversational MPP
DFSIVA31[2]	Pascal	DFSIVP31	DFSIVF31	IVTCP	DFSIVD2	DFSIVJP3	Conversational MPP
DFSIVA32[2]	C	DFSIVP32	DFSIVF32	IVTCC	DFSIVD2	DFSIVJC3	Conversational MPP
DFSIVA34[2]	COBOL	DFSIVP34	DFSIVF34	IVTCB	DFSIVD2	DFSIVJB3	Conversational MPP
DFSIVA35[5]	REXX	DFSIVP35	DFSIVF35	IVTCX	DFSIVD2	n/a	Conversational MPP
DFSIVP37	Java	DFSIVP37	DFSIVF37	IVTCM	DFSIVD2	n/a	Conversational JMP
DFSIVA4	assembler	DFSIVP4	DFSIVF4	IVTFD	DFSIVD3	n/a	Non-conversational IFP (EMH)
DFSIVA5	assembler	DFSIVP5	DFSIVF5	IVTFM	DFSIVD4	n/a	Non-conversational IFP (EMH). Display and Replace only
DFSIVA6	assembler	DFSIVP6	n/a	n/a	DFSIVD1	n/a	DB batch, BMP
DFSIVA61[6]	Pascal	DFSIVP61	n/a	n/a	DFSIVD1	DFSIVJP6	DB batch, BMP

Table 6. IVP sample application parts (continued)

Part name	Language	PSB	MFS	TRANCD	DBD	Compile and BIND JCL	Description
DFSIVA62[6]	C	DFSIVP62	n/a	n/a	DFSIVD1	DFSIVJC6	DB batch, BMP
DFSIVA64[6]	COBOL	DFSIVP64	n/a	n/a	DFSIVD1	DFSIVJB6	DB batch, BMP
DFSIVA65[5]	REXX	DFSIVP65	n/a	n/a	DFSIVD1	n/a	DB batch, BMP
DFSIVP67	Java	DFSIVP67	n/a	n/a	DFSIVD2	n/a	JBP
DFSIVA7	assembler	DFSIVP7	n/a	n/a	DFSIVD2	n/a	DB batch, BMP
	assembler	DFSIVPD3	n/a	n/a	IVPDRD1	n/a	DB batch, BMP
DFSIVA8	assembler	DFSIVP8	n/a	n/a	DFSIVD3	n/a	DB batch, BMP
n/a	n/a	DFSIVP9	n/a	n/a	DFSIVD1	n/a	Online image copy
DFSDDLT0	n/a	DFSIVPA	n/a	n/a	DFSIVD1	n/a	HIDAM load
DFSDDLT0	n/a	DFSIVPB	n/a	n/a	DFSIVD2	n/a	HDAM load
DFSIVAC	assembler	DFSIVPC	n/a	n/a	DFSIVD3	n/a	DEDB load BMP
DFSIVAD	assembler	DFSIVPD	DFSIVFD	IVTC1[7]	Simulated	n/a	Message-driven WFI BMP
	assembler	DFSIVPH	DFSIVFH	IVTC5	Simulated	n/a	Message-driven WFI BMP
DFSIVAE	assembler	DFSIVPE	DFSIVFE	IVTC2[7]	n/a	n/a	Non-conversational MPP. MSG switch to DFSIVAD
DFSIVAF	assembler	DFSIVPF	DFSIVFF	IVTC3[7]	n/a	n/a	Conversational MPP. MSG switch to DFSIVAD
DFSIVAG	assembler	DFSIVPG	DFSIVFG	IVTC4[7]	n/a	n/a	IFP (EMH). MSG switch to DFSIVAD
DFSIVG20[8]	assembler	n/a	n/a	n/a	n/a	DFSIVJG2	WTOR routine for Pascal
DFSIVG30[8]	assembler	n/a	n/a	n/a	n/a	DFSIVJG3	WTOR routine for C
DFSIVC04	assembler	n/a	n/a	n/a	n/a	n/a	Control statements for HD DB load (DFSDDLT0)
DFSIVC05	assembler	n/a	n/a	n/a	n/a	n/a	DB batch, BMP GSAM input
DFSIVC06	assembler	n/a	n/a	n/a	n/a	n/a	Control statements for MSDB load
DFSIVC07	assembler	n/a	n/a	n/a	n/a	n/a	WFI BMP GSAM input

Table 6. IVP sample application parts (continued)

Part name	Language	PSB	MFS	TRANCD	DBD	Compile and BIND JCL	Description

[1] These parts are installed by the IVP.

[2] After being compiled and bound, programs DFSIVA31/32/34 are executable from any 24x80 (3270) MFS device. You must add runtime libraries for either PL/I or Pascal to the IVP execution JCL.

[5] Programs DFSIVA35 and DFSIVA65 are fully installed by the IVP. DFSIVA35 can be executed from any 24x80 (3270) MFS device. DFSIVA65 can be executed by modifying the IVP execution JCL for DFSIVA6.

[6] After being compiled and bound, programs DFSIVA61/62/64 can be execute by modifying the IVP execution JCL for DFSIVA6. You must add runtime libraries for either PL/I or Pascal to the IVP execution JCL.

[7] These transaction codes are provided only in a DCCTL system.

[8] DFSIVG20 and DFSIVG30 are assembler subroutines that provide WTOR support for the Pascal and C programs.

Related concepts

"IVP sample application" on page 25
The IVP sample application is a simple telephone book application, also known as the phonebook application, that sends a transaction to request information from the IMS Telephone database.

Related reference

"IVP sample application databases" on page 150
In the DCCTL environment, the IVP database is simulated through the use of a data area within program DFSIVAD.

IMS sample application table

The following table provides information about the parts used by the IMS sample application. It includes the language, PSB, transaction code, DBD, and description associated with those parts.

Table 7. IMS sample application parts

Part name	Language	PSB	TRANCD	DBD	Description
DI21PART	assembler	n/a	n/a	DI21PART	HISAM/VSAM database
DFSSAM1	COBOL	DFSSAM11	n/a	DI21PART	DB batch - Database Load
DFSSAM2	COBOL	DFSSAM12	PART	DI21PART	Non-conversational MPP
DFSSAM3	COBOL	DFSSAM13	DSPINV	DI21PART	Non-conversational MPP
DFSSAM4	COBOL	DFSSAM14	ADDPART ADDINV DLETPART DLETINV	DI21PART	Non-conversational MPP
DFSSAM5	COBOL	DFSSAM15	CLOSE	DI21PART	Non-conversational MPP
DFSSAM6	COBOL	DFSSAM16	DISBURSE	DI21PART	Non-conversational MPP
DFSSAM7	COBOL	DFSSAM17	DSPALLI	DI21PART	Non-conversational MPP
DFSSAM08	assembler	DFSSAM18	n/a	DI21PART	DB batch
DFSDDLT0	assembler	DFSSAM19	n/a	DI21PART	DB batch/BMP
DFSSUT04	REXX	n/a	n/a	n/a	Status code subroutine

Table 7. IMS sample application parts (continued)

Part name	Language	PSB	TRANCD	DBD	Description
MFDFSYSN	n/a	n/a	n/a	n/a	Control statements for database load
DFSSAMC1	n/a	n/a	n/a	n/a	Control statements for database dump (DFSDDLT0)

IMS sample application PSBs

These PSBs are provided to process the Parts database that is used by the IMS sample application.

The following figure shows the PSBs that are provided for you to run the IMS sample application from an IMS user terminal.

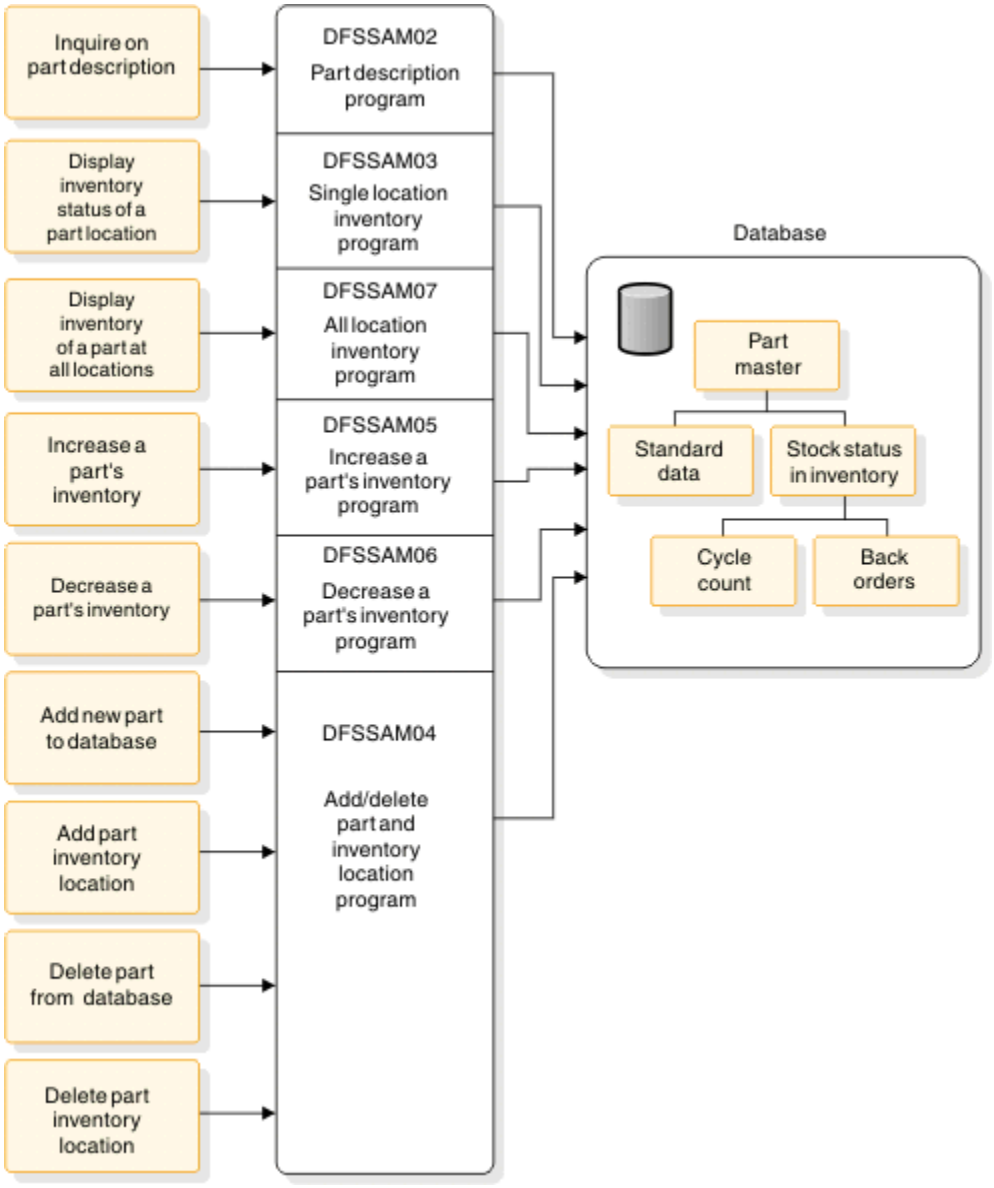


Figure 50. PSBs processing the Parts database

The six PSBs and their associated transactions enable you to perform the following nine online functions:

MPP transaction	Online function
PART	Inquire about a part and its description.
DSPALLI	Inquire about a part's inventory, cycle count, and back-order information.
DSPINV	Inquire about a part's total inventory in all locations or by specific inventory location.
ADDPART	Add a new part and its description.
ADDINV	Add part inventory information, by location, to an existing part description.
DLETINV	Delete part inventory information, by location.
DLETPART	Delete a part after deletion of all its subordinate part inventory information.
CLOSE	Close a part order to increase the part inventory at a specific location.
DISBURSE	Disburse a specific quantity of a particular part, on a planned or unplanned basis, at a particular part inventory location, reducing inventory.

Related tasks

"Running the IMS sample application" on page 32
To verify the IMS sample application databases, you must run the IMS sample application using the six PSBs with their associated transactions and nine online functions.

Related reference

"IVP sample application table" on page 144
The SDFSISRC target library contains the source for programs, PSBs, DBDs, and MFSs, and other supporting materials that are used by the application.

Fast Path sample application table

The following table provides information about the parts used by the Fast Path sample application. It includes the language, PSB, MFS, transaction code, DBD, and description associated with those parts.

Table 8. Fast Path sample application parts

Part name	Language	PSB	MFS	TRANCD	DBD	Description
DBFSAMD1	assembler	n/a	n/a	n/a	DBFSAMD1	MSDB - General Ledger Database
DBFSAMD2	assembler	n/a	n/a	n/a	DBFSAMD2	MSDB - Teller Database
DBFSAMD3	assembler	n/a	n/a	n/a	DBFSAMD3	DEDB/VSAM - Customer Account Database
DBFSAMD4	assembler	n/a	n/a	n/a	DBFSAMD4	HDAM/VSAM - Loan Database
DBFSAMA1	assembler	DBFSAMP1	n/a	n/a	DBFSAMD3	BMP - DEDB/VSAM load
DBFSAMA2	assembler	DBFSAMP2	n/a	n/a	DBFSAMD4	DB Batch - HDAM/ VSAM load
DBFSAMA3	assembler	DBFSAMP3	DBFSAMF1	FPSAMP1	DBFSAMD1 DBFSAMD2 DBFSAMD3 DBFSAMD4	Non-conversational IFP (EMH)

Table 8. Fast Path sample application parts (continued)

Part name	Language	PSB	MFS	TRANCD	DBD	Description
DBFSAMA3	assembler	DBFSAMP4	DBFSAMF1	FPSAMP2	DBFSAMD1 DBFSAMD2 DBFSAMD3 DBFSAMD4	Non-conversational MPP
DFSDDLT0	assembler	DBFSAMP5	n/a	n/a	DBFSAMD4	DB batch/BMP - HDAM/VSAM
DFSDDLT0	assembler	DBFSAMP6	n/a	n/a	DBFSAMD3	BMP - DEDB/VSAM
DFSIVC06	n/a	n/a	n/a	n/a	DBFSAMD1 DBFSAMD2	MSDB load control statements

Related concepts

"Fast Path sample application" on page 37
The Fast Path sample application demonstrates a banking application.

Related reference

"Fast Path sample application error messages" on page 154
Use the following information to diagnose Fast Path sample application errors.

Partitioning sample application table

The parts used by the IVP sample partitioning application are identified in the following table.

These parts are all installed by the IVP.

Table 9. IVP sample partitioning application parts

Part name	Language	PSB	MFS	TRANCD	DBD	Compile and BIND JCL	Description
DFSIVD1	assembler	n/a	n/a	n/a	DFSIVD1	n/a[1]	PHIDAM/OSAM database
DFSIVD1	assembler	n/a	n/a	n/a	DFSIVD1I	n/a[1]	PHIDAM/OSAM Primary Index database
DFSIVA1	assembler	DFSIVP1	DFSIVF1	IVTNO	DFSIVD1	n/a[1]	Non-conv. MPP

Related concepts

"Partitioning sample application" on page 27
The Partitioning sample application provided by the IVP demonstrates the conversion of a non-partitioning database to a partitioned database.

IMS Connect sample application table

The following table shows the parts used by the IMS Connect sample application. These parts are all installed by the IVP jobs and tasks.

Table 10. IVP sample IMS Connect parts

Part name	Language	PSB	TRANCD	DBD	Description
DFSSAM02	REXX	DFSSAM12	PART	DI21PART	Part inquiry programs

Related concepts

"IMS Connect sample application" on page 27

The IMS Connect sample application demonstrates that IMS Connect is operational by running a full-function MPP transaction using IMS Connect.

Sample application database

The IVP jobs and tasks load the databases that the sample applications query in order to test particular components of IMS.

- "IVP sample application databases" on page 150
- "Fast Path sample application databases" on page 153
- "Partitioning sample application databases" on page 154

IVP sample application databases

In the DCCTL environment, the IVP database is simulated through the use of a data area within program DFSIVAD.

Each of four root-only databases in the IVP contains the same six records. The following figure displays the contents (last name, first name, extension number, and zip code) of these records.

Table 11. Contents of IVP root-only database records

Record number	Last name	First name	Ext. number	Zip code
1	LAST1	FIRST1	8-111-1111	D01/R01
2	LAST2	FIRST2	8-111-2222	D01/R02
3	LAST3	FIRST3	8-111-3333	D01/R03
4	LAST4	FIRST4	8-111-4444	D02/R04
5	LAST5	FIRST5	8-111-5555	D02/R05
6	LAST6	FIRST6	8-111-6666	D03/R06

DFSIVD1 - HIDAM/OSAM

- Database description

  Database name:
  IVPDB1

  Segment name:
  A1111111

  Segment length:
  40

  Key field name:
  A1111111

  Key field length:
  10

- Database Record Format:

Table 12. Database record format of DFSIVD1

Offset	Length	Field name	Description
0	10	A1111111	Last Name
10	10	N/A	First Name
20	10	N/A	Extension Number

Table 12. Database record format of DFSIVD1 (continued)			
Offset	**Length**	**Field name**	**Description**
30	7	N/A	Internal Zip Code
37	3	N/A	Reserved

DFSIVD2 - HDAM/VSAM

- Database description

  Database name:
  IVPDB2

  Segment name:
  A1111111

  Segment length:
  40

  Key field name:
  A1111111

  Key field length:
  10

- Database record format:

Table 13. Database record format of DFSIVD2			
Offset	**Length**	**Field name**	**Description**
0	10	A1111111	Last Name
10	10	N/A	First Name
20	10	N/A	Extension Number
30	7	N/A	Internal Zip Code
37	3	N/A	Reserved

DFSIVD3 - DEDB/VSAM

- Database Description

  Database name:
  IVPDB3

  Segment name:
  A1111111

  Segment length:
  42

  Key field name:
  A1111111

  Key field length:
  10

- Database record format:

Table 14. Database record format of DFSIVD3			
Offset	**Length**	**Field name**	**Description**
0	2	N/A	Segment Length

Table 14. Database record format of DFSIVD3 (continued)			
Offset	**Length**	**Field name**	**Description**
2	10	A1111111	Last Name
12	10	N/A	First Name
22	10	N/A	Extension Number
32	7	N/A	Internal Zip Code
39	3	N/A	Reserved

DFSIVD4 - MSDB

- Database description

  Database name:
  IVPDB4

  Segment name:
  A1111111

  Segment length:
  40

  Key field name:
  A1111111

  Key field length:
  10

- Database record format:

Table 15. Database record format of DFSIVD4			
Offset	**Length**	**Field name**	**Description**
0	10	A1111111	Last Name
10	10	N/A	First Name
20	10	N/A	Extension Number
30	7	N/A	Internal Zip Code
37	3	N/A	Reserved

DFSIVD5 - GSAM/BSAM

- Database description

  Database name:
  IVPDB5

  RECFM:
  F

  RECORD:
  80

Related concepts

“IVP sample application” on page 25
The IVP sample application is a simple telephone book application, also known as the phonebook application, that sends a transaction to request information from the IMS Telephone database.

Related reference

“IVP sample application table” on page 144

The SDFSISRC target library contains the source for programs, PSBs, DBDs, and MFSs, and other supporting materials that are used by the application.

Fast Path sample application databases

The transactions that are described in the following tables can help you become familiar with the Fast Path sample application databases. Along with the following customer account information, the tables provide you the resources that you can use to prepare online training exercises for operators and programmers.

The following table shows customer account numbers for the Customer Savings Account database that is loaded into the DEDB, which can be used to run the Fast Path sample application.

Table 16. Customer Savings Account database – root segment (DEDB)

Customer account number	Customer name	Customer address	Account type	Account balance $
BR01-B01S1 C1	Robert Bennett	1601 California Ave. Palo Alto, CA 95432	S JT C TS	4,000.00 1,500.00
BR01-A01S1	Mary Adams	2044 Hamilton Ave. Campbell, CA 95030	S JT	2,000.00
BR01-F01S1 C1	John Ford	4312 Skyline Road Mt. View, CA 96048	S BA C TR	15,000.00 800.00
BR01-H01C1	Betty Hill	7676 Santa Teresa Rd San Jose, CA 97050	C TR	6,000.00
BR02-B02U1	Samuel Brown	9624 Prospect Ave. San Jose, CA 95129	U UB	13,000.00
BR02-T01U1	James Taylor	5411 Ocean Dr. Santa Cruz, CA 96080	U UA	41,500.00
BR02-T02C1	Peter Thomas	1900 Stanford Ave. Palo Alto, CA 95432	C TR	9,000.00

The following table shows customer account numbers for the Customer Loan Account database that is loaded into the HDAM database, which can be used to run the Fast Path sample application.

Table 17. Customer Loan Account database (HDAM)

Customer account number	Customer name	Loan amount $	Monthly payment $
BR01-B01A1	Robert Bennett	4,500.00	145.00
BR01-A01V1	Mary Adams	1,200.00	106.06
BR01-F01H1	John Ford	60,000.00	76.01
BR01-H01M1	Betty Hill	60,000.00	482.77
BR02-B02P1	Samuel Brown	1,000.00	88.38
BR02-T01H1	James Taylor	6,000.00	76.01

Table 17. Customer Loan Account database (HDAM) (continued)

Customer account number	Customer name	Loan amount $	Monthly payment $
BR02-T02A1	Peter Thomas	4,000.00	129.07

Related concepts

"Fast Path sample application" on page 37
The Fast Path sample application demonstrates a banking application.

Related reference

"Fast Path sample application error messages" on page 154
Use the following information to diagnose Fast Path sample application errors.

Partitioning sample application databases

The following description about the database can assist you in becoming familiar with the sample database used by the partitioning sample application.

- Database description

  Database name:
  IVPDB1

  Segment name:
  A1111111

  Segment length:
  40

  Key field name:
  A1111111

  Key field length:
  10

- Database record format:

Table 18. Database record format of DFSIVD1

Offset	Length	Field name	Description
0	10	A1111111	Last name
10	10	N/A	First name
20	10	N/A	Extension number
30	7	N/A	Internal zip code
37	3	N/A	Reserved

Related concepts

"Partitioning sample application" on page 27
The Partitioning sample application provided by the IVP demonstrates the conversion of a non-partitioning database to a partitioned database.

Fast Path sample application error messages

Use the following information to diagnose Fast Path sample application errors.

Error message format

```
REQUEST CAN NOT BE SERVICED:
PROCSG ERROR xx yy zz...
```

Where:

xx:
Error code set by application program

yy:
PCB status code, if applicable

zz:
Input data

ERROR CODES

IE -
Invalid input data

LM -
Missing loan segment (HDAM)

LU -
Error in updating loan segment (HDAM)

MA -
Missing customer account segment (DEDB)

MR -
Missing customer root segment (DEDB)

MT -
Missing teller segment (MSDB)

MX -
Missing transaction segment (DEDB)

OD -
Transaction amount on withdrawal greater than customer account balance

RB -
Error in processing and rollback

TR -
Terminal transmission error on input

UA -
Error in updating account segment (DEDB)

UG -
Error in updating general ledger (MSDB)

UI -
Error in adding sequential dependent (DEDB)

UT -
Error in updating teller database (MSDB)

UX -
Error in adding account transaction segment (DEDB)

Related tasks

“Running the Fast Path sample transactions” on page 39
Run the Fast Path sample transactions from your terminal to verify the proper setup of your Fast Path infrastructure.

Related reference

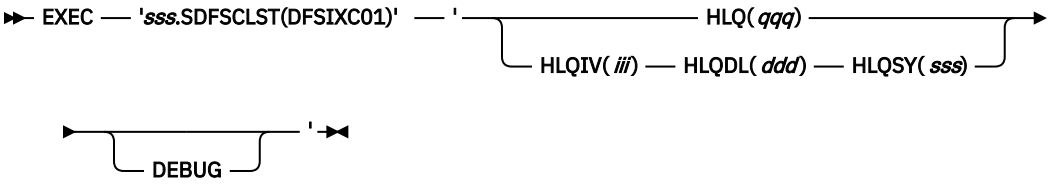
“Fast Path sample application table ” on page 148
The following table provides information about the parts used by the Fast Path sample application. It includes the language, PSB, MFS, transaction code, DBD, and description associated with those parts.

“Fast Path sample application databases” on page 153
The transactions that are described in the following tables can help you become familiar with the Fast Path sample application databases. Along with the following customer account information, the tables

provide you the resources that you can use to prepare online training exercises for operators and programmers.

TSO EXEC command syntax for invoking the IVP start-up CLIST

The following syntax diagram illustrates how to invoke the IVP start-up CLIST using the TSO EXEC command.



sss
The IMS system (SYS) data sets.

The default is IVPDLB15.

EXEC
TSO command to run CLISTs and REXX EXECs.

HLQ(*qqq*)
The keyword that identifies the high-level qualifier for the IVP, system, and distribution libraries data set (*qqq*).

HLQIV(*iii*)
The keyword that identifies the high-level qualifier for the IVP data sets, INSTALIB and INSTATBL (*iii*).

The default is IVPIVP15.

HLQDL(*ddd*)
The keyword that identifies the high-level qualifier for the IMS distribution library (DLB) data sets (*ddd*).

The default is IVPDLB15.

HLQSY(*sss*)
The keyword that identifies the high-level qualifier for the IMS system (SYS) data sets (*sss*).

The default is IVPSYS15.

DEBUG
A parameter that sets diagnostic options and causes trace information to be displayed; each line of the ISPF code is displayed as it is processed.

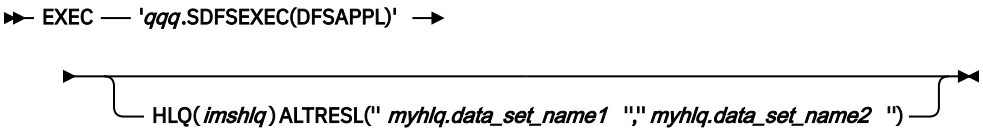
Related tasks

"Starting the IVP by using the EXEC command from within ISPF" on page 11

You can start the IVP dialog from within ISPF either by using partial syntax with a simple command or by using full syntax.

REXX EXEC command syntax for starting the IMS Application Menu

The following syntax applies to the REXX EXEC command for starting the IMS Application Menu.



qqq
High-level qualifiers of the IMS REXX data set.

DFSAPPL
REXX EXEC for starting the IMS Application menu.

imshlq
The high-level qualifier of the IMS data sets.

ALTRESL*(''myhlq.data_set_ name1'')*
Keyword that specifies a list of data sets that contain load modules.

myhlq.data_set_name1 is the fully qualified name of the data set that contains the load modules. If you specify the ALTRESL parameter, you should include the SDFSRESL data set in the list of data set names. If you do not specify the ALTRESL parameter, *imshlq*.SDFSRESL is used as the ISPLLIB data set.

If you need to specify more than one data set, separate the names with a comma.

Because both the EXEC command and data set names use single quotation marks as punctuation characters, use two adjacent single quotation marks when you specify a data set name within the EXEC parameter string.

Related tasks

“Starting the IVP from the IMS Application Menu” on page 12
You can start the IVP dialog from the IMS Application Menu.

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

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

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

IBM Director of Licensing IBM Corporation North Castle Drive, MD-NC119 Armonk, NY 10504-1785 US

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

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

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

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 Director of Licensing IBM Corporation North Castle Drive, MD-NC119 Armonk, NY 10504-1785 US

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

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and

cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows: © (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_.

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 http://www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, 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.

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

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

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

IBM Online Privacy Statement

IBM Software products, including software as a service solutions, (“Software Offerings”) may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering’s use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

To learn more, see IBM Privacy Statement.

Bibliography

This bibliography lists all of the publications in the IMS 15 library.

Title	Acronym	Order number
IMS Version 15 Application Programming	APG	SC27-6778
IMS Version 15 Application Programming APIs	APR	SC27-6779
IMS Version 15 Commands, Volume 1: IMS Commands A-M	CR1	SC27-6780
IMS Version 15 Commands, Volume 2: IMS Commands N-V	CR2	SC27-6781
IMS Version 15 Commands, Volume 3: IMS Component and z/OS Commands	CR3	SC27-6782
IMS Version 15 Communications and Connections	CCG	SC27-6783
IMS Version 15 Database Administration	DAG	SC27-6784
IMS Version 15 Database Utilities	DUR	SC27-6785
IMS Version 15 Diagnosis	DGR	GC27-6786
IMS Version 15 Exit Routines	ERR	SC27-6787SC19-4208
IMS Version 15 Installation	INS	SC27-6788
IMS Version 15 Licensed Program Specifications	LPS	GC27-6799
IMS Version 15 Messages and Codes, Volume 1: DFS Messages	MC1	GC27-6789
IMS Version 15 Messages and Codes, Volume 2: Non-DFS Messages	MC2	GC27-6790
IMS Version 15 Messages and Codes, Volume 3: IMS Abend Codes	MC3	GC27-6791
IMS Version 15 Messages and Codes, Volume 4: IMS Component Codes	MC4	GC27-6792
IMS Version 15 Operations and Automation	OAG	SC27-6793
IMS Version 15 Release Planning	RPG	GC27-6794
IMS Version 15 System Administration	SAG	SC27-6795
IMS Version 15 System Definition	SDG	GC27-6796
IMS Version 15 System Programming APIs	SPR	SC27-6797
IMS Version 15 System Utilities	SUR	SC27-6798

Index

Special Characters

A

B

C

D

E

F

G

I

K

L

M

N

O

P

Q

R

S

T

U

V

X

Product Number: 5635-A06
5655-DS5
5655-TM4

GC27-6788-00