IBM InfoSphere Optim for z/OS
Version 11 Release 3

Customization Guide

IBM

IBM InfoSphere Optim for z/OS
Version 11 Release 3

Customization Guide

IBM

Version 11 Release 3

This edition applies to version 11, release 3 of IBM InfoSphere Optim for z/OS and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

About this publication

This document describes how to customize and maintain the components of IBM® InfoSphere® Optim™ for z/OS®.

InfoSphere Optim for z/OS is hereinafter referred to as Optim.

For detailed information on each component of Optim, refer to the appropriate user manual and the *Common Elements Manual*.

Chapter 1. Overview - Customization Guide

This section provides an overview of the steps that are required to install Optim. This section also contains a description of the installation process for remote subsystem access, the upgrade process, and administration tasks.

IBM Optim z/OS manages enterprise data throughout every stage of the information lifecycle. Optim enables you to assess, classify, subset, archive, store, and access enterprise application data. Optim uses the relationships defined in the DB2® Catalog, where available, and supplements these relationships with those defined in the Optim Directory. Optim runs as a TSO/ISPF application and incorporates familiar ISPF commands. Optim handles any number of tables and any number of relationships, regardless of the complexity.

The Optim components include Access, Archive, Compare, and Move. Your site may be licensed for one, all, or a combination of these components. The Main Menu provides release and copyright information for Optim.

Software Requirements

Optim requires IBM DB2 Release 10.1 or later. Supported operating systems are z/OS 1.12 and later. All current maintenance should be applied.

Refer to the *Program Directory for IBM Infosphere Optim for z/OS* for full details on disk space required.

Memory requirements vary according to the operation performed and the data processed. If insufficient memory or unsuccessful sort errors occur, verify the batch or TSO region size:
- If you are using the EMC Centera networked storage system, the TSO region size should be 200000.
- A TSO region size that is a minimum of 220 MB is required to use the Optim data privacy providers.

`Prerequisites`

Optim uses DSNREXX from the SDSNLOAD library to obtain information regarding the DB2 release Optim is being installed on. The DSNREXX plan must be bound and accessible by the Optim installer in the target DB2 subsystem. The DB2 load library containing the DSNREXX load module must be made available either by the TSO logon procedure (STEPLIB or ISPLLIB), the system link list or the link pack area. If Optim fails to obtain the DB2 release information using DSNREXX, Optim will attempt to load DSNHDECP to obtain the release value.

For all DB2 releases:

grant all Optim users SELECT access to SYSIBM.SYSTABLESPACESTATS or SYSIBM.TABLESPACESTATS

grant all Optim users SELECT authority on SYSIBM.SYSTABLESPACE catalog table

Centera Pool Entry Authorization File

If you are using the EMC Centera networked storage system, access to the Centera server can be secured with a Pool Entry Authorization (PEA) file.

A PEA file prevents unauthorized applications from storing data on or retrieving data from a Centera server. To use a PEA file with Optim, you must create the file and specify the file name when you configure the site options.

A PEA file must be created on an open systems platform, and then copied to the mainframe. (Refer to your Centera documentation for complete information on creating the file.) Use FTP to copy the file to a mainframe dataset in binary format. If you browse the PEA file with ISPF, use the DISPLAY UTF8 command, since the file is stored in UTF 8 format.

When you specify site options to configure the environment, specify the PEA file name as shown in Figure 30. Centera Pool Name Entry.

You can optionally secure the PEA file using a site-installed security package. Create a discrete profile for the PEA file, specifying universal access None (UACC (NONE)). You can then grant specific users READ access to the profile.

Double-Byte Character Set Support

When you dynamically switch to a new DB2 subsystem while working with Optim, Optim connects to the subsystem and loads a copy of DSNHDECP to obtain various DB2 subsystem settings.

The load finds the first DSNHDECP copy in the load library order, which may not belong to the current DB2 subsystem. Errors may occur when different DB2 subsystems use different or conflicting settings, for example, Coded Character Set Identifiers (CCSIDs).

The DB2 subsystem MIXED DATA option must be set to YES for DB2 Double-Byte Character Set (DBCS) support and, subsequently, Optim DBCS support. The Optim installation procedure does not explicitly specify the subtype for CHAR/VARCHAR columns during directory creation. Therefore, Optim automatically creates the CHAR/VARCHAR columns of the Optim directory with the subtype FOR MIXED DATA when MIXED DATA is set to YES.

If your DB2 subsystem has MIXED DATA set to NO and you want DBCS support, MIXED DATA must be changed to YES and the proper Multi-Byte Character Set (MBCS) CCSIDs must be specified to DB2 for the ASCII and EBCDIC encoding schemes. Changing the CCSID of a DB2 subsystem requires careful consideration. Contact IBM Support for assistance to avoid corrupting existing tables (such as the Optim directory tables).

Prior to processing VSAM, sequential, or IMS data sources that contain DBCS data, set the Optim Legacy MIXED DATA and Legacy EBCDIC CCSID site options appropriately. Refer to "Customizing the Site Options" on page 49.

Unicode Support

While the Optim plan must be bound with an EBCDIC encoding scheme, Optim is still able to process data in ASCII and Unicode DB2 tables. When Optim is reading data from an ASCII or Unicode DB2 table, DB2 will convert characters from the table's encoding scheme to the target Optim EBCDIC encoding scheme. When Optim is writing data to an ASCII or Unicode DB2 table, DB2 will convert characters from the Optim EBCDIC encoding scheme to the target table's encoding scheme.

Characters that cannot be represented in the target encoding scheme will not be preserved.

Installation Overview

The installation process involves preparing for and performing the installation.

To perform the installation, you must have DB2 SYSADM authority and update authority for the load and skeleton libraries. You should also be familiar with site naming convention standards. For remote subsystem access, you must have update authority for the table library.

If you ordered Optim using the Custom-Built Product Delivery Offering (CBPDO), follow the steps in the *IBM Optim Program Directory* for installation. (If Optim is delivered through a ServerPac, the installation is completed for you.)

Optim contains the following libraries:

hlq.SFOPCLST
　　Installation, customization, and production CLISTs

hlq.SFOPINST
　　Installation and customization jobs

hlq.SFOPLLIB
　　Load modules

hlq.SFOPMLIB
　　ISPF messages

hlq.SFOPPLIB
　　ISPF panels

hlq.SFOPSLIB
　　ISPF skeletons

hlq.SFOPTLIB
　　ISPF tables

hlq.SFOPSAMP
　　Optim sample library

Update CLISTs

After installing Optim, modify CLIST FOPCUST in hlq.SFOPCLST to establish the execution environment.

This CLIST initiates an ISPF dialog, which prompts you for the information to perform the following tasks:
* Bind a temporary plan for product installation.
* Specify a storage group, database, and table space(s) for the Optim Directory tables and indexes.
* Create DB2 tables and indexes used by Optim.
* Bind the packages and plan for Optim product execution, and grant user execute authority.

Note: Optim requires synonyms for the DB2 Catalog and Optim Directory tables, rather than explicit references to the DB2 Catalog and Optim Directory tables. You must create the synonyms before you bind the Optim packages and plan whether or not you wish to run against a shadow catalog (that is, an image of the DB2 Catalog and Optim Directory tables).
* Populate the Optim Directory with sample object definitions to set up the sample database.
* Enable Optim for trial or permanent use.
* Customize site options.
* Customize batch processing for the Archive, Move, or Compare components of Optim.

For further information about these modifications see Chapter 2, "Set Up and Modify the Optim CLIST," on page 7.

Remote Subsystem Installation Overview

To access remote subsystems, Optim must be installed and licensed on each DB2 subsystem in which it is to run.

The installation must be performed locally (that is, while directly connected to the subsystem). For remote subsystem access, both the local and the remote subsystems must be running, using the same release of Optim. Also, you must have a bi-directional DB2 remote connection between the subsystems. For example, use the IBM Distributed Data Facility (DDF) or the Distributed Relational Database Architecture™ (DRDA®).

Remote subsystem access requires the full installation process on both the local and remote subsystems, and supplemental activities on the local subsystem. The local subsystem is the subsystem on the same z/OS system as the TSO user. A remote subsystem is a subsystem that resides across a DB2 remote connection from the local subsystem.

Local Subsystem

Before you can install Optim on remote subsystems, you must install Optim on the local subsystem from which you will access the remote subsystems.

During the full installation on the local subsystem:

- Specify the prefix for the table library member that lists available plans for each remote subsystem

  Optim maintains a table library member for each remote subsystem in the table library (ISPTLIB). The install process updates these members and Optim uses them when connecting to remote subsystems. In the FOPCUST CLIST, specify a one- to four-character prefix in the SET RTTBLPFX statement. This prefix is concatenated with the remote subsystem ID to create the name of the table library member used to store the list of available plans for that remote subsystem.

- Create a plan to execute Optim on each remote subsystem

  Create a plan to access each remote subsystem. Perform this step during the final bind of the install process while connected to the local subsystem.

- Establish FOPCSSN to vary the DB2 Load libraries between subsystems

  Copy the FOPCSSN member from the sample library, SFOPSAMP, into the FUNC (SSN) portion of the FOPCUST CLIST. FOPCSSN contains the conditional logic for connecting to remote subsystems. This conditional logic allows you to concatenate the DB2 load, exit, and user libraries for the remote subsystem to the load library (SFOPLLIB) dataset.

  Note: The DB2 library concatenations made in FOPCSSN enable Optim to load the correct DB2 options module when connecting to a subsystem. The DB2 options module (DSNHDECP) provides the DB2 release number and the default DB2 settings for a subsystem. When a remote subsystem resides on a separate z/OS system, the remote DB2 libraries may not be available on the local z/OS system. If so, maintain a copy of the remote DB2 options module in a local library and specify this library in FOPCSSN when concatenating DB2 libraries for access to the remote subsystem.

- Define the remote connection

  Specify the connection to a remote DB2 location in Remote DB2 Connection on the Site Options panel. This setting determines whether the remote connection is application-directed (in other words, DRDA) or system-directed (in other words, DDF).

Optim Location

After you install Optim to the local subsystem, connect to a remote subsystem by specifying the Optim Location of the remote subsystem.

To specify the Optim Location of the remote subsystem, use the **LOCATION** prompt on the main menu. The Optim Location is the site-specified name assigned to the DB2 Location.

Perform an Upgrade

An ISPF dialog prompts you through the steps required when upgrading to a new release or installing any Optim components not already installed.

For more information on upgrading your system, see section 5. Upgrade to a New Release.

Deciding whether to install or upgrade

If you have an existing installation of Optim, you can either perform an upgrade or a full install.

Consider the following when determining whether to perform either an upgrade or a full install for subsequent releases of Optim.
* Perform a full install of release 11.3 if you are currently running a release earlier than v7.1 of Optim.
* Perform a full install to prevent the new release of Optim from replacing previously installed releases in the same subsystem. This allows you to run multiple releases of Optim from one location. To run multiple releases, you must
   – Store the CLISTs for each release in a separate library.
   – Specify a unique plan name for each release.
   – Use a unique plan qualifier to create the synonyms.
   – Specify a unique Creator ID for the Optim Directory tables.

   Once you are satisfied with the testing of the new release, you can use the Export facility to export object definitions from a previous release and use the Import facility to import these object definitions to the new release.
* Perform an upgrade if you are currently running release 7.1 or later of Optim.
* Perform an upgrade if you have tested the new release in a separate environment or simply want to upgrade the existing version of the software. The upgrade updates the software, as well as the Optim Directory tables.

Administration Tasks

To facilitate the administration of Optim, you can perform these tasks.

Optim administration tasks include the following:
* Maintain the Optim Directory and sample database.
* Enable or disable Optim.
* Bind or rebind the packages and plan.
* Create, modify or upgrade site defaults modules.
* Integrate Optim into your environment.
* Establish and customize functional, object, and Archive security.
* Use SQL DDL to populate the Directory with primary and foreign key definitions produced by third party data dictionary tools.
* Establish an interface between Optim and the BMC Catalog Manager, if used at your site.
* Convert an Extract File to an Archive File.
* Customize aging rules, date format tables, and user exits for data aging.

APF-authorized

Note that if your site intends to use an unload program, the Optim load library and any column map exit load library must be APF-authorized. If you will be using IMS data, add the Optim SFOPLLIB data set to the IMS Environment Definition IMS Program Libraries list.

Chapter 2. Set Up and Modify the Optim CLIST

After you install Optim, the next step is to set up the CLISTs.

The Optim CLIST, FOPCUST, must be modified before use. SFOPCLST, the Optim CLIST library, must be made accessible using ISPF with the ALTLIB command or copied to a data set allocated to SYSPROC.

Modify FOPCUST

Modify FOPCUST to specify the default plan name, the correct library names, and the prefix for the remote table name (if remote subsystems are to be accessed). You can also modify FOPCUST to allow multiple site options modules at your site and to allow the plan name and/or DB2 Load libraries to vary between DB2 subsystems or locations at your site.

The modifications that you can make to FOPCUST are as follows:

- Modify Plan Name

  FOP*nnn* is the default plan name, where *nnn* is the release number. To use a plan name other than FOP*nnn*, modify:

  ```
  SET DEFPLAN = plannplan
  ```

  where *planname* is the desired plan name.

- Library Names

  Update the QUAL1 and QUAL2 definitions to match the first two qualifiers of your Optim installation, if you did not use the default values. Note: The DSN prefix for your Optim installation must not exceed a length of 19 bytes.

  ```
  SET QUAL1 = high level DSN qualifier
  SET QUAL2 = middle level DSN qualifier
  ```

- DB2 Load Libraries

  If the DB2 Load Libraries are not in the logon procedure, the system linklist, or in the link pack area, modify:

  ```
  SET DB2XLIB          =   &STR() /* THE DB2 EXIT LOAD LIBRARIES
  SET DB2LLIB          =   &STR() /* THE DB2 LOAD LIBRARIES
  SET DB2ULIB          =   &STR() /* THE DB2 USER LIBRARIES
  ```

  as follows:

  ```
  SET DB2XLIB          = db2exitlibname
  SET DB2LLIB          = db2loadlibrary
  SET DB2ULIB          = db2userlibrary
  ```

  where *db2exitlibname* is the name of the DB2 Exit library, *db2loadlibrary* is the name of the DB2 Load library, and *db2userlibrary* is the name of the DB2 User library.

- Avoid LIBDEF calls

  If you define any Optim libraries in your logon procedure, you can avoid the overhead of unnecessary LIBDEF calls by setting the appropriate low-level qualifier fields in FOPCUST to blank. For example, if you define all the libraries in your logon procedure, modify the following statements as shown:

  ```
  SET ISPLLIB          =   &STR( )
  SET ISPPLIB          =   &STR( )
  SET ISPMLIB          =   &STR( )
  SET ISPSLIB          =   &STR( )
  SET ISPTLIB          =   &STR( )
  ```

  If you do not define the load library in your logon procedure, but add Optim to a menu or application (such as a DB2 products menu) that uses LIBDEF for the necessary libraries, modify the following statement as shown:

```
SET DEFLLIB           =   N
```

If you modify any of the SET ISP*xxxx* statements or the SET DEFLLIB statement as described previously, you must also modify the following statement in FOPCUST in order to be able to update the site defaults module, FOPMDFLT:

```
SET DFLTLIB = dsname
```

where *dsname* is the name of the dataset containing the site defaults module. This is usually the load library for Optim.

- Remote Subsystems

  To use a remote DB2 connection (for example, DDF or DRDA), Optim maintains a partitioned dataset (Table Library DSN) that contains a member for each remote subsystem. Each member contains the list of available plans for that subsystem.

  The name of each member is generated by prefixing the remote subsystem ID with a character string. The string is specified in FOPCUST:

  ```
  SET RTTBLPFX          =    remoteprefix
  ```

  where *remoteprefix* is a one- to four-character string. Leave *remoteprefix* blank if remote access is not available.

- Multiple Plans or DB2 Load Libraries Between DB2 Subsystems or Locations

  Your site may want to run Optim in more than one DB2 subsystem or location. If so, and any of the following conditions apply, you must modify FOPCUST.

  – Each DB2 subsystem installation requires a unique plan name.

    FOPCUST allows you to specify one default plan name (in the SET DEFPLAN statement). However, if your site wants to run multiple installations of Optim in the same DB2 subsystem, you must specify a unique plan name for each installation to avoid overlaying an existing plan.

  – The DB2 subsystems or locations require different DB2 Load libraries.

    For example, if you want to run Optim in separate z/OS systems using different releases of DB2, specify the appropriate DB2 Load libraries for each subsystem or location.

    Near the end of FOPCUST, after the WHEN (SSN) function call, you are instructed to copy the FOPCSSN member from the sample library (SFOPSAMP) into the FUNC (SSN) portion of FOPCUST. FUNC (SSN) is called before Optim attempts to connect to a DB2 subsystem (named in the PARM variable) or location (named in the DB2LOC variable). Modify the sample code in FOPCSSN as follows:

  – To use the appropriate plan for the DB2 subsystem or location, set the DB2PLAN variable to the plan name for the DB2 subsystem or location.

  – To use the appropriate DB2 Load libraries for the DB2 subsystem or location, set the DB2XLIB, DB2LLIB, and DB2ULIB variables to the DSNEXIT, DSNLOAD, and DSNULIB dataset names for the DB2 subsystem or location.

    The additional CLIST code and the skeleton logic near the end of FOPCSSN ensure that these variables are LIBDEF'd for online use and are included in the JCL generated to perform DB2 utilities.

- Multiple Site Defaults Modules

  Optim includes a site defaults module named FOPMDFLT that you can update or use as a model to create customized site defaults modules. After installing Optim, you may specify a unique two-character suffix for the new site defaults module. (See Chapter 9, "Create or Modify Site Defaults Modules," on page 115 for further information on implementing multiple variations of the site defaults module.)

  To use variations of the site defaults module and assign them by subsystem, location, and/or SQLID, you must modify FOPCUST. Near the end of FOPCUST, in the WHEN (SFX) function call, you are instructed to copy the FOPCSFX member from the sample library (SFOPSAMP) into the FUNC (SFX) portion of FOPCUST. You must also modify the sample code in FOPCSFX to set the DFLTSFX variable according to your requirements. In the following statement in FOPCSFX, indicate the two-character suffix for the new defaults module (FOPMDF*xx*).

```
SET DFLTSFX          =  xx
```
The new site defaults module resides in the load library with the distributed site defaults module, FOPMDFLT.

Make CLISTs Accessible

You must also make the CLISTs accessible from an ISPF EDIT Session (ISPF Option 2).

There are two methods to make the CLISTs accessible.

1. Use a CLIST library allocated at logon to SYSPROC.

   Copy all members from the SFOPCLST library into the SYSPROC library. If you are installing Access, also copy those with names that begin with ADB2.

2. Use the CLIST library identified with the TSO ALTLIB command:

   ```
   TSO ALTLIB ACTIVATE APPL(CLIST) DSN('hlq.slq.SFOPCLST')
   ```

   `'hlq.slq.SFOPCLST'` is the name of the CLIST library.

   Copy the FOPINSTL CLIST into a SYSPROC library. If installing Access, also copy the FOPD2EXE, FOPFAD2P, FOPD2EXP, and FOPD2XSQ CLISTs into the same library.

   You must edit FOPINSTL and FOPD2XSQ CLISTs. Locate the necessary ALTLIB commands (commented in the CLISTs), remove the comment characters, and replace the dummy dataset names with the name of the CLIST library for Optim.

Note: To run multiple releases of Optim, you must store the CLISTs for each release in a separate library.

Chapter 3. Installation Dialog

The CLIST FOPINSTL initiates an ISPF dialog that allows you to install, upgrade, and maintain Optim.

An option allows you to store the SQL statements and DSN subcommands generated by the current install procedure in a sequential file for use in subsequent installations. You can reuse the stored statements and subcommands to install Optim in another location, without re-executing the installation dialogs, using SPUFI to review and execute the stored SQL and DSN subcommands. (Information about allocating the output file is provided in "SQL Output Parameters" on page 14. Information about using these stored statements and subcommands is provided in Appendix A, "Reuse DSN Subcommands and SQL," on page 195.)

The Install menu is displayed when you invoke FOPINSTL.

```
------------ IBM's InfoSphere Optim Install --------------------------
OPTION  ===>

                1 - Full Install
                2 - Upgrade to New Release

                3 - Maintain Directory and Sample Database
                4 - Enable/Disable Product(s) Usage
                5 - Bind or Rebind DB2 Plans
                6 - Create/Modify/Upgrade Site Defaults Modules
                7 - Apply Optim Connect Maintenance
                C - Installation and Site Administration Changes


     DB2 Subsystem Name  ===> DSNC
     DB2 Release         ===> 11.1.0 (n.n.n)
     Plan Name           ===>
     Secondary SQLID     ===>
     Load Library DSN    ===> 'OPTUSR.REL113.SFOPLIB'
     Install Library DSN ===> 'OPTUSR.REL113.SFOPINST'
     Save SQL/DSN Output ===> N     (Y-Yes, N-No)

5655-V03 (C)  Copyright IBM Corporation 1989, 2014.
All rights reserved.  Licensed materials - property of IBM.
US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP schedule with IBM Corp.
                     Optim 11.3.0
```

Figure 1. Installation Main Menu

At the Option prompt, type the one-character identifier that corresponds to the function you wish to invoke.

Optim Install Options

The available installation options available from this menu are:

1 Full Install
> Install Optim. Use this option if you have not previously installed any of the Optim products or if you are upgrading from release 6.1.1 or earlier.

2 Upgrade to New Release
> Upgrade to a new release of Optim. Use this option to upgrade from release 7.1.0 or later of Optim, or to add components not previously installed.

3 Maintain Directory and Sample Database

Some of the following options are included for Option 1 Full Install and Option 2 Upgrade to New Release, but are also provided here so that they can be re-executed after installation or upgrade.

- Create the sample database and load the accompanying data distributed with Optim. (For a detailed description of the sample database tables, see Appendix A. Sample Database Tables and Structure in the *Common Elements Manual*.)
- Drop the sample tables to completely refresh the sample database.
- Load the sample Access Definitions and Column Maps distributed with Optim.
- Copy the DB2 relationships into the Optim Directory, which is a set of DB2 tables that serve as an extension to the DB2 Catalog. To improve performance when accessing the database, Optim maintains copies of DB2 relationships in the Directory.
- Create Optim primary keys for DB2 tables with unique indexes that do not have a DB2-defined primary key.
- Create sample Optim Legacy VSAM files.
- Load sample Optim Legacy VSAM data.
- Delete the sample Optim Legacy VSAM data.
- Create the sample Optim Legacy IMS™ files.
- Load the sample Optim Legacy IMS files and objects.
- Drop the sample Optim Legacy IMS files.

4 Enable/Disable Product(s) Usage

Enable or disable any Optim component for evaluation or permanent license. The CPU ID is displayed on this panel. (This option is also available from Option 1 Full Install and Option 2 Upgrade to New Release.)

5 Bind or Rebind DB2 Plans

Bind or rebind existing packages and plans, as needed. (This option is also available from Option 1 Full Install and Option 2 Upgrade to New Release.)

6 Create/Modify/Upgrade Site Defaults Modules

Create or modify the Site Options. You can model individualized site defaults modules on the basis of an existing site defaults module.

7 Apply Optim Connect Maintenance

Unload maintenance for Optim Connect. You will have the option to specify unit and volume for the data sets.

C Installation and Site Administration Changes

Display documentation about changes to the installation and site administration functions for the current release of Optim.

Optim Install Parameters

Provide values for the following parameters:

DB2 Subsystem Name

The DB2 subsystem into which Optim is installed. Optim must be installed in each subsystem in which it is used.

DB2 Release

The number of the DB2 Release for which the plan is to be created. Specify the number of the DB2 subsystem where you are installingOptim.

The release you enter is checked against the DB2 subsystem. If there is a discrepancy, you are prompted to confirm the release number. (See "Verify DB2 Release" on page 13.)

Plan Name

Secondary SQLID
> Optional name of a secondary SQLID to be used for the install process. If **Secondary SQLID** is blank, your primary SQLID is used. The SQLID used for the installation must have SYSADM authority.

Load Library DSN
> Fully qualified name of the load library (SFOPLLIB) for Optim. This library was created and populated as part of the SMP/E install process.

Install Library DSN
> Fully qualified name of the install dataset (SFOPINST).

Save SQL/DSN Output
> Indicator for storing the SQL statements and DSN subcommands generated by the current install procedure. Specify:
>
> **Y**   Store the SQL and DSN output. You are prompted for a destination of SYSOUT or a sequential dataset. (See "SQL Output Parameters" on page 14.)
>
> **N**   Do not store the SQL and DSN output.

To install Optim, your TSO session must be able to connect to the DB2 subsystem you have specified.

Note: Only one copy of the Optim libraries is needed, regardless of the number of subsystems into which you are installing.

Verify DB2 Release

When the DB2 release entry at the **DB2 Release** prompt on the Install menu does not match the DB2 subsystem, you must confirm the release number.

The Verify DB2 Release panel prompts you to confirm the release number.

```
----------------------------- Verify DB2 Release --------------------------
Command ===>                                                    DSNC



         DB2 Release determined by Optim using SESSION VAR : 10.1.0
         DB2 Release selected previously by user           : 9.1.5
         Please confirm the correct release to use     ===>


********************************* N O T E S *********************************
*       Optim has determined that the target DB2 subsystem is at        *
*    a different release from the one specified on a previous panel.     *
*    Different components are needed for these releases.  Therefore,     *
*    specify the DB2 release the target subsystem is running.            *
*                                                                        *
*    The DB2 release is displayed in format vv.r.m where:                *
*      vv = DB2 version identifier (such as 10)                          *
*      r  = DB2 release identifier (such as 1)                           *
*      m  = DB2 maintenance level (values 0-4 indicate compatibility and *
*           enabling-new-function-mode, values 5-9 indicate new-function *
*           mode).                                                       *
****************************************************************************
```

Enter the correct release number at the confirmation prompt. Verify that the DB2 release number you enter is the correct release for the DB2 subsystem where you are installing Optim. Optim performs the installation as appropriate for the target DB2 subsystem.

SQL Output Parameters

If you enter Yes for **Save SQL/DSN Output** on the Install menu, the SQL Output Parameters panel prompts for the output parameters.

```
-------------------------- SQL Output Parameters -------------------------
COMMAND ===>

 SQL Output Parameters:
   Output Type      ===> S          D-Dataset, S-SYSOUT

   If Dataset: DSN  ===>
               Disposition   ===>    M-MOD, O-OLD

   If SYSOUT:  SYSOUT Class  ===> *   A - Z, 0 - 9, *
               Destination   ===>
               Hold          ===> N   Y-Yes, N-No

Press ENTER key to Allocate Output
Press END Command to Cancel Allocation

******************************* N O T E S ********************************
*    Many DB2 objects (e.g. tables,indexes) are created as a result of this *
*    installation.  You may want to save the generated SQL if you will be   *
*    installing Optim in multiple subsystems, or if you simply want to      *
*    document what the installation facility did.  In either case, you must *
*    specify a destination for the output on this panel.                    *
**************************************************************************
```

Figure 2. SQL Output Parameters

The SQL Output Parameters panel includes:

Output Type
> The destination for stored SQL statements and DSN subcommands generated by the current install procedure.
>
> Specify D for Dataset or S for Sysout.

If Dataset:
> Information needed if **Output Type** is D for Dataset. Specify the following:
>
> **DSN**　Name of a sequential dataset. If the dataset does not exist, you are prompted to allocate it.
>
> **Disposition**
> > Specify M or O.
> > - M means append current SQL to that in the dataset.
> > - O means save current SQL only (overwrite any information in the dataset).

If SYSOUT:
> Information needed if **Output Type** is S for Sysout. Specify:
>
> **SYSOUT Class**
> > The output class for the printed output. Specify an alphabetic or numeric character, or an asterisk (*).
>
> **Destination**
> > The SYSOUT destination. Specify a valid local or remote terminal, a node in the JES network, a local or remote printer or workstation, or a TSO User ID.
>
> **Hold**　Disposition of the output. Specify Y to hold the output until released or deleted by an operator. Specify N to not hold the output.

If the output is to an existing dataset that conforms to the dataset requirements (sequential dataset with 0 directory blocks, FB, and LRECL=80), the dataset is used. If the dataset does not conform to the requirements, a message is displayed and you must respecify the dataset name.

If the named dataset does not exist, the Allocate Dataset panel prompts you for allocation information.

```
+--------------------------Allocate Dataset--------------------------+
|                                                                    |
| Data Set Name: Z13600MP.TEST.DSN                                   |
| Management Class     ===>         (Blank for Default Management Class) |
| Storage Class        ===>         (Blank for Default Storage Class) |
|   Use Stored Values  ===> N       (Y-Yes, N-No, V-View Stored Values) |
|   Volume Serial      ===>         (Blank for Authorized Default Volume) |
|   Device Type        ===>         (Generic Unit or Device Address) |
| Data Class           ===>         (Blank for Default Data Class)   |
|   Space Units        : BLKS       (BLKS, CYLS, TRKS, KB or MB)     |
|   Primary Quantity   ===> 1       (In Above Units)                 |
|   Secondary Quantity ===> 1       (In Above Units)                 |
|   Directory Blocks   : 0                                           |
|   Record Format      : FB                                          |
|   Record Length      : 80         (Value will be set to Block Size) |
|   Block Size         ===> 3200    (Must be a multiple of Record Length) |
|   Data set name type ===>         (BASIC, LARGE, EXTREQ, EXTPREF or Blank) |
|                                                                    |
|                                                                    |
|Calculated Values Shown, Change to Profiled Values ===> N (Y-Yes N-No) |
+--------------------------------------------------------------------+
```

Figure 3. Allocate Dataset Panel

Allocate Dataset Panel

The allocate dataset panel includes these prompts:

Dataset Name
> Name of the dataset specified on the SQL Output Parameters panel. Use CANCEL to redisplay the panel and respecify the dataset name, if needed.

Management Class
> The management class used to obtain the data management-related information (migration, backup, and retention criteria) for the allocation of the dataset. Leave blank for default. **Management Class** is active only if SMS is active.

Storage Class
> The storage class used to obtain the storage-related information for the allocation of the dataset. Leave blank for default. **Storage Class** is active only if SMS is active.

Use Stored Values
> The source of volume information. Specify:

> **Y**   Use the volume information specified on the File Unit Parameters panel, accessed from User Options.

> If you select Y, but no explicit or default volume information was specified on the File Unit Parameters panel, an error message is displayed.

> **N**   Use the volume information specified on the Allocate Dataset panel.

> **V**   Display the File Unit Parameters panel to specify values for unit, volume number, and one to six volume serial numbers.

Volume Serial
> The serial number of the volume on which the dataset is to reside. This value must contain only alphanumerics, national characters, and the hyphen. Alternatively, you can leave **Volume Serial** blank to use a default volume for which you are authorized. You cannot provide both **Volume Serial** and **Device Type**.

Device Type

The generic unit or device address of the device on which the dataset is to reside. Only alphanumeric characters are valid. Alternatively, you can leave **Device Type** blank to use the default device type. You cannot provide both **Device Type** and **Volume Serial**.

Data Class

The data class used to obtain the data-related information (SPACE, LRECL, and so on) for the allocation of the dataset. Leave blank for default. **Data Class** is active only if SMS is active.

Space Units

The DASD space for the dataset to be allocated.

BLKS Blocks, abbreviated BLK or B.

TRKS Tracks, abbreviated TRK or T.

CYLS Cylinders abbreviated CYL or C.

Primary Quantity

The primary quantity of space to be allocated in the indicated space units. Site management may provide guidelines regarding appropriate values.

Secondary Quantity

The secondary quantity of space to be allocated in the indicated space units. Site management may provide guidelines.

Directory Blocks

The number of directory blocks to be allocated. This value is 0 and cannot be modified.

Record Format

The record format is FB and cannot be modified.

Record Length

The logical record length is specified as 80 and cannot be modified.

Block Size

The block size in bytes. The value must be a multiple of 80. Site management may provide guidelines.

Data set name type

The type of data set being allocated. Specify:

BASIC

The data set is not an extended or large format data set.

LARGE

Allocates a large format sequential data set.

EXTREQ

Specifies that an extended data set is required.

EXTPREF

Specifies that an extended data set is preferred.

blank Allocates a partitioned or sequential data set based on the values entered on the panel.

Calculated Values Shown, Change to Profiled Values

The values used to allocate the file. Specify:

Y Use values previously specified and profiled. Also, if you have modified the values on this panel, they are profiled and replace any previously specified values.

N Use the values calculated by Optim.

Calculated values are based on the "worst case." Any unused space is freed when the process using the dataset is complete.

Chapter 4. Full Install

Optim must be installed on each DB2 subsystem in which it is to run.

You can install Optim on each subsystem by using the installation panel, or you can store the DSN subcommands and SQL generated by your first installation and execute the stored SQL in other subsystems as desired. (For information about storing the SQL, see Chapter 3, "Installation Dialog," on page 11. For information about using the stored SQL, see Appendix B, "Review SQL Statements and DSN Subcommands," on page 197.)

Note: If you are using release of Optim earlier than v7.1, you must perform a full install to migrate to release 11.3. Sites running Optim release 7.1 or later can upgrade to release 11.3. See Chapter 5, "Upgrade to a New Release," on page 71.

Select Option 1 from the main Install menu to perform a full install.

When you select this option, the Full Installation menu is displayed to prompt you for the necessary information.

```
-----------  Full Installation of Optim -------------------------------------
OPTION  ===>

The installation consists of the following steps:

     1 - Temporary Bind of Optim Plan for Product Install
     2 - Specification of Optim Directory Storage Group
     3 - Specification of Optim Directory Database
     4 - Specification of Optim Directory Table Space(s)
     5 - Creation of Optim Directory Tables
     6 - Creation of Optim Directory Indexes
     7 - Final Bind of Optim Plan for Product Execution
     8 - Enable Optim Product(s) for Execution
     9 - Unload Optim Sample Data and ODM Optim Connect
    10 - Customize Optim Site Options
    11 - Setup Optim Directory and Sample Database
    12 - Customize the Batch Execution Environment

 DB2 Subsys for Install ===>
 DB2 Release            ===>            (n.n.n)
 Secondary SQLID        ===>
 Load Library DSN       ===> 'OPTUSR.RELnnn.SFOPLLIB'
 Install Library DSN    ===> 'OPTUSR.RELnnn.SFOPINST'
```

Figure 4. Full Installation Menu

The Full Installation menu lists the steps required to install Optim. By pressing ENTER each time this panel is displayed, you can proceed through the installation to execute the steps in the order in which they are listed. Using this method, you can suspend the installation process and restart it later without having to repeat steps. The installation process restarts at the point where you suspended it, as long as you use the TSO User ID and the DB2 subsystem used previously.

When you are installing Optim for the first time, you must execute the Temporary Bind step first. Then proceed with steps 2, 3 and 4 to create the storage group and specify the database and table space for the Optim Directory. Steps 5 and 6 create the Directory tables and indexes that contain the object definitions used by Optim. You must execute Step 7, the Final Bind, prior to performing the remaining steps. Step 8, Enable Optim for Execution, must be performed before you can use Optim. Step 9 loads the sample data and ODM Optim Connect. It is recommended that you review steps 10 and 12 to customize Optim for your site.

Note: If you are installing into multiple subsystems and sharing the product libraries, steps 8, 10, and 12 need be performed only once.

In addition to the list of steps, you are prompted for:

DB2 Subsys for Install
    DB2 subsystem into which Optim is installed.

DB2 Release
    Release of DB2 used by Optim.

Secondary SQLID
    Optional name of a secondary SQLID to be used for the install process. If blank, your primary SQLID will be used.

Load Library DSN
    Fully qualified name of the load library (SFOPLLIB).

Install Library DSN
    Fully qualified name of the installation library (SFOPINST).

The values entered on the Install main menu are displayed initially and can be modified from this panel.

The following discussion describes each step of the install process in the order in which they are presented on the Full Installation menu.

Bind the Plan for Product Installation

A bind is performed twice during the installation process.

The first step in the installation process binds a temporary plan. The temporary plan enables execution of the SQL generated in subsequent steps to create the tables needed to execute the solution. (See "Bind for Product Execution" on page 30 for a description of the second bind process that enables execution of the solution.) The following panel prompts for the information used to bind a temporary plan.

```
----------------------- Bind Plan for Product Install ---------------------
COMMAND ===>

Plan Owner              ===>
Plan Name               ===>
DBRM Library DSN        ===>

Review DSN subcommand prior to execution ===> NO  (Y-Yes, N-No)

Enter Options:
  Bind Action               ===> ADD       (Add, Replace)
  Retain Execution Authority ===>          (Y-Yes, N-No)
  Isolation Level           ===> CS        (CS, RR)
  Validate                  ===> BIND      (Run, Bind)
  Resource Acquisition Time ===> USE       (Use, Allocate)
  Resource Release Time     ===> COMMIT    (Commit, Deallocate)
  Explain Plan Selection    ===> NO        (Y-Yes, N-No)
  Bind FOPEMSGS Level       ===> ALL       (All, Warning, Error, Complete)
  Data Currency             ===> YES       (Y-Yes, N-No)


******************************** N O T E S ********************************
*       The plan owner must have BINDADD privilege or SYSADM authority    *
*       to complete this function successfully.                           *
**************************************************************************
```

Figure 5. Bind Plan - Install

The panel includes the following prompts (defaults are provided for all required prompts):

Plan Owner

The plan owner. Enter the name of an owner with INDADD privileges or SYSADM authority to perform the bind. Leave blank to use the SQLID being used for the install; either the primary SQLID or the secondary SQLID for the user, if one was entered on the Full Installation menu.

Plan Name

The name of the plan specified in FOPCUST. If you enter a different plan name, you must modify FOPCUST to reflect the new plan name before attempting to execute the solution. (See Chapter 2, "Set Up and Modify the Optim CLIST," on page 7 for information on modifying FOPCUST.)

DBRM Library DSN

The name of the library containing the DBRMs. Unless you have moved the DBRMs, enter the name of the install library.

Review DSN subcommand

Setting for reviewing the DSN subcommand that is generated in response to the prompts before it is executed. This value is profiled. (See Appendix B, "Review SQL Statements and DSN Subcommands," on page 197 for details.) Specify:

Y   Review the DSN subcommand prior to execution.

N   Execute the DSN subcommand without reviewing it.

Bind Action

The bind action. Specify:

Add   Add a new plan. Used for the initial bind for product installation.

Replace

Overlay an existing plan of the same name with a new plan. Used for the final bind for product execution.

Retain Execution Authority

When the **Bind Action** is Replace, use this option to overlay an existing plan of the same name with a new plan. Specify:

Y   Retain BIND and EXECUTE authority.

N   Do not retain BIND and EXECUTE authority.

Isolation Level

Indicator for the effect of executing one application bound to this plan on the other applications accessing the DB2 subsystem. Specify:

CS   For Cursor Stability. Read database values are protected only while in use and changed values are protected until committed. This setting is recommended.

RR   For Repeatable Read. Read or changed database values are protected and cannot be changed by other applications until Optim commits or terminates.

Validate

Indicator for validity checking. Specify:

Run   Defer checking until plan is executed.

Bind   Check validity during bind processing. This setting is recommended to speed catalog access and reduce authorization requirements.

Resource Acquisition Time

Indicator for acquiring resources. Specify:

Use   Open table spaces and acquire locks when the plan first uses them. This setting is recommended.

Allocate

Open all table spaces and acquire all locks when the plan is allocated. If you use Allocate, you must use Deallocate for **Resource Release Time**.

Resource Release Time

Indicator for releasing resources. Specify:

Commit

Release resources at each commit point.

Deallocate

Release resources when the application terminates.

Explain Plan Selection

Setting for information about the execution of SQL statements in the PLAN.

Y   EXPLAIN information is provided.

N   EXPLAIN information is not provided.

Bind FOPEMSGS Level

Indicator for the severity of messages to be displayed. Specify:

All   All messages.

Warning

Warning, error, and completion messages.

Error   Only error and completion messages.

Complete

Completion messages.

Data Currency

Indicator for data currency for read-only and ambiguous cursors when the isolation level of cursor stability is in effect.

Y   Data currency is required. Default.

N   Data currency is not required.

Specify Storage Group for Directory Tables

The second option on the Full Installation menu prompts for the storage group for the Directory tables.

```
------------   Specify Storage Group for Directory Tables --------------------
OPTION  ===>

    Select type of storage group allocation desired:

              1 - Create new storage group
              2 - Use existing storage group
              3 - Use previously defined VCAT
              X - Bypass storage group specification

Storage Group Name ===>            (Options 1 or 2)
VCAT Name          ===>            (Options 1 or 3)

For OPTION 1:
         Volumes  ===>        ===>        ===>        ===>        ===>
         Password ===>

 Review SQL prior to execution ===> YES (Y-Yes N-No)

******************************** N O T E S ********************************
*                You must have CREATESG or SYSADM authority              *
*                to complete this function successfully.                 *
**************************************************************************
```

Specify the type of storage group allocations as:

1   Create a new storage group. Enter a new storage group name, a VCAT name, and one or more explicit volumes.

2   Use an existing storage group. Enter the name of an existing storage group.

3   Use a previously defined VCAT. Enter an explicit VSAM catalog name.

X   Bypass storage group specification.

In addition to selecting an option, specify the following:

Storage Group Name
Name of the storage group to be used for Option 1 or 2. For Option 1, you may not name a storage group already defined in the DB2 Catalog.

VCAT Name
Name or alias of the VCAT catalog for Option 1 or 3. Use the alias if the name of the VCAT is longer than 8 characters.

For Option 1, you must specify both a storage group name and a VCAT name. For Option 2, you must specify a storage group name. For Option 3, you must specify a VCAT name. You may select Option X to bypass this function.

Specify the following information for Option 1 only:

Volumes
Volumes on which the storage group is to be placed. Specify at least one volume and maximum of five volumes. Use the tab key to position to successive **Volume** prompts on the line. If you are using DFSMS for storage management, you may enter an asterisk delimited by single quotes '*' to direct DFSMS to manage the storage needed.

Review SQL

You can review the generated SQL before it is executed by entering Y at the **Review SQL prior to execution** prompt. To execute the SQL without review, enter N.

After you display the SQL, press ENTER to execute it. Alternatively, use END to modify the SQL. The Specify Storage Group for Directory Tables panel is redisplayed. You can modify any entries, redisplay the generated SQL, and execute it.

The **Review SQL prior to execution** prompt is included on every panel that generates SQL statements. The value entered here is retained and used throughout the installation until you change it.

For more information about the **Review SQL prior to execution** prompt, see Appendix B, "Review SQL Statements and DSN Subcommands," on page 197.

Specify Database for Directory Tables

The third option on the Full Installation menu prompts for information regarding the database for the Directory tables.

```
---------------- Specify Database for Directory Tables ----------------------
OPTION  ===>

 Select type of database allocation desired:

            1 - Create new database
            2 - Use existing database
            X - Bypass database specification


Database Name        ===>

For OPTION 1:
  Storage Group Name ===>
  Buffer Pool        ===>              (BP0-BP49)
  Index Buffer Pool  ===>              (BP0-BP49)

Review SQL prior to execution ===> NO  (Y-Yes, N-No)

****************************** N O T E S ********************************
*         You must have CREATEDBA, CREATEDBC, or SYSADM authority       *
*         in order to complete this function successfully.              *
************************************************************************
```

Select the type of database allocation:

1   Create a new database. You must provide the database name, as well as the storage group name, the buffer pool, and the index buffer pool.

2   Use an existing database. You must provide the name.

X   Bypass database specification.

In addition to selecting an option, supply the information for the individual prompts:

Database Name
Name of the database. If you select Option 1 the database cannot be defined in the DB2 Catalog.

For Option 1 only:

Storage Group Name
Name of a storage group as specified on the Specify Storage Group for Directory Tables panel. (See "Specify Storage Group for Directory Tables" on page 20.) If you did not enter a name on that panel or bypassed that step, you may enter the name of an existing storage group at this prompt. If you leave **Storage Group Name** blank and did not provide a name on the Specify Storage Group for Directory Tables panel, the DB2 default, SYSDEFLT, is used.

Buffer Pool
Name of the default buffer pool for table spaces within the database. The panel shows the valid selections (BP0-BP49). The default value is BP0.

Index Buffer Pool
Name of the default buffer pool for indexes within the database. The panel shows the valid selections (BP0-BP49). The default is the database buffer pool.

The **Review SQL prior to execution** prompt on this panel allows you to review the generated SQL prior to its execution and to re-specify parameters, as necessary. For more information about this prompt, see Appendix B, "Review SQL Statements and DSN Subcommands," on page 197.

Specify Table Space for Directory Tables

The fourth option on the Full Installation menu prompts for the table space allocation for the Directory tables.

```
----------- Specify Table Space(s) for Directory Tables --------------------
OPTION  ===>

Select type of table space allocation desired:

  1 - Create new table space(s)
  2 - Use existing table space(s)
  X - Bypass table space creation

      Entries left blank will use the ADDEF table space.

                ADDEF        PKDEF        RELDEF    CMDEF     ARCHDEF
  Table Space  ===>

                ADB2AUDIT   MDB2PROCESS   TMDEF     CDDEF     ARCHIDX
  Table Space  ===>

                LTDEF        ARCHLOG      ENVDEF    RDDEF     AFCOLLCT
  Table Space  ===>

                CPMETER      CMPROC
  Table Space  ===>

If the audit facility will be used,ADB2AUDIT should be placed in a separate
tablespace. ADB2AUDIT and ARCHIDX must reside in a 8K tablespace.
```

Select the type of table space allocation:

1   Create one or more new table spaces. You are prompted for the information needed to create each specified table space. Do not define a table space with the IMPLICIT attribute.

To use a single table space for all tables, specify a table space name for ADDEF only in an 8K bufferpool. If you intend to use the Access audit facility, a separate table space in an 8K bufferpool for ADB2AUDIT is advised; this table may require more maintenance than other tables in the Directory.

2   Use the existing table spaces identified in **Table Space**. Note that the ADB2AUDIT and ARCHIDX directory tables require table space in an 8K buffer pool. The remaining directory tables can use table space in a 4K or 8K buffer pool. Do not use a table space that was defined with the IMPLICIT attribute.

X   Bypass table space creation. The individual table spaces are created implicitly using the "IN DATABASE" clause on the CREATE TABLE statement.

Segmentation

When more than one table is to be in a single table space, the table space must be segmented. When a table space is to contain a single table, segmentation is optional.

Create Table Space

The Create Directory Tables Table Space panel is displayed once for each table space. Entries are profiled and presented when the panel is displayed for the next table space. The title of the panel includes the name of the table that will be in that table space. When all directory tables are in a single table space, the panel is displayed once, and the title of the panel is as shown.

```
------------------ Create ADDEF Table Space -------------------------------
Command ===>

Table Space Name      ===>           Database Name ===>
Storage Group Name    ===>                  PRIQTY ===> 2000
       OR                                   SECQTY ===> 1000
VCAT Name             ===>                  ERASE  ===> YES (Y-Yes, N-No)

FreeBlock:  FREEPAGE  ===>    (0-255)     PCTFREE ===>      (0-99)

Buffer Pool           ===>         (BP8K0-BP8K9)
Close                 ===> NO      (Y-Yes, N-No)
Segment Size          ===> 4       (multiple of 4 between 4 and 64)
Lock Size             ===>         (ANY, PAGE, ROW, TABLESPACE, TABLE)
LockMax               ===>         (0 - 2147483647, SYSTEM)
MaxRows               ===>         (1 - 255)

Review SQL prior to execution ===> NO  (Y-Yes, N-No)

****************************** N O T E S *********************************
*   You need CREATETS, DBADM, DBCTRL, DBMAINT authority for the database   *
*   or SYSADM authority in order to complete this function successfully.   *
*************************************************************************
```

Figure 6. Create Directory Tables Table Space - Install

The Create Directory Tables Table Space panel includes:

Table Space Name
> Name of the table space to be used. The name specified on the previous panel is displayed and can be modified on this panel. This is required.

Database Name
> Name of the database provided on the Specify Database for Directory Tables panel. This value can be modified on this panel.

Storage Group Name
> Name of the storage group as provided on the Specify Storage Group for Directory Tables panel. The named storage group must be described in the catalog. You can specify a storage group name or a VCAT name, not both.

VCAT Name
> Name of the VCAT catalog as provided on the Specify Storage Group for Directory Tables panel. If specified, the named VCAT must be defined in the ICF Catalog.

PRIQTY
> Primary space allocation. Optional. Default is 2000.

SECQTY
> Secondary space allocation. Optional. Default is 1000.

ERASE
> Indicator for filling the DB2-defined datasets with zeroes when the table space is dropped. Optional. Specify:

> **Y**   Erase datasets. Default.

> **N**   Do not erase datasets.

FreeBlock
> Indicator of how often to leave a page of free space when the table space is loaded or reorganized. Supply a value from 0 through 255. Optional.

PCTFREE
> Percentage of a page to be used as free space when the table space is loaded or reorganized. Supply a value from 0 through 99. Optional.

Buffer Pool

The names of the buffer pools. Optional. The panel shows the valid selections.

Note: The ADB2AUDIT and ARCHIDX directory tables table space require 8K buffer pools.

Close Indicator for opening files once per session or once per logical unit of work. Optional. Specify:

Y   Files are to be opened once per logical unit of work and then closed.

N   Files are to be opened once per session and closed when the session terminates. Default.

Segment Size

The segment size when a segmented table space is defined for the Directory tables. This is required only when a table space contains multiple tables or `TABLE` locking level is requested.

When multiple tables are in the same table space, this value defaults to 4.

Lock Size

The locking level. Optional. Specify:

ANY   Any locking level, usually page.

PAGE   Page locking level.

TABLESPACE
          Table space locking level.

TABLE   Table locking level.

ROW   Row locking level.

If **Lock Size** is `TABLE`, then **Segment Size** is required and **LockMax** must be 0.

LockMax

The maximum number of rows that can be locked.

MaxRows

The maximum number of rows that can be placed into a table space.

The **Review SQL prior to execution** prompt on this panel allows you to review the SQL generated to create each table space and to re-specify parameters, as necessary. For more information, see Appendix B, "Review SQL Statements and DSN Subcommands," on page 197.

Create the Optim Directory Tables

The fifth option on the Full Installation menu prompts for the information needed to create the tables that hold object definitions used by Optim.

Although the base table names must be ADDEF, CDDEF, PKDEF, RELDEF, TMDEF, CMDEF, ARCHDEF, ARCHIDX, ADB2AUDIT, MDB2PROCESS, LTDEF, ARCHLOG, ENVDEF, RDDEF, AFCOLLCT CPMETER and CMPROC, any user-specified name can be supplied to replace the default SYSFOP Authorization ID. The same user-specified name must be used for all tables.

Note: Both the CMDEF and CMPROC tables contain a LOB column. During installation and upgrade processing, Optim will set the CURRENT RULES special register to STD, allowing DB2 to implicitly create the LOB table space, auxiliary table, and index on the auxiliary table for each LOB column in the base table.

Access Definition Table

This table contains Access Definitions created while using Optim. Some sample Access Definitions are distributed with the products and are loaded into this table through Option 10 (Setup Directory and Sample Database). This table is required to store Access Definitions. The default name is SYSFOP.ADDEF.

Compare Definition Table

This table contains any Compare Definitions created while using Compare. This table is required even if you do not install or use Compare. The default name is SYSFOP.CDDEF.

Primary Key Table

This table stores the primary keys defined through Optim. This table is required. The default name is SYSFOP.PKDEF.

Relationship Table

This table stores relationships defined through Optim as well as shadow copies of the DB2 relationships for faster access. This table is required. The default name is SYSFOP.RELDEF.

Table Map Table

This table stores the Table Maps defined using Optim. This table is required. The default name is SYSFOP.TMDEF.

Column Map Table

This table stores the Column Maps defined using Optim. This table is required. The default name is SYSFOP.CMDEF.

Archive Directory Table

This table stores information about the Archive Files created while using Archive. This table is required even if you do not intend to install Archive. The default name is SYSFOP.ARCHDEF.

Archive Index Table

This table stores the index and content information for Archive Files created while using Archive. This table is required even if you do not intend to install Archive. The default name is SYSFOP.ARCHIDX.

Audit Table

This table stores the audit information accumulated when the Access Audit Facility is activated. This table is required even if you do not currently intend to activate the Audit Facility or Access. The default name is SYSFOP.ADB2AUDIT.

Note: If the Audit Facility is activated, this table has the potential to become quite large. It is recommended that you maintain the table by periodically executing SQL, such as the following:

```
DELETE FROM SYSFOP.ADB2AUDIT
WHERE TIMESTAMP < CURRENT TIMESTAMP - 7 DAYS;
```

This sample SQL is included in the sample library in the member FOP9SQL2.

If you would rather not maintain the audit table, disable the Audit Facility by specifying NEVER for the **Audit Mode** setting on the Site Options panel (see Figure 17 on page 49).

If the table resides in a separate table space, it can be more easily tailored to site requirements. For this reason, IBM recommends a separate table space for ADB2AUDIT if you intend to use the Access Audit Facility.

Process Table

This table stores the information generated for the last execution of a process that can be restarted or retried. This table is required. The default name is SYSFOP.MDB2PROCESS.

Optim Legacy Table

This table contains Legacy Table information for Optim Legacy. This table is required even if you do not intend to install Optim Legacy. The default name is SYSFOP.LTDEF.

Archive Log Table

This table stores information from the Archive Logging facility. It is required even if you do not intend to use Archive Logging. The default name is SYSFOP.ARCHLOG.

IMS Environment Definition Table

This table contains the overall specifications for the IMS environment. This table is required even if you do not currently intend to install Optim Legacy. The default name is SYSFOP.ENVDEF.

Archive File Collections Table

This table contains information for Archive File Collections. This table is required even if you do not currently intend to use Archive File Collections. The default name is SYSFOP.AFCOLLCT.

IMS Retrieval Definition Table

This table contains the information needed to access IMS data. This table is required even if you do not currently intend to install Optim Legacy. The default name is SYSFOP.RDDEF.

Consumption Meter

This table contains a count of the amount of data processed in an Archive, Convert, or Extract. The default name is SYSFOP.CPMETER.

Column Map Procedure Table

This table stores the Column Map Procedures defined using Optim. This table is required. The default name is SYSFOP.CMPROC.

General Information

All Optim system tables are defined on a single panel.

The names of the database and table space are taken from entries on the Specify Table Space(s) for Directory Tables panel, if any. Otherwise, names can be supplied on the panel, or the table space names can be omitted and "IN DATABASE" is assumed.

The SQL for each table is displayed separately when you enter Y at the **Review SQL prior to execution** prompt. (For more information about this prompt, see Appendix B, "Review SQL Statements and DSN Subcommands," on page 197.) However, the tables are actually created in a single unit of work.

```
----------------------- Creation of Directory Tables -----------------------
Command ===>

  Authorization ID ===>  SYSFOP

                ADDEF       PKDEF        RELDEF    CMDEF     ARCHDEF
  Database    ===>
  Tablespace  ===>

                ADB2AUDIT   MDB2PROCESS  TMDEF     CDDEF     ARCHIDX
  Database    ===>
  Tablespace  ===>

                LTDEF       ARCHLOG      ENVDEF    RDDEF     AFCOLLCT
  Database    ===>
  Tablespace  ===>

                CPMETER     CMPROC
  Database    ===>
  Tablespace  ===>

Review SQL prior to execution ===> NO  (Y-Yes, N-No)

  Press ENTER key to create tables
  Press END key to cancel

******************************* N O T E S ********************************
*  You must have CREATETAB privilege DBADM, DBCTRL, or DBMAINT authority   *
*  for the database, or SYSADM authority to complete this function.       *
**************************************************************************
```

The following prompts are displayed:

Authorization ID

Authorization ID used to create the Tables. The default is *SYSFOP*. If you modify this value, the new value is used for the creation process.

Table   Name of each table to be created. This value is provided and cannot be modified.

Database

  Name of the database for each table as specified on the Specify Database for Directory Tables panel. If there is no specification, **Database** is blank.

Tablespace

  Name of the tablespace for each table as specified on the Specify Table Space(s) for Directory Tables panel. If there is no specification, this is blank. You can modify table space names on this panel. If **Tablespace** is blank for all tables, DB2 creates an implicit table space for each table.

Create Indexes

In addition to creating the Optim Directory tables, you must create the indexes for these tables. (Note that an index is not created for ADB2AUDIT.)

Each table is followed by the name of its index. By default, these indexes are named:

SYSFOP.ADDEF

  SYSFOP.XADDEF

SYSFOP.PKDEF

  SYSFOP.XPKDEF

SYSFOP.CMDEF

  SYSFOP.XCMDEF

SYSFOP.RELDEF

  SYSFOP.XRELDEF

SYSFOP.MDB2PROCESS

  SYSFOP.XPTDEF

SYSFOP.CDDEF

  SYSFOP.XCDDEF

SYSFOP.TMDEF

  SYSFOP.XTMDEF

SYSFOP.ARCHDEF

  SYSFOP.XARCHDEF

  SYSFOP.XARCNAME

SYSFOP.ARCHIDX

  SYSFOP.XARCHIDX

  SYSFOP.XARCIDX2 (non-clustering index)

SYSFOP.LTDEF

  SYSFOP.XLTDEF

SYSFOP.ARCHLOG

  SYSFOP.XARCHLOG

SYSFOP.ENVDEF

  SYSFOP.XENVDEF

SYSFOP.RDDEF

  SYSFOP.XRDDEF

SYSFOP.AFCOLLCT

  SYSFOP.XAFCOLCT

SYSFOP.CPMETER
>SYSFOP.XCPMETER

SYSFOP.CMPROC
>SYSFOP.XCMPROC

The indexes for each table are specified individually. Thus, the Creating Index panel is displayed for each index that is created. The panel title indicates the name of the table for which the index is to be created. Similarly, the **Review SQL** option displays the SQL for the index being defined. (For more information about this option, see Appendix B, "Review SQL Statements and DSN Subcommands," on page 197.) Although the index creation panels are displayed separately, the indexes are created at one time, after the final index is defined.

Once you enter values for an index, the entries are profiled for the succeeding indexes. You can modify these values.

The following figure shows the panel that prompts to create the index for the table ADDEF.

```
-------------- Creating Clustering Index for SYSFOP.ADDEF ------------------
Command ===>

Index Name ===>   SYSFOP.XADDEF

Storage Group Name ===>                 PRIQTY ===>  100
      OR                                SECQTY ===>  50
VCAT Name         ===>                  ERASE  ===>  YES   (Y-Yes, N-No)

FREEPAGE ===>     (0-255)      PCTFREE        ===>        (0-99)
Close    ===> NO  (Y-Yes, N-No)  Buffer Pool ===>        (BP0-BP49)
Copy     ===> NO  (Y-Yes, N-No)

Press ENTER key to continue
Press END key to cancel                Review SQL ===>        (Y-Yes, N-No)

******************************** N O T E *S ********************************
* You must be the table owner, have INDEX privilege on the tables, DBADM  *
* authority on the database or SYSADM authority to complete this function. *
* If the index is explicitly qualified with other than your primary or    *
* a secondary AUTHID, DBADM, or DBCTRL authority on the database or SYSADM *
* is required.  In addition, specifying a BUFFER POOL or STORAGE GROUP     *
* requires the USE privilege for that resource or SYSADM authority.        *
***************************************************************************
```

Supply the information for the prompts:

Index Name
>Name assigned to the index. The default index name is provided automatically. You can modify this value to suit site conventions.

Storage Group Name
>Name of the storage group as specified in the Specify Storage Group for Directory Tables panel. You can specify either a storage group name or a VCAT catalog name, but not both.

VCAT Name
>Name of the VCAT catalog as specified in the Specify Storage Group for Directory Tables panel. You can specify either a storage group name or a VCAT catalog name, but not both.

PRIQTY
>Minimum primary space allocation. Optional for Storage Group, not allowed for VCAT.

SECQTY
>Secondary space allocation. Optional for Storage Group, not allowed for VCAT.

ERASE

Setting for filling the DB2-defined datasets with zeroes when the table space is dropped. Optional for Storage Group; not allowed for VCAT. Specify:

Y   Erase datasets.

N   Do not erase datasets.

FREEPAGE

Setting for how often to leave a page of free space when the index is loaded or reorganized. Enter a value from 0 through 255. Optional.

PCTFREE

Percentage of a page to leave as free space when the index is loaded or reorganized. Enter a value from 0 through 99. Optional.

Buffer Pool

The name of the Index buffer pool. Optional. The panel shows the valid selections (BP0-BP49).

Close   Setting for opening files. Optional. Specify Y or N.

Y   Files are to be opened and closed once per logical unit of work.

N   Files are to be opened once per session and closed when the session terminates. Default.

Copy   Setting for performing an Image Copy of this index space. Specify Y or N.

Y   Image Copy can be performed for this index space.

N   Image Copy cannot be performed for this index space. Default.

You are given the opportunity to review the specifications for each index being created. The prompts for the other indexes are the same as those described when creating the index for ADDEF. Although the index creation panels are displayed separately, the indexes are actually created in a single unit of work after the final index is defined.

Bind for Product Execution

Option 7 on the Full Installation menu performs the final bind to enable execution of Optim and to provide access to the tables created during installation.

The Provide Access to Optim at Remote/Local Subsystems menu lists the steps in the final bind. As with the Full Installation menu, you can press ENTER to proceed through the steps.

```
------------- Provide Access to Optim at Remote/Local Subsystems -------------
OPTION  ===>

     1 - Create SYNONYMs to access the DB2 Catalog & Product Directory.
     2 - Setup Plan for Product Execution at the Current Location.
     3 - Setup Plans for Product Execution at Remote Locations.

     Plan Owner          ===>
     Plan Qualifier      ===>
     Directory CreatorID    : SYSFOP
     Table Library DSN   ===> 'OPTUSR.PROD.SFOPTENU'

Optim does not explicitly reference either the Product Directory or the
DB2 Catalog.  Synonyms for the Plan Qualifier must be established to the
directory and catalog (or an alternate set of tables/views) before binding.

******************************* N O T E S ********************************
* Plan/Package owner must have SYSADM or BINDADD authority, SELECT privilege *
* on Catalog, ownership of Optim Directory Tables, and authority to change   *
* the SQLID.                                                                 *
****************************************************************************
```

Select the steps in the order listed:

1. Create the synonyms required to bind the plan.

   The synonyms are for the DB2 Catalog and the Optim Directory tables. You will have the opportunity to specify the synonyms to be used for a shadow catalog in order to run against tables other than the DB2 Catalog tables. If you do not modify the default values specified, the synonyms for the DB2 Catalog tables are created.

2. Setup plan for product execution at the current location.

   You are prompted for the specifications to bind the packages and to grant user execute authority on the packages. Also, you are prompted for the specifications to bind the plan and then to grant user execute authority on the plan.

3. Setup plans for product execution at remote locations.

   Once the packages have been created, you can setup plans to access the remote subsystems. Remote subsystems are available to users by specifying the location name in the **LOCATION** prompt provided on the Optim main menu.

   This step is performed only if you intend to access remote subsystems from the current location. This step is performed at the local subsystem.

You are prompted for the plan owner and qualifier. The plan qualifier owns the synonyms created prior to binding the plan. The plan owner must have the authorization to change the SQLID to the plan qualifier.

Note that if you intend to use more than one release of Optim at your site, you should provide a unique plan qualifier. If you specify the plan qualifier used for an existing plan, any existing synonyms are dropped to create the new synonyms, and the existing plan is then invalidated. (See Appendix C, "Run Multiple Releases," on page 199 for additional information.)

The **Directory CreatorID** is provided for your information and cannot be modified here.

You are also prompted to specify the dataset name of the table library. This table library dataset contains a table of the plans created for remote access. You can maintain this table using Option 3 on the Provide Access to Optim at Remote/Local Subsystems menu. Also, this table is used to provide the selection list of available locations if requested by a percent sign (%) in the **LOCATION** prompt on the main menu.

Option 1 - Create Synonyms

Optim runs against its own synonyms for the DB2 Catalog table and the Directory tables. Use this option to create the synonyms.

When you select Option 1 on the Provide Access to Optim at Remote/Local Subsystems menu, you are given the opportunity to override the DB2 Catalog specification prior to creating the synonyms for the Directory and the DB2 Catalog. The Specify Synonyms for Catalog Access panel is shown. On this panel, you can specify the Default Creator ID for the catalog tables or views and the names of those catalog tables or views.

In the following figure, the defaults are displayed mapping the Optim synonyms to the DB2 Catalog. If you wish to run against other tables you can change the Default Creator ID and/or the catalog table names to map the synonyms to other tables or views.

```
--------------------- Specify Synonyms for Catalog Access -------------------
 Command ===>                                           SCROLL ===> PAGE

 To execute Optim using the DB2 system catalog, press ENTER key

 To execute using alternate tables or views, change the catalog table names
 to the desired alternate names and press ENTER key

 Default Creator ID  ===> SYSIBM          1 of 34

 Synonym:                Catalog Table:
 ******************************** TOP ***************************************
 FOP_SYSCOLUMNS      ===>  SYSIBM.SYSCOLUMNS
 FOP_SYSCOPY         ===>  SYSIBM.SYSCOPY
 FOP_SYSDATABASE     ===>  SYSIBM.SYSDATABASE
 FOP_SYSFIELDS       ===>  SYSIBM.SYSFIELDS
 FOP_SYSFOREIGNKEYS  ===>  SYSIBM.SYSFOREIGNKEYS
 FOP_SYSINDEXES      ===>  SYSIBM.SYSINDEXES
 FOP_SYSINDEXPART    ===>  SYSIBM.SYSINDEXPART
 FOP_SYSKEYS         ===>  SYSIBM.SYSKEYS
 FOP_SYSRELS         ===>  SYSIBM.SYSRELS
 FOP_SYSSTOGROUP     ===>  SYSIBM.SYSSTOGROUP
 FOP_SYSSYNONYMS     ===>  SYSIBM.SYSSYNONYMS
 FOP_SYSTABLES       ===>  SYSIBM.SYSTABLES
 FOP_SYSTABLESPACE   ===>  SYSIBM.SYSTABLESPACE
 FOP_SYSVIEWS        ===>  SYSIBM.SYSVIEWS
 FOP_SYSVIEWDEP      ===>  SYSIBM.SYSVIEWDEP
 FOP_SYSTABLEPART    ===>  SYSIBM.SYSTABLEPART
 FOP_SYSPLANDEP      ===>  SYSIBM.SYSPLANDEP
 FOP_SYSPACKDEP      ===>  SYSIBM.SYSPACKDEP
 FOP_SYSPACKAGE      ===>  SYSIBM.SYSPACKAGE
 FOP_SYSPLAN         ===>  SYSIBM.SYSPLAN
 FOP_SYSCHECKS       ===>  SYSIBM.SYSCHECKS
 FOP_LOCATIONS       ===>  SYSIBM.LOCATIONS
 FOP_SYSAUXRELS      ===>  SYSIBM.SYSAUXRELS
 FOP_SYSCHECKDEP     ===>  SYSIBM.SYSCHECKDEP
 FOP_SYSCONSTDEP     ===>  SYSIBM.SYSCONSTDEP
 FOP_SYSDATATYPES    ===>  SYSIBM.SYSDATATYPES
 FOP_SYSLOBSTATS     ===>  SYSIBM.SYSLOBSTATS
 FOP_SYSPARMS        ===>  SYSIBM.SYSPARMS
 FOP_SYSROUTINES     ===>  SYSIBM.SYSROUTINES
 FOP_SYSTRIGGERS     ===>  SYSIBM.SYSTRIGGERS
 FOP_SYSSEQUENCES    ===>  SYSIBM.SYSSEQUENCES
 FOP_SYSSEQUENCESDP  ===>  SYSIBM.SYSSEQUENCESDEP
 FOP_SYSTABCONST     ===>  SYSIBM.SYSTABCONST
 FOP_SYSTBLSPCSTATS  ===>  SYSIBM.SYSTBLSPCSTATS
 ****************************** BOTTOM **************************************

 Review SQL prior to execution ===> YES  (Y-Yes, N-No)
```

Figure 7. Specify Synonyms for Catalog Access - Install

Existing Synonyms Dropped

Any existing plans that are bound with the names used for the Optim synonyms will be invalidated. Any existing synonyms that match those on this panel are dropped and new synonyms are created to satisfy this request. It is recommended that you check any names beginning with FOP_SYS since all the synonyms defined by Optim begin with this string.

If you enter Yes to the prompt **Review SQL prior to execution**, the SQL for synonyms that are to be dropped and created is displayed.

The Review DROP Synonym SQL to Bind Plan panel is displayed if the name of a synonym on the Specify Synonyms for Catalog Access panel matches that of an existing synonym. This condition occurs most frequently when you choose to re-specify the Catalog Access.

The Review DROP Synonym SQL to Bind Plan panel is displayed prior to the Review CREATE Synonym SQL to Bind Plan panel to warn that an existing synonym will be affected. The SQL statements for

dropping and creating synonyms are executed at one time, after all specifications are complete and, if desired, reviewed.

Create Synonym SQL panel

If **Review SQL prior to execution** is selected the first time this mapping is executed in the initial install process, the SQL to create all synonyms is displayed. The following figure shows the default names on the Review CREATE Synonym SQL to Bind Plan panel for the initial install. Synonyms for Optim directory tables (ADB2AUDIT, ADDEF, CMDEF, RELDEF, TMDEF, CDDEF, MDB2PROCESS, ARCHDEF, ARCHIDX, PKDEF, LTDEF, ARCHLOG, ENVDEF, RDDEF, AFCOLLCT, CPMETER, CMPROC) are also created. The names for these tables, including the Creator ID, are as entered on the Specify Synonyms for Catalog Access panel. The default values are displayed in the following figure.

```
----------------- Review CREATE Synonym SQL to Bind Plan -------------------
 Command ===>                                              SCROLL ===> PAGE

 SQL to be executed:               1 of 47


 ******************************* TOP **********************************
 CREATE SYNONYM FOP_SYSCOLUMNS      FOR SYSIBM.SYSCOLUMNS
 CREATE SYNONYM FOP_SYSCOPY         FOR SYSIBM.SYSCOPY
 CREATE SYNONYM FOP_SYSDATABASE     FOR SYSIBM.SYSDATABASE
 CREATE SYNONYM FOP_SYSFIELDS       FOR SYSIBM.SYSFIELDS
 CREATE SYNONYM FOP_SYSFOREIGNKEYS  FOR SYSIBM.SYSFOREIGNKEYS
 CREATE SYNONYM FOP_SYSINDEXES      FOR SYSIBM.SYSINDEXES
 CREATE SYNONYM FOP_SYSINDEXPART    FOR SYSIBM.SYSINDEXPART
 CREATE SYNONYM FOP_SYSKEYS         FOR SYSIBM.SYSKEYS
 CREATE SYNONYM FOP_SYSRELS         FOR SYSIBM.SYSRELS
 CREATE SYNONYM FOP_SYSSTOGROUP     FOR SYSIBM.SYSSTOGROUP
 CREATE SYNONYM FOP_SYSSYNONYMS     FOR SYSIBM.SYSSYNONYMS
 CREATE SYNONYM FOP_SYSTABLES       FOR SYSIBM.SYSTABLES
 CREATE SYNONYM FOP_SYSTABLESPACE   FOR SYSIBM.SYSTABLESPACE
 CREATE SYNONYM FOP_SYSVIEWS        FOR SYSIBM.SYSVIEWS
 CREATE SYNONYM FOP_SYSVIEWDEP      FOR SYSIBM.SYSVIEWDEP
 CREATE SYNONYM FOP_SYSTABLEPART    FOR SYSIBM.SYSTABLEPART
 CREATE SYNONYM FOP_SYSPLANDEP      FOR SYSIBM.SYSPLANDEP
 CREATE SYNONYM FOP_SYSPACKDEP      FOR SYSIBM.SYSPACKDEP
 CREATE SYNONYM FOP_SYSPACKAGE      FOR SYSIBM.SYSPACKAGE
 CREATE SYNONYM FOP_SYSPLAN         FOR SYSIBM.SYSPLAN
 CREATE SYNONYM FOP_SYSCHECKS       FOR SYSIBM.SYSCHECKS
 CREATE SYNONYM FOP_LOCATIONS       FOR SYSIBM.LOCATIONS
 CREATE SYNONYM FOP_SYSAUXRELS      FOR SYSIBM.SYSAUXRELS
 CREATE SYNONYM FOP_SYSCHECKDEP     FOR SYSIBM.SYSCHECKDEP
 CREATE SYNONYM FOP_SYSCONSTDEP     FOR SYSIBM.SYSCONSTDEP
 CREATE SYNONYM FOP_SYSDATATYPES    FOR SYSIBM.SYSDATATYPES
 CREATE SYNONYM FOP_SYSLOBSTATS     FOR SYSIBM.SYSLOBSTATS
 CREATE SYNONYM FOP_SYSPARMS        FOR SYSIBM.SYSPARMS
 CREATE SYNONYM FOP_SYSROUTINES     FOR SYSIBM.SYSROUTINES
 CREATE SYNONYM FOP_SYSTRIGGERS     FOR SYSIBM.SYSTRIGGERS
 CREATE SYNONYM FOP_SYSTBLSPCSTATS  FOR SYSFOP.SYSTBLSPCSTATS
 CREATE SYNONYM FOP_ADB2AUDIT       FOR SYSFOP.ADB2AUDIT
 CREATE SYNONYM FOP_ADDEF          FOR SYSFOP.ADDEF
 CREATE SYNONYM FOP_CMDEF          FOR SYSFOP.CMDEF
 CREATE SYNONYM FOP_RELDEF         FOR SYSFOP.RELDEF
 CREATE SYNONYM FOP_MDB2PROCESS    FOR SYSFOP.MDB2PROCESS
 CREATE SYNONYM FOP_PKDEF          FOR SYSFOP.PKDEF
 CREATE SYNONYM FOP_CDDEF          FOR SYSFOP.CDDEF
 CREATE SYNONYM FOP_TMDEF          FOR SYSFOP.TMDEF
 CREATE SYNONYM FOP_ARCHDEF        FOR SYSFOP.ARCHDEF
 CREATE SYNONYM FOP_ARCHIDX        FOR SYSFOP.ARCHIDX
 CREATE SYNONYM FOP_LTDEF          FOR SYSFOP.LTDEF
 CREATE SYNONYM FOP_ARCLG          FOR SYSFOP.ARCHLOG
 CREATE SYNONYM FOP_ENVDEF         FOR SYSFOP.ENVDEF
 CREATE SYNONYM FOP_RDDEF          FOR SYSFOP.RDDEF
 CREATE SYNONYM FOP_AFCOLLCT       FOR SYSFOP.AFCOLLCT
 CREATE SYNONYM FOP_CPMETER        FOR SYSFOP.CPMETER
 CREATE SYNONYM FOP_CMPROC         FOR SYSFOP.CMPROC
  **************************** BOTTOM **********************************


 Press ENTER key to continue
 Press END key to respecify
```

Figure 8. Review CREATE Synonym SQL to Bind Plan - Install

Press ENTER to execute the SQL. However, if you wish to change any values, press END to redisplay the Specify Synonyms for Catalog Access panel.

Note: Since the Review DROP Synonym SQL to Bind Plan and the Review CREATE Synonym SQL to Bind Plan panels are similar, only the CREATE Synonym SQL panel is shown here.

Option 2 - Setup Plan for Product Execution at Current Location

When you select Option 2 on the Provide Access to Optim at Remote/Local Subsystems menu, the Setup Packages to allow Access to Current Location panel is displayed.

```
  ------------  Setup Packages to allow Access to Current Location -----------
OPTION  ===>


     1 - Create Packages to allow Access to Current Location.
     2 - Grant User Execute Authority on the Packages.
     3 - Create Plan to allow Access to Current Location.
     4 - Grant User Execute Authority on the Plan.




******************************** N O T E S ********************************
*     The Package Owner must have BINDADD privilege or SYSADM authority    *
*     to complete these functions successfully.                           *
**************************************************************************
```

This panel lists the steps required to bind the packages and plan to access the current location.

1. Bind the packages.

   Specify the information to bind the packages. You must bind the packages prior to binding the plan.

2. Grant user execute authority on the packages.

   Assign Grant authority on the packages.

3. Bind the plan.

   Specify the information to perform the final bind. Note that if you intend to provide both local and remote access, you must assign a unique plan name to each subsystem installation.

4. Grant user execute authority on the plan.

   Even if you intend to provide remote access only, you should perform the bind for local access; it is required for Directory maintenance. You can still limit general use to remote access. For local access on the remote subsystem, assign Grant authority to the installer only or to those responsible for Directory maintenance. (Note that if you FREE the plan for local access, you cannot perform Directory maintenance using the Optim facilities.)

Bind Package for Current Location

When you select Option 1 on the Setup Packages to allow Access to Current Location panel, the Bind Package for Current Location panel is displayed. This panel collects the information to bind the packages to allow access to the current location. You must execute this step prior to the final bind of Optim.

```
--------------------- Bind Package for Current Location ----------------------
COMMAND ===>


Package Owner              : OPTUSR        Package Qualifier:

Collection-ID Prefix       ===> OPTPACKAGE
DBRM Library DSN           ===>

    Review DSN subcommand prior to execution ===> NO  (Y-Yes, N-No)


Enter Options:
  Bind Action             ===> REPLACE    (Add, Replace)
  Isolation Level         ===> CS         (CS, RR)
  Validate                ===> BIND       (Run, Bind)
  Resource Release Time   ===> COMMIT     (Commit, Deallocate)
  Explain Plan Selection  ===> NO         (Y-Yes, N-No)
  Bind Message Level      ===> ALL        (All, Warning, Error, Complete)
  Data Currency           ===> YES        (Y-Yes, N-No)
  Encoding Scheme         ===> 0037       (Non Unicode Code Page)


******************************** N O T E S ********************************
*     The Package Owner must have BINDADD privilege or SYSADM authority    *
*     to complete this function successfully.                             *
**************************************************************************
```

The following prompts are presented:

Package Owner
> Name of the package owner. Only the owner can modify or free the package. This value, which was entered on the Provide Access to Optim at Remote/Local Subsystems panel, is protected, and is provided for information only.

Package Qualifier
> The authorization ID for which the synonyms were created. This value, which was entered on the Provide Access to Optim at Remote/Local Subsystems panel, is protected, and is provided for information only.

Collection-ID Prefix
> The prefix used to generate the name assigned to the collection of packages. The name is formed by concatenating this prefix with internal codes to identify the release and the browse or edit specification.

DBRM Library DSN
> Name of the library containing the DBRMs. By default, this is the name of the install library unless changed for the temporary bind on the Bind Plan for Product Install panel. If you have moved the DBRMs, specify the new DBRM library.

Review DSN subcommand
> Setting to review the generated DSN subcommand before it is executed. This value is profiled. (See Appendix B, "Review SQL Statements and DSN Subcommands," on page 197 for details.) Specify:
>
> `Y`   Review the DSN subcommand prior to execution.
>
> `N`   Execute the DSN subcommand without reviewing it.

Bind Action
> The bind action. Specify:
>
> `Add`   Add a new package.
>
> `Replace`
> > Overlay an existing package of the same name with a new package.

Isolation Level
> Setting for the effect of executing one application bound to this plan on the other applications in this subsystem. Specify:
>
> `CS`   For Cursor Stability. Database values read are protected only while they are used. Values that are changed are protected until they are committed. IBM recommends Cursor Stability.
>
> `RR`   For Repeatable Read. Database values read or changed by this application are protected and cannot be changed by other applications until this one commits or terminates.

Validate
> Setting for validity checking. Specify:
>
> `Run`   Defer checking until the plan is executed.
>
> `Bind`   Check validity during bind processing.
>
> IBM recommends Bind to speed catalog access and reduce authorization requirements. Product access to user data uses dynamic SQL, which is unaffected by this option.

Resource Release Time
> Setting for release of resources. Specify:
>
> `Commit` Release resources at each COMMIT point.

Deallocate
> Release resources when application terminates.

Explain Plan Selection
> Setting for providing information about the execution of SQL statements in the PLAN. Specify:

> **Y**   EXPLAIN information is provided.

> **N**   EXPLAIN information is not provided.

Bind Message Level
> Setting for level of messages to be displayed. Specify:

> **All**   All messages.

> **Warning**
> > Warning, error and completion messages.

> **Error**   Only error and completion messages.

> **Complete**
> > Completion messages.

Data Currency
> Data currency setting for read-only and ambiguous cursors when the isolation level of cursor stability is in effect. Specify Y or N.

> **Y**   Data currency is required. Default.

> **N**   Data currency is not required.

Encoding Scheme
> If your DB2 Default Encoding Scheme (DSNHDECM module ENSCHEME value) in the DSNHDECP system file is set to Unicode, enter a 4-digit EBCDIC code page value. The recommended value is 0037.

> If you do not supply a value at this prompt, the DSNHDECM value is used.

> **Notes:**
> - While the Optim plan must be bound with an EBCDIC encoding scheme, Optim is still able to process data in ASCII and Unicode DB2 tables. When Optim is reading data from an ASCII or Unicode DB2 table, DB2 will convert characters from the table's encoding scheme to the target Optim EBCDIC encoding scheme. When Optim is writing data to an ASCII or Unicode DB2 table, DB2 will convert characters from the Optim EBCDIC encoding scheme to the target table's encoding scheme.
> - If you use a value other than 0037 for code page, then the loading of the sample database will fail with a mismatched CCSID error. To prevent this failure, set the Site Option value for Allow Mismatched CCSIDs to Y during customization of the Site Options step.
> - The code page value is not validated. Use caution when entering a value.

If the packages are successfully bound, the Setup Packages to allow Access to Current Location panel is redisplayed. If the bind is unsuccessful, the Setup Packages to allow Access to Current Location panel is redisplayed with a message. You can re-specify the bind options for the packages, as needed.

Grant Authority on Package

Select Option 2 on the Setup Packages to allow Access to Current Location panel to display the Grant Authority on Optim Package panel.

```
--------------  Grant Authority on Optim Package --------------------------
COMMAND ===>

Grant EXECUTE authority to:

             PUBLIC    ===>    (Y-Yes, N-No)

             < OR >

The following AUTHID(s) ===>          ==>           ===>          ===>
                        ===>          ==>           ===>          ===>
                        ===>          ==>           ===>          ===>
                        ===>          ==>           ===>          ===>

Review SQL prior to execution ===> YES  (Y-Yes, N-No)

******************************* N O T E S *********************************
*         You must have authority to grant access to the package         *
*         in order to complete this function successfully.                *
**************************************************************************
```

Figure 9. Grant Authority on Package

The following are presented:

PUBLIC
> Setting for public access. Specify:
>
> **Y**   The package is PUBLIC and anyone can access it.
>
> **N**   The package is not PUBLIC. You must enter the Authorization ID for each user who is to have access.

AUTHID(s)
> The Authorization IDs for the package. The panel provides 16 8-character prompts for Authorization IDs and if the setting for **PUBLIC** is N, the panel is redisplayed automatically to allow you enter additional names. If this panel is redisplayed, it is populated with the previously specified values.

Commands

Use the CLEAR command (or the short form CLE) to clear **AUTHID** values. Press the TAB key to move from one **AUTHID** prompt to the next. Use END to exit the panel.

Bind Plan

When you select Option 3 on the Setup Packages to allow Access to Current Location panel, the Bind Plan for Product Execution panel is displayed. This panel collects the information used to bind the plan in order to execute Optim.

```
----------------------- Bind Plan for Product Execution --------------------
COMMAND ===>

Plan Owner                    : OPTUSER      Plan Qualifier:
Plan Name                  ===> FOPnnn
Collection-ID Prefix       ===> FOPPKG

Include Package List                     ===> NO   (Y-Yes, N-No)
Allow Only Browse Using Plan             ===> NO   (Y-Yes, N-No)
Review DSN subcommand prior to execution ===> NO   (Y-Yes, N-No)

Enter Options:
  Bind Action               ===> REPLACE    (Add, Replace)
  Retain Execution Authority ===>           (Y-Yes, N-No)
  Isolation Level           ===> CS         (CS, RR)
  Validate                  ===> BIND       (Run, Bind)
  Resource Acquisition Time ===> USE        (Use, Allocate)
  Resource Release Time     ===> COMMIT     (Commit, Deallocate)
  Explain Plan Selection    ===> NO         (Y-Yes, N-No)
  Bind Message Level        ===> ALL        (All, Warning, Error, Complete)
  Data Currency             ===> YES        (Y-Yes, N-No)
  Encoding Scheme           ===> 0037       (Non Unicode Code Page)

******************************* N O T E S *******************************
*      The plan owner must have BINDADD privilege or SYSADM authority      *
*      to complete this function successfully.                             *
*************************************************************************
```

Figure 10. Bind Plan for Product Execution - Install

The Bind Plan for Product Execution panel includes:

Plan Owner
> Name of the plan owner. The value entered on the Provide Access to Optim at Remote/Local Subsystems panel is displayed for information only.

Plan Qualifier
> The authorization ID for which the synonyms were created. The value entered on the Provide Access to Optim at Remote/Local Subsystems panel is displayed for information only.

Plan Name
> Name of the current plan. The default is the value specified for the initial temporary bind on the Bind Plan for Product Install panel. If the plan name has been changed, you must modify the FOPCUST CLIST to reflect the new plan name before attempting to execute Optim. (See Chapter 2, "Set Up and Modify the Optim CLIST," on page 7 for information on modifying FOPCUST.)

Collection-ID Prefix
> The prefix used to identify the collection of packages. By default, the value specified on the Bind Package for Current Location panel is displayed.

Include Package List
> Setting for packages in the PKLIST option in the BIND PLAN statement. Specify:
>
> **Y**   Include package list. Display a panel on which you can specify the names of packages to add to the PKLIST option in the BIND PLAN statement. For details, see "Package List for Bind Plan" on page 41.
>
> **N**   Do not include package list. Default.

Allow Only Browse Using Plan
> Setting for data modification permissions. Specify:
>
> **Y**   Plan permits browse only.
>
> **N**   Plan permits browsing and data modification.

Review DSN subcommand

Setting for review of the generated DSN subcommand before it is executed. Specify:

`Y` Review the DSN subcommand prior to execution.

`N` Execute the DSN subcommand without review.

This setting is profiled. See Appendix B, "Review SQL Statements and DSN Subcommands," on page 197 for details.

Bind Action

Setting for the bind action. Specify:

`Add` Add a new plan.

`Replace`

Overlay an existing plan of the same name. This is the default for the final bind.

Retain Execution Authority

Setting for retention of authorizations when Bind Action is Replace. Specify:

`Y` Retain BIND authority and EXECUTE authority.

`N` Do not retain BIND authority and EXECUTE authority.

Isolation Level

Setting for the effect of executing an application bound to this plan on other applications in this subsystem. Specify:

`CS` For Cursor Stability. Database values read are protected only while they are used and values that are changed are protected until they are committed. IBM recommends Cursor Stability.

`RR` For Repeatable Read. Database values read or changed by this application are protected and cannot be changed by other applications until this one commits or terminates.

Validate

Setting for validity checking. Specify:

`Run` Defer checking until the plan is executed.

`Bind` Check validity during bind processing.

IBM recommends Bind to speed catalog access and reduce authorization requirements. Optim accesses user data using dynamic SQL which is unaffected by this option.

Resource Acquisition Time

Setting for acquiring resources. Specify:

`Use` Open table spaces and acquire locks when the plan first uses them.

`Allocate`

Open all table spaces and acquire all locks when the plan is allocated.

Note that if you use Allocate, you must use Deallocate for **Resource Release Time**.

Resource Release Time

Setting for releasing resources. Specify:

`Commit` Release resources at each commit point.

`Deallocate`

Release resources when application terminates.

Explain Plan Selection

Setting for provision of information about the execution of SQL statements in the PLAN. Specify:

`Y` EXPLAIN information is provided.

N   EXPLAIN information is not provided.

Bind Message Level

Setting for type of messages to be displayed. Specify:

All   All messages.

Warning
Warning, error and completion messages.

Error   Only error and completion messages.

Complete
Completion messages.

Data Currency

Indicator for data currency of read-only and ambiguous cursors when **Isolation Level** setting is CS. Specify:

Y   Data currency is required. Default.

N   Data currency is not required.

Encoding Scheme

If your DB2 Default Encoding Scheme (DSNHDECM module ENSCHEME value) in the DSNHDECP system file is set to Unicode, enter a 4-digit EBCDIC code page value. The recommended value is 0037.

If you do not supply a value for this prompt, the DSNHDECM value is used.

Notes:

- While the Optim plan must be bound with an EBCDIC encoding scheme, Optim is still able to process data in ASCII and Unicode DB2 tables. When Optim is reading data from an ASCII or Unicode DB2 table, DB2 will convert characters from the table's encoding scheme to the target Optim EBCDIC encoding scheme. When Optim is writing data to an ASCII or Unicode DB2 table, DB2 will convert characters from the Optim EBCDIC encoding scheme to the target table's encoding scheme.
- The code page value is not validated. Use caution when entering a value.

Package List for Bind Plan

If you enter Yes for **Include Package List**, the Package List for Bind Plan panel is displayed.

```
------------------------ Package List for Bind Plan -------------------------
OPTION  ===>                                              SCROLL ===> PAGE

Enter names to be included in PACKAGE list for:    FOPnnn

     COLLECTION         PACKAGE-ID        COLLECTION         PACKAGE-ID
  ------------------    ----------     ------------------    ----------
  _____      _____        _____      _____
  _____      _____        _____      _____
  _____      _____        _____      _____
  _____      _____        _____      _____
  _____      _____        _____      _____
  _____      _____        _____      _____



************************** I N S T R U C T I O N S **************************
*                                                                          *
*     Any packages specified on this panel must exist and be valid         *
*                                                                          *
****************************************************************************
```

Figure 11. Package List for Bind Plan

Use this panel to enter the names of any collections and packages to add to the PKLIST option in the BIND PLAN statement. Any names you enter on this panel are profiled and will be included in subsequent BIND PLAN statements if you set **Include Package List** to Yes.

Note: It is assumed that any packages you enter on this panel exist and are valid.

When you have completed entries on this panel, press ENTER to perform the final bind, or use END to return to the Bind Plan for Product Execution panel.

Grant Authority on Plan

Use Option 4 on the Setup Packages to allow Access to Current Location panel to grant execute authority on the plan. The Grant Authority on Optim Plan panel allows you to indicate the users that will have access to the current Optim plan.

This information can be re-specified as needed. For information about the prompts on this panel, see "Grant Authority on Package" on page 37.

Option 3 - Setup Plans for Execution at Remote Subsystems

Use Option 3 on the Provide Access to Optim at Remote/Local Subsystems menu to build a plan to access remote subsystems from the current location. If you intend to access remote subsystems from the current location, you must perform this step at the local subsystem.

The local subsystem is the subsystem that resides on the same z/OS system as the TSO user. A remote subsystem is a subsystem that resides across a remote connection (for example, DDF or DRDA) from the local subsystem.

Note: In addition to the steps documented in this manual, the DB2 libraries for the remote subsystem must be physically accessible by the current z/OS system to establish remote access.

When you select Option 3 on the Provide Access to Optim at Remote/Local Subsystems menu, the Setup Plans for Product Execution at a Remote Subsystem panel is displayed. This panel collects the information to bind the plan in order to provide access to remote subsystems.

```
-- Setup Plans for Product Execution at a Remote Subsystem --------------------
OPTION  ===>                                            SCROLL ===> PAGE

Directory CreatorID    :
Table Library DSN      :
Plan Owner        ===>                    Plan Qualifier ===>


*************************** I N S T R U C T I O N S ***************************
*Use input fields to specify each remote subsystem. These line commands are  *
*available: (C)opy Plan, (DR)elete Table Entry and Drop Plan, (R)eplace Plan,*
*          (G)rant Authority to Plan, (D)elete Table Entry for Plan.         *
******************************************************************************


Cmd  Plan Name   DB2 Location     Optim Location     Browse/Edit  Show SQL(Y/N)
----  ---------  ---------------  ---------------  -----------  --------------
         _____     _____  _____    _____             YES

**** *************** Existing Plans and Locations in Optim Table ************
  _   GOTOFOP1  FOP1DB241       SUBSYS1         EDIT
  _   GOTOFOP2  FOP2DB241       SUBSYS2         EDIT
**** ************************** BOTTOM OF DATA *****************************
```

Figure 12. Setup Plans for Remote Subsystem - Install

Use this panel to create, modify, and delete plans for access to remote subsystems.

Note:
- If you do not use the same CODEPAGE for remote and local systems, unpredictable errors may occur when accessing Optim objects.
- If you use a value other than 0037 for code page, then the loading of the sample database will fail with a mismatched CCSID error. To prevent this failure, set the Site Option value for Allow Mismatched CCSIDs to Y during customization of the Site Options step.

Panel

The following prompts are presented:

Directory CreatorID
>The Optim Directory for the bind step. This value cannot be modified.

Table Library DSN
>Name of the table library for the table that includes an entry for each available remote plan. This table is maintained by Optim as plans are created, modified, and deleted. Specify a suffix in the RTTBLPFX variable in the FOPCUST CLIST. The table name is generated by suffixing the value you define for RTTLBPFX with a subsystem identifier.

Plan Owner
>The name of the plan owner. The default value was entered on the Provide Access to Optim at Remote/Local Subsystems panel.

Plan Qualifier
>The authorization ID for which the synonyms were created. The default value was entered on the Provide Access to Optim at Remote/Local Subsystems panel.

Input Area

An input area is provided for defining a new plan or modifying an existing plan. You can define a new plan by supplying values for the following:

Plan Name
>Name of the plan.

DB2 Location

DB2 Location name for the subsystem to be accessed. (This corresponds to the LOCATION column in SYSIBM.LOCATIONS.)

Optim Location

Site-specified name assigned to the DB2 Location. Since DB2 Location names are sometimes unfamiliar to the users, you can provide a nickname for the actual DB2 Location name. This nickname is specified in **LOCATION** on the main menu to access a remote DB2 subsystem.

Browse/Edit

Privileges for the plan. Enter Edit if users are allowed to modify the data, perform Insert or Restore Processes or to delete data.

You can establish a plan to limit specific users to read-only privilege and a second plan that does not restrict other users.

Show SQL (Y/N)

Setting for review of generated SQL.

Plan Selection List

In addition to entering the information directly in the input area, a scrollable list of available plans for the subsystem is provided on the panel. This list is maintained in the table library member that you named using the RTTBLPFX variable in the FOPCUST CLIST. Use the following line commands to maintain existing plans and create new ones.

D Delete the plan. Use D to remove the entry from the table. The plan is not FREEd.

DR Delete the plan. Use DR to FREE the plan and remove the entry from the table.

R Replace the plan. Use R to modify the plan. The plan information is copied into the input area. You can modify the **DB2 Location** and the **Browse/Edit** values. Do not modify the other values.

C Copy the plan. Use C to model a new plan on an existing plan. The plan information is copied into the input area. You must modify the **Plan Name** and the **Optim Location** values. You can also modify the other values.

G Grant authority. Use G to display the Grant Authority panel and specify access privileges for the designated plan.

Bind Plan for Remote Access

When you create a new plan or modify an existing plan for remote access, the Bind Plan for Remote Access panel is displayed.

```
----------------------- Bind Plan for Remote Access ----------------------
COMMAND ===>

Plan Owner                   :            Plan Qualifier:
Plan Name                    : planname
Location                     :
Collection ID Prefix    ===>

Review DSN subcommand prior to execution ===> NO  (Y-Yes, N-No)

Enter Options:
  Bind Action              ===> REPLACE    (Add, Replace)
  Retain Execution Authority ===> YES      (Y-Yes, N-No)
  Isolation Level          ===> CS         (CS, RR)
  Validate                 ===> BIND       (Run, Bind)
  Resource Acquisition Time  ===> USE      (Use, Allocate)
  Resource Release Time    ===> COMMIT     (Commit, Deallocate)
  Explain Plan Selection   ===> NO         (Y-Yes, N-No)
  Bind Message Level       ===> ALL        (All, Warning, Error, Complete)
  Data Currency            ===> YES        (Y-Yes, N-No)

******************************* N O T E S *****************************
*    The plan owner must have BINDADD privilege or SYSADM authority    *
*    to complete this function successfully.                          *
**********************************************************************
```

Figure 13. Bind Plan for Remote Access - Install

Panel

Many of the prompts on this panel are documented in "Bind the Plan for Product Installation" on page 18. The following are specific to binding a remote access plan or have additional limitations.

Plan Owner
Name of the plan owner. This value, entered on the Setup Plans for Product Execution at a Remote Subsystem panel, cannot be modified.

Plan Qualifier
The authorization ID for which the synonyms were created. This value, entered on the Setup Plans for Product Execution at a Remote Subsystem panel, cannot be modified.

Plan Name
Name of the remote plan that you are binding. This value, specified in **Plan Name** in the input area of the Setup Plans for Product Execution at a Remote Subsystem panel, cannot be modified.

Location
DB2 location. This value, specified in **DB2 Location** on the Setup Plans for Product Execution at a Remote Subsystem panel, cannot be modified.

Collection-ID Prefix
Prefix used to generate the name assigned to the collection of packages. The name is formed by concatenating this prefix with internal codes to identify the release and the browse or edit specification. The default value, specified on the Bind Package for Current Location panel, is displayed.

Bind Action
The bind action is determined by whether the plan is created or updated. When creating a new plan, **Bind Action** must be ADD. When modifying a plan, **Bind Action** must be REPLACE. **Bind Action** is supplied appropriately.

After the bind is successfully executed, the Grant Authority panel is displayed. For information about this panel, see "Grant Authority on Plan" on page 42.

Accessing Remote Systems

After Optim is installed, you access remote systems by entering the name of the location at the **LOCATION** prompt on the main menu. If you leave **LOCATION** blank, the local subsystem is assumed.

```
-------------------------- IBM's InfoSphere Optim --------------------------
OPTION  ===>

0  OPTIONS         - Site and User Options       SQLID  ===> IBMUSR
1  BROWSE TABLE    - Browse a DB2 Table          SUBSYS ===> DSNC
2  EDIT TABLE      - Edit a DB2 Table            LOCATION ===>
3  BROWSE USING AD - Browse DB2 Tables Using Access Definition
4  EDIT USING AD   - Edit DB2 Tables Using Access Definition
5  ADS             - Create or Modify Access Definitions
6  DEFINITIONS     - Maintain InfoSphere Optim Definitions (Keys, Maps, ...)
7  MIGRATION       - Data Migration - Extract, Insert, Update, ...
8  COMPARE         - Compare Two Sets of Data
9  ARCHIVE         - Archive and Restore Data

T  TUTORIAL        - Information About IBM's InfoSphere Optim
C  CHANGES         - Changes from Prior Release(s)
X  EXIT            - Terminate Product Use
P  LICENSING       - Product Licensing Modification
```

Figure 14. Main Menu—Remote Access

Enter a % to display a selection list of available location names. Use the Select line command, S, to select from the list.

Enable Optim

Option 8 on the Full Installation menu is used to enable Optim for execution. This is established on the Product Licensing/Configuration panel.

The Product Licensing/Configuration panel is displayed initially as shown in the following figure.

```
------------------ Product Licensing/Configuration ---------------------------
Command ===>                                          Scroll ===> PAGE

Site CPUID        :
Load Library DSN  :

     Product / Release        Current Product Status       Password
  ------------------------   ----------------------------   --------- 1 of 9
  ***************************** TOP *******************************
  Optim Access             Being Installed                _____
  Optim Move w/ Delete/DP  Being Installed                _____
  Optim Compare            Being Installed                _____
  Optim Archive            Being Installed                _____
  Optim Move for VSAM/Seq  Being Installed                _____
  Optim Move for IMS       Being Installed                _____
  Optim Compare for VSAM/Seq  Being Installed             _____
  Optim Compare for IMS    Being Installed                _____
  Optim Data Privacy       Being Installed                _____

INSTRUCTIONS: Press ENTER to Continue, Enter END to return or CANCEL to Cancel
              Enter Password to Enable, Extend Evaluation or License Product
              Enter DISABLE in Password field to Disable Product.


***************************** N O T E S ********************************
*         You must have authority to update the Load Library          *
*         to complete this function successfully.                     *
***********************************************************************
```

Figure 15. Product Licensing - Being Installed

The Product Licensing/Configuration panel identifies the products that are being installed, along with their status, which is listed as "Being Installed." After you press ENTER, the status is revised to "In Evaluation" followed by the number of days remaining in the evaluation period.

Note that for the initial trial period, the product is installed for a standard evaluation period. Product names may vary, depending upon your site's license. For example, when Move includes the options for Delete and Data Privacy functions, its name appears as shown in Figures 24 and 25.

The products that are not being installed are noted as "Not installed" and will not be available to users.

See Chapter 7, "Enable and Disable Products," on page 109 for further discussion of the Product Licensing/Configuration panel.

Unload Optim Sample Data and ODM Optim Connect

Option 9 on the Full Installation menu unloads Optim samples and ODM Optim Connect.

A sample database is distributed with Optim to help demonstrate various functions. Sample tables for the database are distributed in XMIT format. These tables are created and loaded in Step 11 of the installation.

ODM Optim Connect provides read-only access to Archive Files. The ODM data sets are also provided in XMIT format. See Appendix F, "ODM Optim Connect," on page 215 for details on installing ODM.

```
───────────── Unload Optim Sample Data and ODM Optim Connect ─────────
OPTION ===>                                                    More:    +

Below is a list of the Optim samples and ODM Optim Connect
files that have been packaged with Optim in XMIT format.
Select the files that you wish to unload and use in your environment.
 You may optionally specify a unit, volume, and STORCLAS for the unloaded
data sets.

Unload selection list:
    Enter "/" to select the data sets to unload.

    _ Unload All

    _ FOPNDEMO - Sample extract file
    _ FOPNLGCY - Sample legacy file
    _ FOPNBKOR - Sample sequential legacy file
    _ FOPNVEND - Sample legacy file (VSAM)
    _ FOPNVENI - Sample legacy file (IMS)
    _ FOPNDEPT - Sample legacy file (IMS)
    _ FOPNDEIX - Sample legacy file (IMS)
    _ FOPNJOB  - Sample legacy file (IMS)
    _ FOPNJOIX - Sample legacy file (IMS)
    _ FOPODMKT - ODM Install Kit
    _ FOPODMLD - ODM Install Load

.  Optional attributes for unloaded data sets:
.      UNIT             ===>
.      VOLUME           ===>
.      STORCLAS         ===>
```

Figure 16. Unload Optim Sample Data and ODM Optim Connect

On this panel, select the files you want to unload. You can optionally enter values for UNIT, VOLUME and STORCLAS for the unloaded data sets. When you press Enter, Optim unloads those selected and creates data sets using the naming convention *qual1.qual2*.DEMO.EXTRACT, where *qual1.qual2.* are the first two qualifiers of your installation.

Sample data privacy tables

Optim provides enhanced sample data privacy tables that can be used with the Optim data privacy providers for masking data. You can mask company and personal data such as employee names, customer names, social security numbers, credit card numbers, and email addresses. Use the tables to generate transformed data that is both unique and valid within the context of the application.

You can use the data privacy tables to:
- Prevent internal privacy breaches by de-identifying or masking the data available to developers, quality assurance testers, and other personnel.
- Improve privacy compliance initiatives by substituting customer data with contextually accurate, but fictionalized data.
- Protect confidential customer information and employee data in your application development and testing environments.
- Ensure valid test results by propagating masked elements across related tables to ensure the referential integrity of the database.

Each category of personal data is provided in a separate table for the following countries (abbreviations are in parentheses): Australia (AU), Canada (CA), France (FR), Germany (DE), Italy (IT), Japan (JP), Spain (ES), United Kingdom (UK), and United States (US). Each table includes a column of sequential numbers that is used with lookup policies that use hash values to select a row in the lookup table.

Each table name is composed of a country abbreviation prefix and the category (*countryabbreviation_category*). For example, the address table for Canada is named CA_ADDRESSES and the address table for Germany is named DE_ADDRESSES.

The schema includes the following categories:

ADDRESSES
> Tables that include columns for street address, city, locality (e.g., state or province), and postal code.

FIRSTNAME
> Tables that include a column with male and female given names.

FIRSTNAME_F
> Tables that include a column with female given names.

FIRSTNAME_M
> Tables that include a column with male given names.

LASTNAME
> Tables that include a column with family names.

PERSON
> Tables that include columns for birth date, given name, family name, gender, phone number, national ID number, company name, and email address.

CCN Tables that include a credit card number for the associated issuer (MasterCard, VISA, etc.).

DOMAIN_NAMES
> Table that includes domain names for masking email addresses.

To create and load the sample data privacy tables, modify and run the following sample jobs in the SFOPSAMP library:
- FOPPDENG (English)
- FOPPDLA (English, Spanish, French, German, Italian)
- FOPPDJPN (Japanese)

Customizing the Site Options

The Site Options panel provides parameters for configuring the Optim environment at your site. Using this panel modifies FOPMDFLT, the distributed site defaults module that contains the default site options.

Use Option 10 on the Full Installation menu to review and modify the Site Options. The Site Options panel can also be invoked from any of the following menus or panels:

- Option 6 on the main install menu.
- Option 0.0 on the main product menu (provided the user is authorized).
- **OPTIONS SITE** command on any panel during a user session (provided the user is authorized).

The following figure shows the first page of the default Site Options panel.

```
------------------------ FOPMDFLT Site Options ----------------------------
Command ===>                                          Scroll ===>  CSR



******************************** TOP **********************************
Audit Mode                ===> N      A-Always, N-Never, U-User
Load Utility to Use       ===> D      D-DB2 Load, B-BMC, U-User
Use Special Unload Utility ===> N     N-None, B-BMC, I-IBM, C|O-CDB
Unload Utility Strategy    ===> H     U-Unload Utility Only, H-Hybrid
Confirm on Deletes        ===> Y      Y-Yes, N-No
Submit Jobs with END      ===> Y      Y-Yes, N-No
Display DB2 Subsystem      ===> Y     Y-Yes, N-No
Selection List Format     ===> S      S-Short, F-Full, D-Desc
User Supplies Defaults    ===> U      A-Always, N-Never, U-User
NULL as Insert Default    ===> U      A-Always, N-Never, U-User
Table Lock Mode           ===> C      C-Can Not Lock, U-User
Editor Fetch Rows         ===> 1000   In Range (1 - 4294967295)
Maximum Compare Fetch Rows ===> 100000 In Range (1 - 4294967295)
Process Batch Compare Only ===> N     Y-Yes, N-No
Maximum Archive Rows      ===> 100000 In Range (1 - 4294967295)
Maximum Restore Rows      ===> 100000 In Range (1 - 4294967295)
****************************************************************************
```

Figure 17. Site Options

The site options are:

Audit Mode

For Access only. Audit activity for edit sessions. Specify:

- **A** Activate the Audit Facility. Users are unable to deactivate it.

- **N** Deactivate the Audit Facility. Users are unable to activate it. Default.

- **U** Allow users to activate and deactivate, as desired.

Load Utility to Use

The load utility used for the Load process. Specify:

- **D** Use the DB2 Load utility. Users cannot override. Default.

- **B** Use the BMC LOADPLUS utility. Users cannot override.

- **U** Allow users to use DB2 Load or LOADPLUS, as desired.

Use Special Unload Utility

The unload utility available when extracting data. Specify:

- **N** None. No special unload utility is available. Default.

- **B** The BMC UNLOAD PLUS utility is available.

I   The IBM High Performance Unload utility is available.

C|O   A CDB unload utility is available. Indicate the version by specifying `C` or `O`:
- `C` for CDB Auto-Unload; formerly known as Super Unload.
- `O` for CDB Auto-Online Unload; formerly known as RW-Unload.

Unload Utility Strategy

The strategy for using a special unload utility, when the response to **Use Special Unload Utility** is anything but `N`. Specify:

U   Use only the special unload utility. Exclusive use of the special unload utility allows a site to avoid overhead or contention that is possible using DB2.

H   Use the faster method, either DB2 or the special unload utility. This option is the default and allows a site to obtain the best performance.

Confirm on Deletes

Setting for user-enabled confirmation prompt on deletion of an Optim object. Specify:

N   Users cannot request confirmation prompt.

Y   User option allows user to request confirmation prompt. Default.

Submit Jobs with END

The default for a User Option to submit jobs while reviewing JCL and control statements for any batch job. Specify:

N   Users cannot submit job by using END. The **SUBMIT** command must be used to submit the job.

Y   Users can choose to submit job by using **END**.

Display DB2 Subsystem

The default for a User Option governing the display of the DB2 subsystem name on panels. Specify:

N   DB2 subsystem name is not displayed.

Y   DB2 subsystem name is displayed on every panel. Default. If connected to a remote subsystem, the DB2 subsystem name is replaced with the location of the remote subsystem.

Selection List Format

The information that is presented on a selection list of Optim objects. Specify:

S   Short format. Modification information is displayed with the object name. Default. (This is the display format used in figures throughout this documentation.)

D   Description is displayed with the object name.

F   Each entry is presented on two lines. The first line is the same as for Short format. The second line displays the description and the security status, if other than PUBLIC.

User Supplies Defaults

For Access only. Setting for user-supplied default values for columns with no DB2 default. Specify:

A   Always require user to supply a default value.

N   Never require user to supply a default value.

U   User determines whether to supply default values. Default.

NULL as Insert Default

For Access only. Indicator for blank column defined as nullable. Specify:

A   The NULL value is always supplied when inserting data.

N The NULL value is never supplied.

U User determines whether to supply the NULL value. Default.

Table Lock Mode
The table lock option for Insert, Restore, Delete or Archive with Delete Processes. Specify:

C Users cannot lock tables. The prompt for locking tables for each process is always set to No and cannot be changed. This setting lets sites force the frequency of commits because the commit logic occurs only for tables that are not locked. Default.

U Users can determine whether tables are to be locked for each process individually.

Editor Fetch Rows
The maximum number of rows that can be retrieved in one fetch set, when editing data, or browsing an extract or archive file, for example. This value can be from 1 through 4,294,967,295, and is displayed on the Editor and Display Options panel, where a lower limit can be specified. The default is 1,000.

Maximum Compare Fetch Rows
The maximum number of rows that can be fetched when browsing a Compare File. This value is displayed on the Compare Options panel where a lower limit can be specified. The default is 100,000.

Process Batch Compare Only
The batch processing default for Compare. Specify:

N Allow users to execute Compare Processes online or in batch. Default.

Y Perform all Compare Processes in batch.

Maximum Archive Rows
The maximum number of rows that can be archived in a single Archive Process. This value is displayed on the Specify ARCHIVE Parameters and Execute panel, where a lower limit can be specified. The default value is 100,000.

Maximum Restore Rows
The maximum number of rows that can be restored in a single Restore Process. The default value is 100,000.

Maximum Extract Rows
The maximum number of rows that can be extracted in a single process. This value is displayed on the Extract and Compare Specify Parameters and Execute panels, where a lower limit can be specified. The default value is 10,000.

Maximum Insert Rows
The maximum number of rows that can be inserted in a single process. For Extract Files that exceed this number, use the Load Facility. The default value is 10,000.

Maximum Delete Rows
The maximum number of rows that can be deleted in a single process. The default value is 10,000.

Maximum Online Archive Rows
The maximum number of rows that can be archived online in a single process. If this value is exceeded, the Archive Process must be executed as a batch job. This value cannot exceed the value in **Maximum Archive Rows**.

Maximum Online Restore Rows
The maximum number of rows that can be restored online in a single process. If this value is exceeded, the Restore Process must be executed as a batch job. This value cannot exceed the value specified in **Maximum Restore Rows**.

Maximum Online Extract Rows

The maximum number of rows that can be extracted online in a single process. (Does not apply to Archive Processes). If this value is exceeded, the Extract Process must be executed as a batch job. This value cannot exceed the value specified in **Maximum Extract Rows**.

Maximum Online Insert Rows

The maximum number of rows that can be inserted online in a single process. (Does not apply to Restore Processes). If this value is exceeded, the Insert Process must be executed as a batch job. This value cannot exceed the value specified in **Maximum Insert Rows**.

Maximum Online Delete Rows

The maximum number of rows that can be deleted online in a single process. If this value is exceeded, the Delete Process must be executed as a batch job. This value cannot exceed the value specified in **Maximum Delete Rows**.

Maximum Online Convert Rows

The maximum number of rows that can be converted online in a single process. If this value is exceeded, the Convert Process must be executed as a batch job.

Commit Frequency Rate

The maximum number of rows that can be processed before a commit. This value is displayed on the Specify Parameters and Execute panels for an Insert, Delete, or Restore Process and can be overridden with a lower number. The default value is 1,000.

Administrator Password

The password needed for Administrator privilege. A user has Administrator privilege if the value specified at the **User Supplied Password** prompt on the User Options panel matches this value.

At installation, the Administrator password is set to PSTDB2. It may be changed to any one- to eight-character value. A change to the Administrator Password automatically updates the User Password (of the same value) for the current user.

Trace SYSOUT Job Class

The location where output from the diagnostic trace is to be printed. The trace should be executed only when requested by IBM Technical Support to help diagnose and resolve problems.

Object Security Suffix

The two-character suffix, to be appended to FOP2OS to indicate the name of the exit called for object security. Clear **Object Security Suffix** to disable object security, or change the suffix appropriately to use a different exit.

If you change the object security suffix, it will take effect immediately. The security exit is loaded when a session is invoked or the suffix is changed. (See "Object Security" on page 125.)

Object Security Mode

An indicator for the display and user assignment of security attributes. (This setting is valid only if you are using object security.) Specify:

N   Do not display security attributes. The user cannot assign the security status. Default.

Y   Display security attributes. The user can assign the security status explicitly. Security attributes may be displayed on the Object Attributes panel, selection lists, and object information panels.

Default Security Status

The default security for a new object. (This setting is relevant only if **Object Security Mode** is Y.) Specify:

P   For PUBLIC. Allow any user to access, edit, or delete an object. Default.

R   For READ-ONLY. Allow any user Read-only access to an object.

V   For PRIVATE. Allow only the object owner or administrator to access an object.

Archive Security Suffix

The two-character suffix, to be appended to FOP2AS to indicate the name of the exit called for archive security. For Archive only. To use a different exit, change the number appropriately or, to disable archive security, clear **Archive Security Suffix**.

If you change the archive security suffix, the change takes effect immediately. The security exit is loaded when a session is invoked or the suffix is changed. (See "Archive Security" on page 132.)

Archive Security Mode

An indicator for the display and user assignment of security attributes. (This setting is valid only if you are using Archive security.) Specify:

N   Do not display security attributes. The user cannot assign the security status. Default.

Y   Display security attributes. The user can assign the security status explicitly. Security attributes may be displayed on the Object Attributes panel, selection lists, and object information panels.

Archive Default Security

The default security status to assign to a new Archive File. (This setting is relevant only if **Archive Security Mode** is Y.) Specify:

P   PUBLIC. Allow any user to read, modify, or delete an Archive File. Default.

R   READ-ONLY. Allow any user Read-only access to an Archive File.

V   PRIVATE. Allow only the owner or administrator to access an Archive File.

Functional Security Suffix

The two-character suffix, to be appended to FOP2FS to indicate the name of the exit called for external function security. If blank, functional security is not enabled.

If you change the functional security suffix, the change takes effect immediately. The security exit is loaded when a session is invoked or the suffix is changed. (See "Functional Security" on page 136.)

Temp DASD Unit, Primary

The primary location of temporary work files created during a process. Specify a valid unit name or VIO, which is the default value. Offline tape units cannot be specified. Primary and secondary Temp DASD specifications are validated by performing a test allocation. If neither primary nor secondary allocations can be made during a session, SYSDA is used.

Temp DASD Unit, Secondary

The secondary location of temporary work files. Specify a valid unit name or SYSDA, which is the default value. Primary and secondary Temp DASD specifications are validated by performing a test allocation. If neither primary nor secondary allocations can be made during a session, SYSDA is used.

Default Aging Rule Table

The name of the default Aging Rule Table. This table is used whenever an explicit default Aging Rule Table is not defined for an Insert, Convert, or Load process. The distributed default is FOP2RUSA.

Change Line Characters

Edit the characters used to create borders around pop-ups. Specify:

N   Retain current or default settings for characters used to delineate pop-ups. Default.

Y   Display prompts to change characters used to delineate pop-ups.

Review Del After Archive

Indicator for user review of names of tables marked for deletion after archiving. Specify:

N   Users cannot review tables after archiving and before deleting.

Y   Users can review tables after archiving but before deleting. Default.

Restore Processing Option
   Indicator for user-chosen insert or update during a Restore Process. Specify:

  S   Allow users to choose a Restore processing option. Default.

  I   In a Restore process, all data rows are inserted. Restoring data rows that exist at the destination causes errors.

  U   In a Restore process, rows can be updated or both updated and inserted, according to the **If Update, Process Mode** option.

If Update, Process Mode
   The processing mode when the **Restore Processing Option** is U. Specify:

  B   All data rows are updated if they exist. Rows that do not exist are inserted. Default.

  U   All data rows are updated. Any rows not found are considered errors.

Restore after Table Change
   The action taken when data is restored to a table for which attributes have changed since the data was archived. Specify:

  N   None. Restore proceeds.

  I   An informational message is displayed and the Restore proceeds.

  F   Restore fails.

  U   User is prompted to continue or abort. Default.

Use DB2 LOB Defaults
   The LOB tablespace used by the Create Process when creating an auxiliary table. Specify:

  N   User must create tablespace explicitly.

  Y   Use DB2 default LOB tablespace.

  U   User can specify the type of LOB tablespace on the User Options panel. Default.

Allow Dual Archive Output
   Indicator for writing an Archive File to more than one output. Specify:

  N   Do not allow user to specify more than one output.

  Y   Allow user to specify more than one output. Default.

Tape VOLSER required
   Indicator for a volume serial number for Archive File tape processing. Specify:

  N   No volume serial number is required.

  Y   Volume serial number is required. Default.

Change Archive Logging
   Settings for Archive Logging. Specify:

  N   Do not display the Archive Logging Operations pop-up. Logging settings remain as specified previously. Default.

  Y   Display the Archive Logging Operations pop-up. Use Y to enable logging or N to disable logging for each listed operation.

Remote DB2 Connection
   Indicator for remote connections. Specify:

  A   Application-directed connection (that is, DRDA). The application must issue the **SQL CONNECT** statement to connect to the remote location. Default.

S   System-directed connection (that is, DDF). The system connects to the remote location based on a three-part name or an alias of that name, in which one part names the location.

Replace Arch Dir in Batch

Batch Archive processing when an entry for the Archive File exists. Specify:

N   Halt process with error message.

Y   Proceed with process, replacing Archive File entry in Directory.

U   Enable User Option to determine processing when entry exists. Default.

Target Rows Del Before Isrt

Indicator for delete before Move insert processing. Specify:

N   Users cannot delete target rows before insert or specify commit frequency. These options are omitted from the Specify INSERT Parameters and Execute panel.

Y   Users determine whether to delete target rows before insert, and specify the commit frequency. Profiled values are displayed. Default.

I   Users can determine whether to delete target rows before insert, and can specify the commit frequency. Profiled values are not displayed.

Maximum Data Space Blocks

The maximum size of each data space in 4K blocks. This value can be from 500 through 524,288. The default value is 524,288. If this value exceeds limits imposed by system defaults or the IEFUSI exit, data space creation fails and a system error occurs. (For information on these error codes, refer to your IBM documentation for the DSPSERV macro.)

Maximum Data Spaces

The maximum number of data spaces created by the temp database. This value can be from 1 through 10. The default value is 10. If this value exceeds limits imposed by system defaults or the IEFUSI exit, data space creation fails and a system error occurs. (For information on these error codes, refer to your IBM documentation for the DSPSERV macro.)

Compare Row Contents

Indicator for comparing rows of data in the Source File with rows in the database prior to deletion. Specify:

A   Rows are deleted from the database only if they exactly match rows in the Source File.

N   Row comparison is not performed during the Delete Process. If a table does not have a DB2 primary key or an Optim primary key with a unique index, the Delete Process always compares row contents prior to deletion.

U   User can determine whether rows are compared prior to the Delete Process. Default. If a table does not have a DB2 primary key or an Optim primary key with a unique index, the Delete Process always compares row contents prior to deletion.

Enable Rels Prefetch

Indicator for whether all relationships for a Creator ID are retrieved. Specify:

N   Retrieve only those relationships in which the base table participates.

Y   Retrieve all defined relationships for the specified Creator ID at the first lookup request. Default.

Change Storage Options

Indicator to configure options that allow you to store Archive Files on a Centera networked storage system or Tivoli® Storage Manager backup device. Specify:

N   Do not display the Storage Site Options pop-up. Default.

Y Display the Storage Site Options pop-up. When you close the Storage Site Options pop-up, Change Storage Options reverts to N.

Ignore Generic Rels

Option for ignoring generic relationships to increase performance. Specify:

A Do not use generic relationships to retrieve data.

N Use generic relationships to retrieve data from related tables.

S Use generic relationships to retrieve data, only if the Creator ID is the same for both the parent and child tables.

U Allow user to set a User Option to determine whether generic relationships should be ignored.

Specify Arc File Retention

Indicator for specifying a retention period for Archive Files. You cannot delete or overwrite an Archive File, its associated Archive Directory entry, or any associated Archive Index File until after the specified retention period.

N No retention period is assigned.

Y Assign a retention period to Archive Files, according to the **If Y, Retention Period** option.

U Allow the user to specify a retention period for an Archive File on the Specify ARCHIVE Parameters and Execute panel.

Note: You can alter the retention period assigned to an Archive File, or add a retention period to an existing Archive File, using the **ALTER** statement of the IBM Utility program, IDCAMS, to set the expiration date parameter of the z/OS file. Archive automatically recognizes any change to the retention period.

If Y, Retention Period

The retention period for Archive Files when **Specify Arc File Retention** is Y. If blank, the profiled value is used. If there is no profiled value, a default value of 1Y (one year), is assigned. Specify one of the following:

*n*D The number of days to retain the Archive File.

*n*Y The number of years to retain the Archive File.

yyyy-*mm*-*dd*
An explicit date, after which you can delete or overwrite the Archive File.

Note: Any Archive File assigned a retention period over 9999 days or 27 years is considered permanent and can be deleted or overwritten only if you reduce the retention period using the **ALTER** statement of IDCAMS.

Defer Auth-Check for Batch

Options to defer authorization checks on objects during the creation of JCL and control statements for batch Extract and Archive Processes. Specify:

N Use authorization checks in batch.

Y Defer authorization checks in batch.

U Allow the user to set a User Option to determine whether authorization checks are deferred in batch.

Default Defer DAA

Default value for **Defer Delete After Archive** on the Specify ARCHIVE Parameters and Execute panel. Specify:

N Set the default value to No.

Y   Set the default value to Yes.

Use Uncommitted Reads

Option to extract uncommitted data from the database during an Archive or Extract Process. Specify:

N   Do not extract uncommitted data from the database. Default.

Y   Always extract uncommitted data from the database. If you choose to archive or extract uncommitted data, the relational integrity of the data in the Archive or Extract File may be compromised. Use caution if inserting data from any Archive or Extract File with uncommitted data.

U   User can determine whether uncommitted data is extracted during an Archive or Extract process. The **Archive/Extract With Uncommitted Reads** option is enabled on the Specify ARCHIVE/EXTRACT Parameters and Execute panel.

Allow Mismatched CCSIDs

The action taken when the CCSID of a source column does not match the CCSID for the destination column, or the terminal CCSID does not match that of the DB2 subsystem. Specify:

N   Terminate the process if mismatched CCSIDs are encountered. Default.

Y   Continue the process, which will fail if CCSIDs are incompatible.

U   The User Option **Allow Mismatched CCSIDs** determines the action taken. See **Allow Mismatched CCSIDs** in the *Common Elements Manual*.

Generate USEDD Parm in JCL

The action taken in a batch Insert, Restore, Load, Convert or Delete process when the USEDD keyword is encountered. Specify:

N   Do not generate DD cards. Default.

Y   Generate the DD cards for the file.

Legacy MIXED Data

The use of code points X'0E' and X'0F' as shift-out and shift-in controls for double-byte character strings in legacy data. Specify:

Y   Character strings can be single-byte character set (SBCS) or MIXED data with X'0E' and X'0F' serving as shift-out and shift-in controls. The CCSID that is specified in the Legacy EBCDIC CCSID field must be the mixed CCSID for the encoding scheme. During processing, this value determines the associated SBCS and double-byte character set (DBCS) CCSIDs for the encoding scheme. A valid entry in **Legacy EBCDIC CCSID** is required.

N   All character strings in Legacy data are processed as SBCS data.

U   Legacy MIXED DATA and Legacy EBCDIC CCSID values are obtained from the Legacy Options panel, available to all users.

Legacy EBCDIC CCSID

The default CCSID for EBCDIC-encoded character data in your Legacy data. Required if **Legacy MIXED DATA** is set to 'Y'. Specify a value from 1 - 65533.

New Rels Warning in Batch

Behavior when relationships are found to be in NEW status during a batch Extract or Archive process. Specify:

N   No message or return code is produced when relationships are in NEW status.

Y   A message and return code 4 is produced when relationships are in NEW status.

Enable IMS I/O Exit

Indicator for use of an IMS I/O Exit when a segment is read or written. Specify:

N   No I/O exit is used. Any exit specified in **Exit Name** and in Legacy Table definitions is ignored. Default.

Y   The exit specified in **Exit Name** is called when an IMS segment is read or written. Any exit specification in a Legacy Table definition is ignored.

U   An exit specified in a Legacy Table definition, if any, is called. If the Legacy Table definition does not reference an exit, the exit specified in **Exit Name** is called.

Exit Name
The name of the default I/O exit. FOPIMIOX, the exit distributed with your Optim solution, is specified by default.

Enable VSAM/Seq I/O Exit
Indicator for use of an I/O Exit when a VSAM or sequential record is read or written. Specify:

N   No I/O exit is used. Any exit specified in **Exit Name** and in Legacy Table definitions is ignored. Default.

Y   The exit specified in **Exit Name** is called when a record is read or written. Any exit specification in a Legacy Table definition is ignored.

U   An exit specified in a Legacy Table definition, if any, is called. If the Legacy Table definition does not reference an exit, the exit specified in **Exit Name** is called.

Exit Name
The name of the default I/O exit. FOP2DIOX, the exit distributed with your Optim solution, is specified by default.

Resolve DB2 Rel Change
For batch execution of extract and archive processes, determines whether Optim rebuilds any DB2 relationship if the columns in the relationship have changed since the time when the relationship was originally defined. Specify:

Y   Yes. Rebuild a DB2 relationship if the columns have changed.

N   No. Do not rebuild a DB2 relationship if the columns have changed. If an extract or archive process encounters a relationship with columns that have changed, it will fail with a return code of 12. This is the default

U   User. Allow the user to specify whether or not a DB2 relationship is rebuilt if the columns have changed.

Register DASD Extract File
Allow users to register extract files stored on disk in the Optim directory. Specify:

Y   Yes. Allow user to register an extract file.

N   No. Do not allow extract files stored on disk to be registered in the Directory. This is the default.

U   User. Enable user option to allow the user to register an extract file on disk in the Directory.

Replace Extr Dir in Batch
Batch extract file processing when an entry for this extract file exists in the Directory. Specify:

Y   Yes. Allow processing to continue and replace any existing extract file entry.

N   No. Halt processing with an error message. This is the default.

U   User. Allow user the option to enable user to determine processing when there is an existing Directory entry for the extract file.

Require IMS Data Set Names

> Controls whether or not the user is required to supply the name of an IMS database data set when creating a Retrieval Definition.

> **Y**   Yes. User must supply IMS data set name. This is the default.

> **N**   No. User is not required to supply an IMS data set name. If the field for IMS data set name is left blank, IMS dynamically allocates the data set.

> **Note:** Regardless of the setting for this option, all users can enable IMS to dynamically allocate the data set by specifying '$MDA' as the data set name.

Change Optim Data Privacy

> Determines the environment variable file used for data privacy providers:

> **Y**   Yes. Set a site default environment variable file that applies to all users. The Site Data Privacy Options panel displays and prompts you to enter the name of the data set for the Unix System Services environment variables.

> **N**   No. No default environment variable file is set.

> **U**   User. Allow users to specify the name of the environment variable file on the User options panel.

Specifying Storage Options

Specifying `Y` for **Change Storage Options** displays the Storage Site Options pop-up, allowing you to specify information for your storage site (that is, Centera networked storage system or Tivoli Storage Manager backup device).

```
+--------------------------Storage Site Options--------------------------+
¦                                                                        ¦
¦ Allow Orphaned Files          ===> N     Y-Yes, N-No                   ¦
¦ Recall Processing             ===> U     U-User,P-Prompt,A-Auto,B-Abort¦
¦                                                                        ¦
¦ Centera Options:                                                       ¦
¦                                                                        ¦
¦ Modify Centera Pools          ===> N     Y-Yes, N-No                   ¦
¦ Minimum Retention             ===> N     D-Dflt,I-Intv,F-Inf,N-None,U-User¦
¦  If User or Interval, Years   ===> 0     0-100                         ¦
¦                    Days       ===> 30    0-18300                       ¦
¦                                                                        ¦
¦ Tivoli Options:                                                        ¦
¦                                                                        ¦
¦ Modify Node Definitions       ===> N     Y-Yes, N-No                   ¦
¦ User Name                     ===> TIVOLI1                             ¦
¦ Maintain Archive Collections ===>                                      ¦
¦ Path Name                     ===> \home\path\abc                      ¦
¦                                                                        ¦
¦        Enter EXPand PATH Command to Edit Long Path Name                ¦
+------------------------------------------------------------------------+
```

Figure 18. Storage Site Options

The Storage site options are:

Allow Orphaned Files

> User permission to delete an Archive Directory entry, when the associated Archive File is on a storage site. Specify:

> **N**   Prevent a User from deleting an Archive Directory entry if the associated Archive File cannot be deleted from the storage site. Default.

> **Y**   Allow a User to delete an Archive Directory entry, even if the associated Archive File cannot be deleted from the storage site. Using this setting may cause Archive Files copied

to a storage site to be "orphaned", that is, the Archive Directory entry is deleted and the Archive File on the storage site is inaccessible for Archive processing.

Recall Processing

Management of Archive Files that are recalled from a storage site in Archive processing (for example, browse or restore). Specify:

U Allow user to set a User Option for recall of Archive Files.

P Display an allocation panel when the Archive File is about to be recalled.

A Recall an Archive File to the original location, with no user intervention.

B Abort the request if the original Archive File is no longer on disk.

Centera Options

The following options apply to Centera networked storage systems only.

Modify Centera Pools

Setting for display of the Centera Pool Name List. A Centera Pool Name refers to a group of nodes that correspond to a Centera Server. Specify:

N Do not display the Centera Pool Name List. Default.

Y Display the Centera Pool Name List pop-up, listing any existing Centera Pool Names and create or modify Centera Pool Names. When you close the Centera Pool Name List pop-up, the **Modify Centera Pools** setting reverts to N.

Minimum Retention

The period for which an Archive File is retained on the Centera Server before it can be deleted. Specify:

D Use the default minimum retention period, based on your Centera configuration.

I Protect an Archive File from deletion for a specified period. Once selected, you must specify a number of years, days, or a combination of both.

F Keep an Archive File on Centera forever; the file cannot be deleted.

N Do not use a minimum retention period. An Archive File can be deleted from Centera at any time.

U Allow the user to determine the minimum retention period when the Archive File is created.

If User or Interval,

If Minimum Retention is I for Interval, Years and Days determine the minimum retention period. If Minimum Retention is U for User, Years and Days determine the maximum values the user can specify. The combined total of years and days must be greater than 0.

Years The number of years to protect an Archive File from deletion. Specify a number from 0 through 100.

Days The number of days to protect an Archive File from deletion. Specify a number from 0 through 18,300.

Tivoli Options

The following options only apply to Tivoli Storage Manager backup devices.

Modify Node Definitions

Setting to display the Tivoli Node List. Specify:

N Do not display the Tivoli Node List. Default.

Y   Display the Tivoli Node List pop-up, listing any existing Tivoli Node Names. When you close the Tivoli Node List pop-up, the **Modify Node Definitions** setting reverts to N.

User Name
The default user name for the Tivoli Node Definitions.

Password
The default password for the Tivoli Node Definitions.

Path Name
The default path name for the Tivoli Node Definitions. If the path name is longer than 38 characters, you must use the **EXP PATH** command to open the Enter Path Definition panel, where you can specify the full path name.

Note: The User, Password, and Path Name are used only if that information is not entered for a specified Tivoli Node Name Entry. For more information, see "Tivoli Node Entry" on page 64.

Centera Pool Name List

Specifying Y for **Modify Centera Pools** displays the Centera Pool Name List pop-up.

```
     +--------------------Centera Pool Name List-------------------+
     |                                                             |
     | Line Commands: I-Insert, D-Delete, S-Select, U-Update, P-Ping |
     |                                                             |
     | Cmd  Pool Name       Description                            |
     | ---  ---------  ----------------------------               |
     | ___  CENT_R     Centera R (no Del/write) PEA               |
     | ___  SECOND     Centera R/W (no delete) PEA                |
     | ___  TEST       Test Pool                                   |
     |                                                             |
     |                                                             |
     |                                                             |
     | Enter END or CANCEL command to return to prior panel        |
     +-------------------------------------------------------------+
```
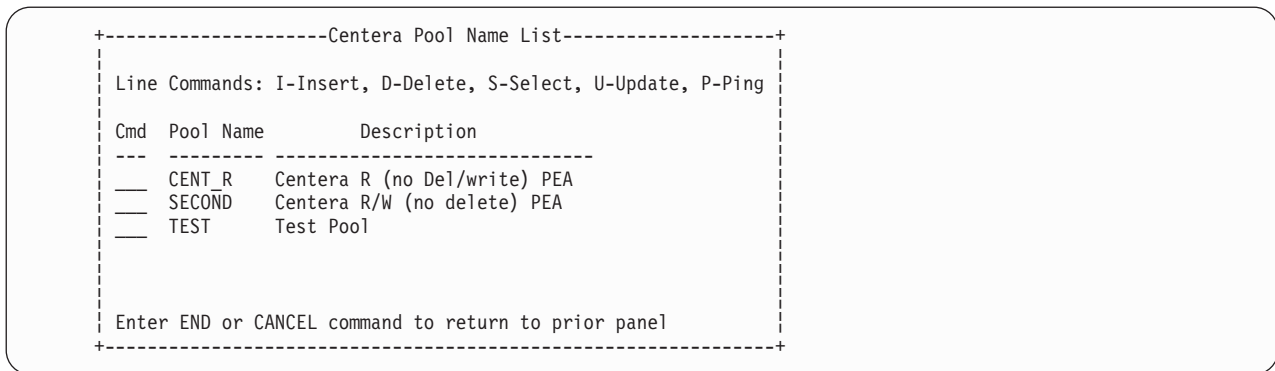
Figure 19. Centera Pool Name List

The Centera Pool Name List pop-up displays any available Centera Pool Names, and allows you to add, remove, or modify up to eight Pool Names. This list includes the following prompts:

Cmd   Area for line commands. Valid line commands are:

        I   Display the Centera Pool Name Entry pop-up to insert a new Centera Pool Name.

        D   Delete an existing Centera Pool Name. If you delete a Centera Pool Name, you cannot recall any Archive Files saved to the Centera Server using that pool name.

        S   Select an existing Centera Pool Name, displaying it in the Centera Pool Name Entry pop-up.

        U   Update a Centera Pool Name. The Centera Pool Name Entry pop-up is displayed, allowing you to modify the Pool Name.

        P   Ping the nodes in a Centera Pool to validate connections. The Process Status pop-up is displayed while the connections are validated. When the process is complete, the Centera Pool Node Report is displayed, indicating the connection results for each node and statistics for nodes with a successful connection.

Pool Name
The 1 to 8 character Centera Pool Name.

Description
The 1 to 30 character description of the Centera Pool Name.

Centera Pool Name Entry

The Centera Pool Name Entry pop-up allows you to create or modify a Centera Pool Name.

```
+----------------------Centera Pool Name Entry----------------------+
|                                                                   |
| Pool Name           ===> FOPBOX                                   |
| Description         ===> Our Centera Box                          |
| Authorization File ===> FOPEJG.CENTERA.PEA.FILES.RWDQEM           |
|                                                                   |
| Node List:                                                        |
|                                                                   |
|   ===> 172.16.6.60                                                |
|   ===> 172.16.6.62                                                |
|   ===> 172.16.6.63                                                |
|   ===>                                                            |
|   ===>                                                            |
|   ===>                                                            |
|   ===>                                                            |
|   ===>                                                            |
|                                                                   |
| Enter EXPand Primary Command to Edit Long Authorization File Names |
| Enter END or CANCEL Command to Return to Previous Panel           |
+-------------------------------------------------------------------+
```

Figure 20. Centera Pool Name Entry

Specify the following:

Pool Name
> The 1 to 8 character Pool Name.

Description
> A 1 to 30 character description for the Centera Pool Name.

Authorization File
> If the site requires a PEA file for access to the Centera Server, enter the name of the Centera Pool
> Entry Authorization File.

Node List
> The DNS names or IP addresses required to access the Centera Server. Enter each DNS name or
> IP address on a separate line following the prompts (===>) displayed under this heading, such as
> ===> 172.16.6.60.
>
> You can specify as many as eight DNS names or IP addresses.
>
> **Note:** DNS Names or IP addresses are not validated.

After you have created or modified a Centera Pool Name entry, use END to save it and redisplay the
Centera Pool Name List pop-up.

Centera Pool Node Report

After the P line command is entered from the Centera Pool Name List panel, the Centera Pool Node
Report displays the connection results for each node in the specified pool and statistics for nodes with a
successful connection.

```
------------------------ Centera Pool Node Report --------------------------
 Command ===>                                              SCROLL ===> CSR
                                                           ROW 0    OF 31
******************************* Top of Data **********************************
Connection Test of Pool TEST Performed 2006-06-11-15.19.21

Node: 199.55.6.66          Connection Test: FAILED
                           No connection with any pool

Node: 199.55.6.62          Connection Test: PASSED
Pool Info Version:         2
Pool Capacity:             0x000004487C000000
Pool Freespace:            0x000005630B57E76F
Pool ClusterID:            cb06djjj2-1bb1-11h2-b9ac-e808djjed9e7
Pool Clustername:          Site
Pool Software Version:     2.0.1.169-1014
Pool Replica Address:
Pool Buffer Size:          16384
Pool Timeout:              120000
Pool Failover:             1
Pool Collision Avoidance:  0
Pool Maxconnections:       100
Pool Retrycount:           6
Pool Retrysleep:           100
READ Capability:           true
WRITE Capability:          true
WRITE DUP Capability:      false
PURGE Capability:          true
DELETE Capability:         true
EXIST Capabilty:           true
CLIPNAME...Capability:     true
RETENTION Capability:      0
BLOBNAMING Capability:     MD5,MG
Client SDK Version:        3.1.477
***************************** Bottom of Data *********************************
```

Figure 21. Centera Pool Node Report

Tivoli Node List

Specifying Y for **Modify Node Definitions** displays the Tivoli Node List.

```
+-----------------------Tivoli Node List----------------------+
|                                                             |
| Line Commands: I-Insert, D-Delete, S-Select, U-Update, P-Ping |
|                                                             |
| Cmd    Name                    Description                  |
| ---   --------    --------------------------------------    |
| ___   TIVOLI1    Main Tivoli Server                         |
| ___   AIXTIV1    Aix Tivoli Server                          |
|                                                             |
|     Enter END or CANCEL command to return to previous panel |
+-------------------------------------------------------------+
```

Figure 22. Tivoli Node List

Specify the following:

Cmd The line commands area. Valid line commands are:

I   Insert a new Tivoli Node Name by displaying the Tivoli Node Entry pop-up.

D   Delete an existing Tivoli Node Name. If you delete a Tivoli Node Name, you cannot
recall any Archive Files saved to the Tivoli Server using that node name.

S   Select an existing Tivoli Node Name, displaying it in the Tivoli Node Entry pop-up.

U   Update a Tivoli Node Name. The Tivoli Node Entry pop-up is displayed, allowing you to
modify the Node Name.

P   Validate the connection. The Process Status pop-up is displayed while the connection is validated. When the process is complete, the Tivoli Node Report indicates the connection results.

Name   The 1 to 64 character Tivoli Node Name. Only the first 8 characters of the name are displayed.

Description
        The 1 to 30 character description of the Tivoli Node Name.

Tivoli Node Entry

The Tivoli Node Entry pop-up allows you to create or modify a Tivoli Node Name.

```
+----------------------Tivoli Node Entry----------------------+
|                                                             |
|                                                             |
|            Primary Commands: EXPand NODE/PATH/OFILE/MFILE   |
|                                                             |
| Node Name    ===>                                           |
| Description  ===>                                           |
| User Name    ===>                                           |
| Password     ===>                                           |
| Path Name    ===>                                           |
| Option File  ===>                                           |
| Mgmt. File   ===>                                           |
| Filespace    ===>                                           |
| Mgmt. Class  ===>                                           |
|                                                             |
|       Enter END or CANCEL command to return to previous panel |
+-------------------------------------------------------------+
```
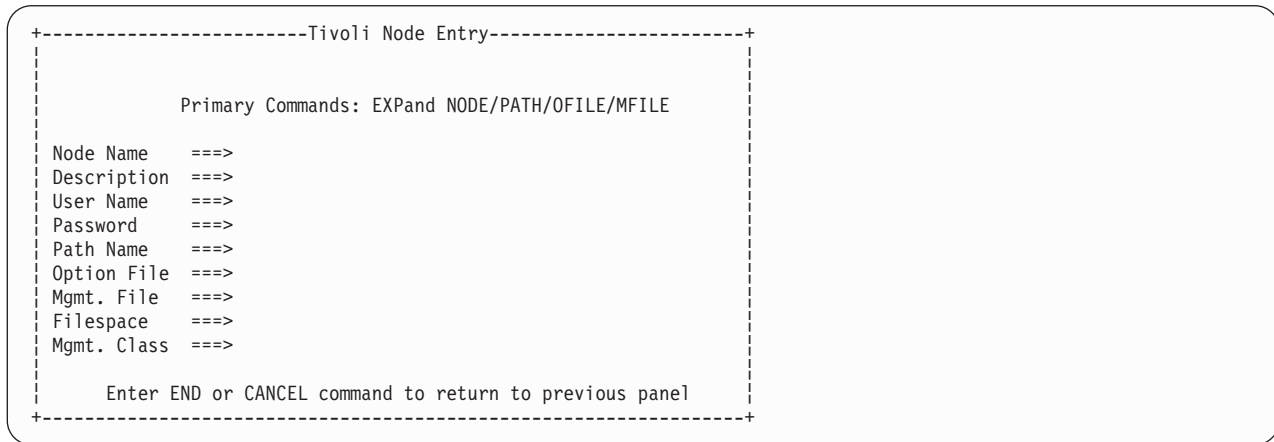
Figure 23. Tivoli Node Entry

Specify the following:

Node Name
        The 1 to 64 character Node Name. Use the EXPAND primary command to enter a long node name.

Description
        A 1 to 35 character description of the Tivoli Node Name. Optional.

User Name
        The 1 to 35 character user name for the Tivoli Server.

Password
        The password associated with the User name.

Path Name
        The fully-qualified path name for the Tivoli Server.

Option File
        The fully-qualified path name for the Option File for the Tivoli Server.

Mgmt. File
        The fully-qualified path name for the Management File for the Tivoli Server.

Filespace
        The name of the file space to be associated with the Archive Files saved to the Tivoli Server.

Mgmt. Class
        The name of the management class object to be associated with Archive Files saved to the Tivoli Server.

When you have created or modified a Tivoli Node entry, use END to save it. The Tivoli Node List pop-up is redisplayed.

If the fully-qualified name for the path name, option file, and management file is longer than 46 characters, use the **EXPand** primary command with one of the following parameters:

NODE   Open the Enter Node Definition panel, allowing you to specify the 1- to 64 character Node Name.

PATH   Open the Enter Path Definition panel, allowing you to specify the fully-qualified path name for the Tivoli Server.

OFILE Open the Option File Definition panel, allowing you to specify the fully-qualified path name of the Option File for the Tivoli Server.

MFILE Open the Enter Management File Definition panel, allowing you to specify the fully-qualified path name of the Management File for the Tivoli Server.

Accessing the Site Options Panel

During product execution, administrators can access the Site Options panel either from the Choose Option Type menu or using the SITE command on the User Options panel.

During product execution, the site options can be modified only by users who are administrators. (Anyone who successfully enables or extends the product license during the installation procedure is assumed to be an Administrator.) Administrators are identified by a password that matches the **Administrator Password** specified on the Site Options panel.

If the Site Options panel is not accessible from the Choose Option Type menu or the User Options panel, you must first select Option 0.1 User Options and supply the current password in **User Supplied Password** on the User Options panel. If this password matches the **Administrator Password**, the Site Options panel is accessible.

Security Considerations for Optim

Optim does not require any special consideration for securing files. In order to perform an administrator function, a user must have update authority on the load library. The **Administrator Password** is not intended as a security measure. It is used to prevent users from viewing options they are not allowed to update.

Optim respects all DB2 and z/OS security including all authorizations and access privileges. Information about applying security to the Optim functions and the objects stored in the Directory (Access Definitions, Compare Definitions, Relationships, Column Maps, and Table Maps) is discussed in Chapter 11, "Establish Security," on page 125.

Set Up Optim Directory and Sample Database

Option 11 on the Full Installation menu populates the Optim Directory with sample object definitions needed to set up the sample database.

A sample database to demonstrate the product capabilities and to support training is provided with Optim. The sample database consists of a set of sample tables with a number of typical RI rules defined. The Extract File for the sample database is included with the installation files. This file was created using code page 37. If your target subsystem uses a different code page, the Site Option for **Allow Mismtched CCSIDs** must be set to **Y**, or attempting to load the data will fail. See "Customizing the Site Options" on page 49.

The Setup Directory and Sample Database menu lists the available options. You can select each option, or press ENTER to proceed through the options in the order listed on the menu.

```
------------  Setup Optim Directory and Sample Database  -------------------
OPTION  ===>

Setting up the Optim Directory and Sample Database consists of the
following steps:

     1 - Create/Load the Sample Database
     2 - Load Sample Access Definitions and Column Maps
     3 - Create Copies of DB2 Relationships
     4 - Create OPTIM Primary Keys for all Unique Indices
     5 - Create Legacy VSAM Sample Files
     6 - Load Legacy VSAM Sample Data
     7 - Create Legacy IMS Sample Databases
     8 - Load Legacy IMS Sample Data



**************************** N O T E S ********************************
*  The authorization levels required to complete these functions vary.  *
*  Detailed requirements are noted with the function specification.      *
*************************************************************************
```

The following options are available:

1. Create/Load the Sample Database

   Creates the distributed sample database and loads the sample data. For more information, see "Create/Load Sample Database" on page 91.

2. Load Sample Access Definitions and Column Maps

   Loads the distributed sample Access Definitions, Column Maps, and primary keys into the Optim Directory. For more information, see "Load Samples" on page 99.

3. Create Copies of DB2 Relationships

   Scans the DB2 Catalog for relationship information and loads it into the Optim Directory. For more information, see "Create Copies of DB2 Relationships" on page 101.

4. Create Optim Primary Keys for all Unique Indices

   Scans for tables that do not have DB2 primary keys. If any of these tables has a unique index, a primary key is loaded into the Optim Directory for that table. For more information, see "Create Optim Primary Keys" on page 102.

5. Create Legacy VSAM Sample Files

   Creates VSAM datasets for Optim Legacy sample files, FOPDEMO.VENDITEM and FOPDEMO.VENDOR. For more information, see "Create Legacy VSAM Sample Files" on page 104.

6. Load Legacy VSAM Sample Data

   Loads sample files and objects (Legacy Tables, primary keys and relationships) for Optim Legacy. For more information, see "Load Legacy VSAM Sample Data" on page 105.

7. Create Legacy IMS Sample Databases

   Creates the IMS datasets (FOPDEMO.FOPDEPDB, FOPDEMO.FOPDEPIX, FOPDEMO.FOPJOBDB, and FOPDEMO.FOPJOBIX) for Optim Legacy. For more information, see "Create Legacy IMS Sample Databases" on page 105.

8. Load Legacy IMS Sample Data

   Loads IMS sample files and objects (Legacy Tables, Environment Definitions, and Retrieval Definitions) for Optim Legacy. For more information, see "Load Legacy IMS Sample Data" on page 106.

These functions are also available from the main menu, Option 3 Maintain Directory and Sample Database. For details about these functions, see Chapter 6, "Directory Maintenance," on page 89.

Customize the Batch Environment

Use Option 12 on the Full Installation menu to customize the distributed skeletons, which are used to generate JCL and control statements needed to execute the batch processes performed by Move, Compare, and Archive. Option 12 also allows you to specify additional batch STEPLIB libraries.

Further information about customizing the JCL skeletons is provided in "Customizing the JCL Skeletons" on page 120.

The following panel prompts for several parameters required by Optim to support the JCL skeletons. The default values are presented on the initial panel display. You can modify these values, as needed.

```
----------  Supply Optim Batch Parameters ------------------------------------
COMMAND ===>

Product Skeleton Library  ===> 'FOPUSR.REL113.SFOPSENU'
Product Load Library      ===> 'FOPUSR.REL113.SFOPLLIB'
Enter Additional Batch STEPLIB Libraries ===> YES (Y-Yes  N-No)
  such as  non-LNKLST DB2 Libraries and/or Alternate Utility Libraries

Batch Region Size ===> 0M    (0-2047 megabytes; 0M requests Max Region Size)

Permanent Dataset Defaults:   Temporary Dataset Defaults:  Tape Defaults:
 Unit Type    ===> SYSDA       Unit Type    ===> SYSDA      MaxUnit ===> 1
 Primary CYLS ===> 3           Primary CYLS ===> 3

Utility Block Sizes: Load Input File ===> 0        Image Copy File ===> 0
Use SMS for New Load and Image Files ===> Y        (Y - Yes  N - No)
Image Copy Dataset Name Suffixes:  (1-4 characters)
  COPYDDN Backup ===> BKUP  RECOVERYDDN ===> RECV RECOVERYDDN Backup ===> REBK


******************************** N O T E S ********************************
*          You must have authority to update the Skeleton Library        *
*          to complete this function successfully.                       *
**************************************************************************
```

Figure 24. Supply Batch Parameters

The following prompts are presented:

Product Skeleton Library
> The dataset name of the skeleton library.

Product Load Library
> The dataset name of the load library.
>
> **Note: Product Load Library** and **Product Skeleton Library** must contain the values specified for the installation and in the FOPCUST CLIST.

Enter Additional Batch STEPLIB Libraries
> Additional batch STEPLIB libraries. For load libraries that are not in the linklist or for alternate utility libraries, specify:
>
> **Y**   Display a panel to specify the libraries. (See "Additional Batch STEPLIB Libraries" on page 68.)
>
> **N**   Do not display the panel.

Batch Region Size
> The batch region size in megabytes. Specify 0M to request the Max Region Size.

Permanent Dataset Defaults
> Default values for permanent datasets. Specify:

Unit Type
> Default is SYSDA.

Primary CYLS
> Default is 3 for 3 cylinders.

Temporary Dataset Defaults
> Default values for temporary dataset allocation. Specify:

Unit Type
> Default is SYSDA.

Primary CYLS
> Default is 3 for 3 cylinders.

Tape Defaults
> The number of tape drives, from 1 through 255, assigned to the DD name allocated for these drives in an Archive batch job.

MaxUnit
> Default is 1 for 1 unit.

Utility Block Sizes
> The block sizes for the datasets created as part of a Load Process.

Load Input File
> Default is 0.

Image Copy File
> Default is 0.

Use SMS for New Load & Image Files
> Indicator for using SMS when load and image files are created.

Y   SMS is used. Default setting.

N   SMS is not used. (**Utility Block Sizes** must be greater than zero.)

Image Copy Dataset Name Suffixes
> The suffix used for the Image Copy datasets created as part of a Load Process. The suffix may contain from one to four characters. Specify the suffix for:

COPYDDN Backup
> The default is BKUP.

RECOVERYDDN
> The default is RECV.

RECOVERYDDN Backup
> The default is REBK.

Additional Batch STEPLIB Libraries

Use the Supply Additional Batch STEPLIB Libraries panel to enter additional batch STEPLIB libraries, such as non-LNKLST DB2 Libraries or Alternate Utility Libraries.

If you specify Yes for **Enter Additional Batch STEPLIB Libraries**, the following panel is displayed.

```
---------- Supply Additional Batch STEPLIB Libraries ----------------------
COMMAND ===>

Product Load Library         : FOPUSR.REL113.SFOPLLIB
    Site Column Map Load     ==>

DB2 Libraries (required if not included in LNKLST):
   DB2 User Load Library    ==>
   DB2 Exit Load Library    ==>
   DB2 Load Library         ==>
   DB2 Template Library     ==>

Alternate Utility Libraries:
   Utility Load Library     ==>
   If BMC Utilities:
     SMS Service Routine     ==>

******************************** N O T E S ********************************
*           You must have authority to update the Skeleton Library        *
*           to complete this function successfully.                       *
**************************************************************************
```

Figure 25. Supply Additional Batch STEPLIB Libraries

This panel includes:

Product Load Library

The data set name for the Product Load Library, specified on the Supply Batch Parameters panel, is entered automatically.

Site Column Map Load DSN

The name of the data set to contain the Column Map exit programs, if Column Map exit programs are to be used.

DB2 Libraries

The data set names of the following load libraries when they are not included in the linklist:

DB2 User Load Library DSN
DB2 Exit Load Library DSN
DB2 Load Library DSN
DB2 Template Library DSN

Alternate Utility Libraries

The data set name of the load library for a load or unload utility, other than DB2:

Utility Load Library DSN

The data set name of the following load library, if using a BMC utility (that is, LOADPLUS or UNLOAD PLUS):

SMS Service Routine DSN

For all entries, be sure to specify the correct libraries.

Dataset names supplied on the Supply Additional Batch STEPLIB Libraries panel are added to the STEPLIB concatenation and included in the JCL generated for batch jobs.

Chapter 5. Upgrade to a New Release

To upgrade your version of Optim, select Option 2 from the main Install menu. Use this option to upgrade to the current release of Optim, to install any component not already installed, or if you have tested the new release in a separate environment.

The upgrade updates the software, as well as the Optim Directory tables. See "Remote Subsystem Installation Overview" on page 3 for guidelines on when to perform an upgrade instead of a full install.

Notes:

- Sites using Optim products earlier than release 7.1 must perform a full install to migrate to release 11.3. See Chapter 4, "Full Install," on page 17.
- If you are using release 7.1 or later, use the instructions in this section to upgrade to release 11.3.

```
---------- Installation Notes for Upgraded Site Options Module Format ----
OPTION ===> _____

 All existing site options modules must be upgraded and copied through the
 Installation Interface for use with Optim 11.3.0.

 Use option 7, Copy and Upgrade Site Options from Previous Release, of the
 Upgrade Path to upgrade your default site options module.  Use option 10,
 Customize Optim Site Options, from the install main menu to
 upgrade any other customized site options modules for use with Optim 11.3.0.
```

Select Option 2 to display the Upgrade menu.

```
----------  Upgrade Optim ------------------------------
OPTION  ===>                                    More:   +

The upgrade consists of the following steps:

     1 - Export Optim Directory Objects (Optional)
     2 - Bind New Plan for Product Upgrade
     3 - Specify Optim Directory to Upgrade
     4 - Upgrade Directory Tables
     5 - Bind Plan for Product Execution
     6 - Unload Optim Sample Data and ODM Optim Connect
     7 - Copy and Upgrade Site Options from Previous Release
     8 - Maintain Optim Directory and Sample Database
     9 - Customize the Batch Execution Environment

  DB2 Subsys to Upgrade  ===>
  DB2 Release            ===>        (n.n.n)
  Secondary SQLID        ===>
  Load Library DSN       ===>
  Install Library DSN    ===>

  Previous release
  Load Library DSN       ===>
```

Figure 26. Upgrade Optim

This menu lists the steps required to perform the upgrade. You can let the installation procedure automatically prompt you through each step in the order in which the steps are listed by simply pressing ENTER each time this panel is redisplayed. Using this method, the upgrade process can be suspended and restarted later without having to repeat any steps. The upgrade process restarts from the point where you suspended it, provided the same TSO User ID and the same DB2 subsystem are used.

Option 9 is available when upgrading or installing Move, Compare, or Archive.

In addition to selecting an option, you are prompted to specify certain parameters:

DB2 Subsys to Upgrade
> The DB2 subsystem that is being upgraded. The upgrade must be executed for each subsystem in which Optim is used.

DB2 Release
> Number of the DB2 release. Enter the correct release number for the DB2 subsystem where you are installing Optim. Optim performs the installation as appropriate for the target DB2 subsystem.
>
> The release you specify is checked against the DB2 subsystem. If there is a discrepancy, you are prompted to confirm the release number to ensure that you have specified the desired release.

Secondary SQLID
> Optional name of a secondary SQLID to be used for the upgrade process. If this is blank, your primary SQLID will be used.

Load Library DSN
> Fully qualified name of the load library (SFOPLLIB) created when Optim was installed.

Install Library DSN
> Fully qualified name of the dataset (SFOPINST) created when Optim was installed.

These parameters will contain the names entered on the Install menu. You can modify the values on this panel.

To upgrade Optim, your TSO session must have the capability of connecting to the DB2 subsystem you have specified.

Only one copy of the product libraries is necessary regardless of how many subsystems you are upgrading.

The following discusses the series of panels in the order in which they are presented when every option is selected to upgrade Optim.

Note: You can execute a prior release of Optim concurrently with the new release. For details, see Appendix C, "Run Multiple Releases," on page 199.

Export Directory Objects

To export the Optim Directory objects to a sequential file, use Option 1 on the Upgrade panel. This option allows you to backup the Directory before it is upgraded and identify errors in the Directory before they are propagated to the new release.

Note: Before exporting, you must disable Optim security under your current installation of Optim.

Use the **Summary File Name** to identify errors in the Directory. If errors exist, you can correct them in the Export File (**Output Dataset**) or correct them in the Directory and run the Export Process again. For information about importing objects into the Directory, see Use the Import Process section in the *Common Elements Manual*.

```
--------------------    Export Optim Directory Objects -----------------------
OPTION  ===>

The following objects will be exported from the Optim directory:

    AD - Access Definitions        PK - Primary Keys
    R  - Relationships,            TM - Table Maps
    CD - Compare Definitions       CM - Column Maps
    ED - Environment Definitions   RD - Retrieval Definitions
    LT - Legacy Tables

 This task must be performed using the load library and plan
 bound with the prior release from which you are upgrading.

 Load Library Name      ===>
 Plan Name              ===>
 Output Dataset         ===>
 Dataset Disposition    ===> O   (M-Mod, O-Old)
 Enter Directory Report File Name or Leave Blank for SYSOUT *
 Dir Report File Name  ===>
 Enter Report File Name or Leave Blank for SYSOUT *
 Report File Name  ===>
 Enter Summary Report File Name or Leave Blank for SYSOUT *
 Summary File Name      ===>
 On Export Error        ===> S   (C-Continue, S-Stop)
(Selecting 'C' will allow the upgrade process to continue if errors
 are encountered during the EXPORT process.  Use this option carefully.)
```

Figure 27. Export Directory Objects

Load Library Name
> Fully qualified name of the load library used by the release you are upgrading from.

Plan Name
> The plan name you are upgrading from.

Output Dataset
> The name of a sequential data set for the exported objects--the Export File.

Dataset Disposition
> Placement of exported objects copied to an existing data set. Specify:
> - M for MOD. The object data is appended to the data set.
> - O for OLD. The contents of the data set are replaced.

Dir Report File Name
> The name of a sequential data set that lists all exported objects. If you leave this prompt blank, the output is directed to a SYSOUT class, and you can use an output processor, such as SDSF, to display and print it.

Report File Name
> The name of a sequential data set that lists general error information. If you leave this prompt blank, the output is directed to a SYSOUT class, and you can use an output processor, such as SDSF, to display and print it.

Summary File Name
> The name of a sequential data set that lists the processed EXPORT syntax along with errors. If you leave this prompt blank, the output is directed to a SYSOUT class, and you can use an output processor, such as SDSF, to display and print it.

On Export Error
> Indicator for whether the EXPORT process should continue if an error is encountered within a directory table. Specify:
> - C for Continue. The EXPORT process will continue and subsequent directory tables will be processed.

Note: Use this option carefully as objects reported to be in error will not be exported.

- S for Stop. The EXPORT process will terminate and subsequent directory tables will not be processed.

When you run the Export, the batch syntax for the Export Process is reproduced in the Process Report; for example:

```
ERROR CONT
EXPORT TYPE ALL SUBDEF YES NAME %
SECURITY_VIOLATION WARN
```

See the *Batch Utilities Guide* for a detailed explanation of this syntax.

Bind Plan for Product Upgrade

To perform a bind of the new plan for product upgrade, use option 2 on the Upgrade panel. If you use the same plan name for the upgrade, you will disable the current version of the product.

The following panel is displayed.

```
---------------------- Bind Plan for Product Upgrade ---------------------
COMMAND ===>

Plan Owner                    :
Plan Name                 ===> FOP1130
DBRM Library DSN          ===>

Review DSN subcommand prior to execution ===> NO (Y-Yes, N-No)

Enter Options:
  Bind Action              ===> REPLACE    (Add, Replace)
  Retain Execution Authority ===>          (Y-Yes, N-No)
  Isolation Level          ===> CS         (CS, RR)
  Validate                 ===> BIND       (Run, Bind)
  Resource Acquisition Time ===> USE       (Use, Allocate)
  Resource Release Time    ===> COMMIT     (Commit, Deallocate)
  Explain Plan Selection   ===> NO         (Y-Yes, N-No)
  Bind Message Level       ===> ALL        (All, Warning, Error, Complete)
  Data Currency            ===> YES        (Y-Yes, N-No)

******************************* N O T E S *******************************
*     The plan owner must have BINDADD privilege or SYSADM authority    *
*     to complete this function successfully.                          *
************************************************************************
```

Figure 28. Bind Plan for Upgrade

A similar panel is displayed when initially installing Optim. For details about this panel, see "Bind the Plan for Product Installation" on page 18.

Specify Directory to Upgrade

To select the Directory to be upgraded, use Option 3 on the Upgrade panel.

The following panel is displayed.

```
----------    Specify Directory to Upgrade  ---------------------------------
COMMAND ===>

Enter the Authorization ID of the Optim Directory:

      Authorization ID ===> SYSFOP

Press ENTER without an AuthID for a Selection list of Directory ID Candidates


********************************* N O T E S ********************************
*               You must have SELECT privilege on the DB2 catalog          *
*               to complete this function successfully.                    *
***************************************************************************
```

Figure 29. Specify Directory to Upgrade

The following prompt is provided.

Authorization ID

> Authorization ID used to create the Optim Directory. This is set to the Authorization ID profiled during the original installation if the same TSOID is being used for this execution of the upgrade. If it is not the same, **Authorization ID** is blank.

If the upgrade is being performed under a different TSO ID and **Authorization ID** is blank, provide an Authorization ID or leave blank and press ENTER. If the Authorization ID you specify is valid, the menu is redisplayed. If it is invalid, an error message is displayed prompting you to re-specify the Authorization ID.

If you leave **Authorization ID** blank or the Authorization ID supplied is invalid, a selection list is displayed. The selection list includes the available Authorization IDs followed by the name of the database and table space in which they reside.

```
------------------ Directory CreatorID Candidate List ----------------------
Command ===>                                              Scroll ===> PAGE

Cmd   CreatorID     DBName    TSName                        1 of 3
---   ---------     --------  -------
***   *************** TOP *******************
__    FOPDEMO    IN TECHDB    INSST1
S__   OPTUSR     IN TESTDB    TS1
___   IBMTECH2   IN SPRTDB    TS2
***   ************** BOTTOM *****************

Enter an 'S' to select the Optim Directory CreatorID
```

Figure 30. Directory Creator ID Candidate List

Type S before the Authorization ID you wish to select. In the previous figure, OPTUSR is selected.

After you have selected an Authorization ID, press ENTER. The Upgrade menu is redisplayed. If you reselect Option 3, the selected Authorization ID is displayed on the Specify Directory to Upgrade panel.

The installation procedure identifies the current version that is to be upgraded based on the Authorization ID.

Upgrade Directory Tables

To upgrade the Optim Directory tables, use Option 4 on the Upgrade panel.

The Optim Directory contains columns and data that are dependant on the release of Optim and DB2. In this step, your existing Directory tables are upgraded as required for the new release. Upgrade Directory tables when:

- upgrading to a new release of Optim
- upgrading the DB2 release of a subsystem where Optim is installed

Your Directory must include all tables for the current release and all earlier releases.

When you perform this step of the upgrade, this panel displays:

```
Information Regarding Directory Table Conversion
Command ===>

      The upgrade process includes a directory table conversion.
      As a result, the process may take a few minutes to complete.

      Please remember to recompile any Column Map and Security
                          exits you may have.


Press ENTER key to continue
```

After converting the tables, you must create the indexes. (Note that an index is not created for ADB2AUDIT.) For each index to be created, the Creating Clustering Index panel prompts for index specifications. (For details about the Creating Clustering Index panel, see "Create Indexes" on page 28.)

Bind Plan for Product Execution

To set up the new local and remote plans and bind the plans for product execution, use Option 5 on the Upgrade panel.

The following panel is displayed.

```
------------- Provide Access to Optim at Remote/Local Subsystems --------------
OPTION  ===>

    1 - Create SYNONYMs to access the DB2 Catalog & Product Directory.
    2 - Setup Plan for Product Execution at the Current Location.
    3 - Setup Plans for Product Execution at Remote Locations.

    Plan Owner          ===>
    Plan Qualifier      ===>
    Directory CreatorID    :
    Table Library DSN   ===>

Optim does not explicitly reference either the Product Directory or the
DB2 Catalog.  Synonyms for the Plan Qualifier must be established to the
the directory and catalog (or an alternate set of tables/views) before binding

******************************* N O T E S ********************************
* Plan/Package owner must have SYSADM or BINDADD authority, SELECT privilege *
* on Catalog, ownership of Optim Directory Tables, and authority to change   *
* the SQLID.                                                                 *
*****************************************************************************
```

Figure 31. Provide Remote/Local Access - Upgrade

Select the steps in the listed order:

1. Create the synonyms required to bind the plan.

The synonyms are for the DB2 Catalog and the Optim Directory tables. You will have the opportunity to specify the synonyms to be used for a shadow catalog. This allows you to run against tables other than the DB2 Catalog tables. If you do not modify the default values specified, the synonyms are created for the DB2 Catalog tables.

2. Setup plan for product execution at the current location.

   You are prompted for the specifications to bind the packages and to grant user execute authority on the packages. Also, you are prompted for the specifications to bind the plan and then to grant user execute authority on the plan. This step is required.

3. Setup plans for product execution at remote locations.

   Once the packages are created, you can setup plans to access the remote subsystems. Remote subsystems are available to users by specifying the location name in the **LOCATION** prompt provided on the main menu.

   This step is performed only if you intend to access remote subsystems from the current location. You must be connected to the local subsystem when you execute this step.

You are prompted for the plan owner and qualifier. The plan qualifier owns the synonyms created to bind the plan. The plan owner must have the authorization to change the SQLID to the plan qualifier.

The **Directory CreatorID** specifies the Directory for the bind in step 2 of the Upgrade and cannot be modified.

You are also prompted to specify the dataset name of the table library. This table library dataset contains a list of the plans created for remote access.

If you specify the plan qualifier as other than the one specified for the original installation, the information on the panels is the same as that documented for a full install; however, the processing is as described in this section. ("Bind for Product Execution" on page 30 describes the information on the full install bind panels.)

The panels in the figures in this section contain the information that is displayed when you perform the upgrade bind using the name of the plan qualifier specified for the original installation.

You can select each step or press ENTER each time the panel is redisplayed to proceed through the steps automatically in the listed order.

Option 1 - Create Synonyms

When you upgrade Optim, you can set up Optim to execute using either the DB2 Catalog or to execute using alternate tables or views. To execute using alternate tables or views, you must create the synonyms to be used for a shadow catalog.

When you select Option 1 on the Provide Access to Optim at Remote/Local Subsystems menu prior to creating the synonyms for the Optim Directory and the DB2 Catalog, you are given the opportunity to override the DB2 Catalog specification. Use the Specify Synonyms for Catalog Access panel to specify the names of the catalog tables or views that correspond to the Optim synonyms. If the same plan qualifier has been specified, the panel will show the existing synonyms.

In the following figure, the defaults are displayed mapping the Optim synonyms to the IBM catalog. If you wish to run against other tables, you can change the Default Creator ID and/or the Catalog Table names to map the synonyms to other tables or views.

```
--------------------- Specify Synonyms for Catalog Access -------------------
Command ===>                                        SCROLL ===> PAGE

To execute Optim using the DB2 system catalog, press ENTER key.

To execute using alternate tables or views, change the catalog table names to
the desired alternate names and press ENTER key.

Default Creator ID  ===>  SYSIBM

Synonym:                    Catalog Table:
******************** TOP **************************
FOP_SYSCOLUMNS       ===>  SYSIBM.SYSCOLUMNS
FOP_SYSCOPY          ===>  SYSIBM.SYSCOPY
FOP_SYSDATABASE      ===>  SYSIBM.SYSDATABASE
FOP_SYSFIELDS        ===>  SYSIBM.SYSFIELDS
FOP_SYSFOREIGNKEYS   ===>  SYSIBM.SYSFOREIGNKEYS
FOP_SYSINDEXES       ===>  SYSIBM.SYSINDEXES
FOP_SYSINDEXPART     ===>  SYSIBM.SYSINDEXPART
FOP_SYSKEYS          ===>  SYSIBM.SYSKEYS
FOP_SYSRELS          ===>  SYSIBM.SYSRELS
FOP_SYSSTOGROUP      ===>  SYSIBM.SYSSTOGROUP
FOP_SYSSYNONYMS      ===>  SYSIBM.SYSSYNONYMS
FOP_SYSTABLES        ===>  SYSIBM.SYSTABLES
FOP_SYSTABLESPACE    ===>  SYSIBM.SYSTABLESPACE
FOP_SYSVIEWS         ===>  SYSIBM.SYSVIEWS
FOP_SYSVIEWDEP       ===>  SYSIBM.SYSVIEWDEP
FOP_SYSTABLEPART     ===>  SYSIBM.SYSTABLEPART
FOP_SYSPLANDEP       ===>  SYSIBM.SYSPLANDEP
FOP_SYSPACKDEP       ===>  SYSIBM.SYSPACKDEP
FOP_SYSPACKAGE       ===>  SYSIBM.SYSPACKAGE
FOP_SYSPLAN          ===>  SYSIBM.SYSPLAN
FOP_SYSCHECKS        ===>  SYSIBM.SYSCHECKS
FOP_LOCATIONS        ===>  SYSIBM.LOCATIONS
FOP_SYSAUXRELS       ===>  SYSIBM.SYSAUXRELS
FOP_SYSCHECKDEP      ===>  SYSIBM.SYSCHECKDEP
FOP_SYSCONSTDEP      ===>  SYSIBM.SYSCONSTDEP
FOP_SYSDATATYPES     ===>  SYSIBM.SYSDATATYPES
FOP_SYSLOBSTATS      ===>  SYSIBM.SYSLOBSTATS
FOP_SYSPARMS         ===>  SYSIBM.SYSPARMS
FOP_SYSROUTINES      ===>  SYSIBM.SYSROUTINES
FOP_SYSTRIGGERS      ===>  SYSIBM.SYSTRIGGERS
FOP_SYSSEQUENCES     ===>  SYSIBM.SYSSEQUENCES
FOP_SYSSEQUENCESDP   ===>  SYSIBM.SYSSEQUENCESDEP
FOP_SYSTABCONST      ===>  SYSIBM.SYSTABCONST
FOP_SYSTBLSPCSTATS   ===>  SYSIBM.SYSTBLSPCSTATS
****************** BOTTOM ************************
Review SQL prior to execution ===> YES (Y-Yes, N-No)
```

Figure 32. Specify Synonyms - Upgrade

Existing Synonyms Dropped

You should be aware that any existing plans, including existing Optim plans, that are bound with the
names used for the Optim synonyms will be invalidated. Any existing synonyms that match the
synonyms on this panel are dropped in order to create the new synonyms. When the existing synonyms
are dropped, the existing plans are invalidated. It is recommended that you check any names beginning
with FOP_SYS and FOP, since all synonyms defined by Optim begin with one of these strings. In
addition, synonyms will be created for the upgraded Optim directory.

If you want to run multiple releases of Optim, you can change the plan name and the plan qualifier to
create a new set of synonyms. See Appendix C, "Run Multiple Releases," on page 199 for more
information.

If you specify Yes to the prompt **Review SQL prior to execution**, the SQL is displayed for those
synonyms that are to be dropped and created.

When reviewing the SQL, the Drop Synonym SQL and Create Synonym SQL panels are displayed. If the SQL that is generated is not what you intended, you can choose to re-specify the names on the Specify Synonyms for Catalog Access panel.

The Drop Synonym SQL panel is displayed only when the name of a synonym specified on the Specify Synonyms for Catalog Access panel is the same as an existing synonym and the name of the catalog table is different. This occurs most frequently when you choose to re-specify the Catalog Access. The Drop Synonym SQL panel is displayed prior to the Create Synonym SQL panel. This provides a warning that an existing synonym will be affected. The Drop and Create Synonym statements are executed in a single unit of work after all specifications are complete and, if desired, reviewed.

Options 2 and 3 - Setup Plan/Packages for Product Execution

When you upgrade Optim, you must specify a plan for executing Optim at the current location or at a remote subsystem. The options that you use to specify plans when upgrading Optim work the same as the options that you use to specify plans when installing Optim.

For additional information:

Option 2
See "Option 2 - Setup Plan for Product Execution at Current Location" on page 34.

Option 3
See "Option 3 - Setup Plans for Execution at Remote Subsystems" on page 42.

Unload Optim Sample Data and ODM Optim Connect

Use Option 6 on the Upgrade panel to unload data for the Optim sample database and ODM Optim Connect.

A sample database is distributed with Optim to help demonstrate various functions. Sample tables for the database are distributed in XMIT format. ODM Optim Connect provides read-only access to Archive Files. The ODM data sets are also provided in XMIT format. See Appendix F, "ODM Optim Connect," on page 215 for details on installing ODM.

```
─────────── Unload Optim Sample Data and ODM Optim Connect ───────────
OPTION ===>                                                   More:   +

Below is a list of the Optim samples and ODM Optim Connect
files that have been packaged with Optim in XMIT format.
Select the files that you wish to unload and use in your environment.
 You may optionally specify a unit, volume, and STORCLAS for the unloaded
data sets.

Unload selection list:
    Enter "/" to select the data sets to unload.

    _ Unload All

    _ FOPNDEMO - Sample extract file
    _ FOPNLGCY - Sample legacy file
    _ FOPNBKOR - Sample sequential legacy file
    _ FOPNVEND - Sample legacy file (VSAM)
    _ FOPNVENI - Sample legacy file (IMS)
    _ FOPNDEPT - Sample legacy file (IMS)
    _ FOPNDEIX - Sample legacy file (IMS)
    _ FOPNJOB  - Sample legacy file (IMS)
    _ FOPNJOIX - Sample legacy file (IMS)
    _ FOPODMKT - ODM Install Kit
    _ FOPODMLD - ODM Install Load

.  Optional attributes for unloaded data sets:
.      UNIT               ===>
.      VOLUME             ===>
.      STORCLAS           ===>
```

Figure 33. Unload Optim Sample Data and ODM Optim Connect

On this panel, select the files you want to unload. You can optionally enter values for UNIT, VOLUME and STORCLAS for the unloaded data sets. When you press Enter, Optim unloads those selected and creates data sets using the naming convention *qual1.qual2.*DEMO.EXTRACT, where *qual1.qual2.* are the first two qualifiers of your installation.

Copy the Site Options

You might wish to copy the existing site defaults from a prior release to apply to this release of Optim. Use Option 7 on the Upgrade menu to copy and upgrade the defaults specified for the previous release, maintaining all current settings.

The Copy and Upgrade Site Options from Prior Release panel is displayed.

```
--------------- Copy and Upgrade Site Options from Prior Release ---------
 COMMAND ===>

 New Release Load Library   ===>
 Prior Release Load Library ===>


 Enable Optim Product(s) After Copy          ===> YES (Y-Yes, N-No)
 Modify Site Options After Copy              ===> YES (Y-Yes, N-No)

 Migrate Custom Options (Zaps)               ===> YES (Y-Yes, N-No)

****************************** N O T E S ********************************
*     You must have authority to update the New Release Load Library    *
*     to complete this function successfully.                           *
************************************************************************
```

Figure 34. Copy and Upgrade Site Options from Prior Release

The following prompts are presented:

New Release Load Library
> The name of the load library containing the release of Optim.

Prior Release Load Library
> The name of the load library containing the prior release of Optim.

Enable Optim Product(s) After Copy
> Indicator for enabling Optim after copying the site options. If you specify Yes, the Product Licensing/Configuration panel is displayed. This panel is described in "Enable Optim" on page 46.

Modify Site Options After Copy
> Indicator for modifying the site options after they are copied. If you specify Yes, the Site Options panel is presented. This panel is described in "Customizing the Site Options" on page 49.

Migrate Custom Options (Zaps)
> Indicator for migrating all enabled custom zaps in the library specified as the **Prior Release Load Library** to the new release, using the library specified as **New Release Load Library**. The default is Yes.
>
> To display the Custom Options panel, on the Optim main menu enter the command **0.C**:

```
----------------------- Custom Options Display ----------------------
Command ===>

    Available Line Cmds: E-Enable, D-Disable, M-Modify              1 OF 98


CMD   Status   Num        Description
---   --------  ---   ---------------------------------------------------------
      ***************************** TOP *************************************
___   Enabled    1        Enable AD security exit
___   Disabled   2        Show step report in French
___   Disabled   4        Enforce DB2 RI during load
___   Disabled   5        Always perform autocommit (Override AUTOCOMMIT Use Option)
___   .
___   .
___   .
      **************************** BOTTOM ***********************************
```

Figure 35. Custom Options Selection

The panel displays a list of numbered custom options, brief descriptions, and their status, Enabled or Disabled. Use the line commands E (Enable) or D (Disable) to select zaps to be migrated to your upgraded release of Optim. IBM recommends that you enable only those custom zaps that are enabled on the release of Optim that you are upgrading from. Some custom options also require a value. For options that require a value, you will be prompted to enter the value immediately after enabling the option. Once enabled, you may also specify the M (Modify) line command to modify the current value of an option which must contain a value.

Maintain Directory and Sample Database

Option 8 on the Upgrade panel is used to update the Primary Keys and Relationships in the Optim directory and to install or upgrade the Optim sample data.

```
----------- Upgrade Optim Directory ----------------------------------------
OPTION ===>


    1 - Create Copies of DB2 Relationships
    2 - Create OPTIM Primary Keys for all Unique Indices
    3 - Upgrade Sample Database
    4 - Create Legacy VSAM Sample Files
    5 - Load Legacy VSAM Sample Data
    6 - Create Legacy IMS Sample Databases
    7 - Load Legacy IMS Sample Data

********************************* N O T E S *********************************
*     The authorization levels required to complete these functions vary.   *
*     Detailed requirements are noted with the function specification.       *
****************************************************************************
```

Figure 36. Upgrade the Optim Directory

You can select the options individually or press ENTER each time the panel is redisplayed to perform each option in the listed order.

Option 1

>   Create shadow copies of the DB2 relationships in the Optim Directory. Although not required, this option should be executed to provide optimal performance.

Option 2

>   Create primary keys in the Optim Directory for those tables that do not have a primary key but do have a unique index. Although not required, this option should be executed especially if RI has not been defined to DB2.

Option 3

>   Upgrade the sample database to add any additional data or new tables for the current release. These are used in the training provided by IBM.

Option 4

>   Create sample Legacy VSAM files. Required if you are installing *Move* or *Compare for IMS, VSAM, and Sequential Data*.

Option 5

>   Load sample Legacy VSAM data. Required if you are installing *Move* or *Compare for IMS, VSAM, and Sequential Data*.

Option 6

>   Create Legacy IMS sample databases. Required if you are installing *Move* or *Compare for IMS, VSAM, and Sequential Data*.

Option 7

>   Load Legacy IMS sample files and objects. Required if you are installing *Move* or *Compare for IMS, VSAM, and Sequential Data*.

These options are also available from the **Maintain Directory and Sample Database** option on the main Install menu. For details, see the following:

Option 1

>   "Create Copies of DB2 Relationships" on page 101.

Option 2

>   "Create Optim Primary Keys" on page 102.

Option 3

>   "Create/Load Sample Database" on page 91.

Option 4

>   "Create Legacy VSAM Sample Files" on page 104.

Option 5
"Load Legacy VSAM Sample Data" on page 105.

Option 6
"Create Legacy IMS Sample Databases" on page 105.

Option 7
"Load Legacy IMS Sample Data" on page 106.

Option 3 - Upgrade the Sample Database

When Option 3 on the Upgrade menu is selected, the Upgrade Optim Sample Database panel is displayed.

The Upgrade Optim Sample Database panel is similar to the panel displayed when the Directory Maintenance option is selected from the main Install menu. However, the prompts for subsystem, SQLID, and dataset names are not needed here.

```
 -----------  Upgrade Optim Sample Database  --------------------------------
OPTION  ===>

Upgrading the Sample Database involves the following steps:

 1 - Specification of the Storage Group
 2 - Specification of the Database
 3 - Specification of the Table Space
 4 - Specification of the Sample Tables
 5 - Loading of the Sample Tables
 6 - Grant User Authority to Access the Sample Tables




********************************* N O T E S *********************************
*        The authorization level required to complete this process is       *
*        dependent upon the specifications of the various components.        *
*        The details are supplied on the individual specification panels.    *
****************************************************************************
```

Figure 37. Upgrade the Sample Database

You can select the options individually or press ENTER each time the panel is redisplayed to perform each option in the listed order. These options are all discussed in "Create/Load Sample Database" on page 91.

Customize the Batch Execution Environment

Option 9 on the Upgrade menu is available when upgrading or installing Move, Compare, or Archive. Use this option to customize the distributed skeletons used to generate JCL and control statements for batch processes performed by these Optim components, and to specify additional batch STEPLIB libraries.

See "Customizing the JCL Skeletons" on page 120 for information about customizing the JCL skeletons.

The Supply Batch Parameters panel prompts for several parameters required to support the JCL skeletons. You are first prompted for the name of the prior release skeleton library to populate the panel with the parameters from the prior release. The following panel prompts for the prior release skeleton library.

```
----------------- Specify Prior Release Skeleton Library ---------------------
COMMAND ===>


 Prior Release Skeleton Library:
     ===>



****************************** N O T E S ********************************
*          You must have authority to update the Current Release        *
*          Skeleton Library to complete this function successfully.      *
************************************************************************
```

Figure 38. Prior Release Skeleton Library

Use ENTER or END to continue. The Supply Optim Batch Parameters panel is displayed. If you have specified a prior release skeleton library, the values from that library are provided. Otherwise, the default values are displayed. You can modify the parameters.

```
------------ Supply Optim Batch Parameters -------------------------------------
COMMAND ===>

Product Skeleton Library  ===>
Product Load Library      ===>
Enter Additional Batch STEPLIB Libraries ===> YES (Y-Yes, N-No)
  such as, non-LNKLST DB2 Libraries and/or Alternate Utility Libraries

Batch Region Size ===> 0M    (0-2047 megabytes; 0M requests Max Region Size)

Permanent Dataset Defaults:   Temporary Dataset Defaults:  Tape Defaults:
 Unit Type    ===> TSODA       Unit Type    ===> SYSDA     MaxUnit ===> 1
 Primary CYLS ===> 3           Primary CYLS ===> 3

Utility Block Sizes: Load Input File ===> 23476     Image Copy File ===> 23476
Use SMS for New Load and Image Files ===> Y          (Y - Yes, N - No)
Image Copy Dataset Name Suffixes:  (1-4 characters)
  COPYDDN Backup ===> BKUP  RECOVERYDDN ===> RECV RECOVERYDDN Backup ===> REBK

****************************** N O T E S ********************************
*          You must have authority to update the Skeleton Library        *
*          to complete this function successfully.                       *
************************************************************************
```

Figure 39. Supply Batch Parameters - Upgrade

The following are presented:

Product Skeleton Library DSN
> The data set name of the skeleton library.

Product Load Library DSN
> The data set name of the load library.
>
> **Note:** The **Product Load Library** must be the value specified for the installation and in the FOPCUST CLIST.

Enter Additional Batch STEPLIB Libraries
> Additional batch STEPLIB libraries, including alternate utility libraries or load libraries that are not in the link list. Specify:
>
> **Y**   Display a panel to specify the libraries.
>
> **N**   Do not display the panel.

Batch Region Size
> The batch region size in megabytes. Specify 0M to request the Max Region Size.

Permanent Dataset Defaults

The default values for permanent data sets. Specify:

Unit Type
Default is TSODA.

Primary CYLS
Default is 3 cylinders.

Temporary Dataset Defaults

The default values used for temporary dataset allocation. Specify:

Unit Type
Default is SYSDA.

Primary CYLS
Default is 3 cylinders.

Tape Defaults

The default value, in the range 1–255, used for determining the number of tapes that can be assigned to a batch job:

MaxUnit
Default is 1 unit.

Utility Block Sizes

The block sizes for the following:

Load Input File
Default is 23476.

Image Copy File
Default is 23476.

Use SMS for New Load & Image Files

Indicator for using SMS when load and image files are created.

Y   Use SMS. Default.

N   Do not use SMS. (**Utility Block Sizes** must be greater than zero.)

Image Copy Dataset Name Suffixes

The suffix used for the Image Copy data sets. The suffix may contain from one to four characters. Specify the suffix for:

COPYDDN Backup
Default is BKUP.

RECOVERYDDN
Default is RECV.

RECOVERYDDN Backup
Default is REBK.

If you enter YES to **Enter Additional Batch STEPLIB Libraries**, the following panel is displayed.

```
---------------- Supply Additional Batch STEPLIB Libraries ------------------
COMMAND ===>
Enter required field at the cursor position.
Product Load Library      :
   Site Column Map Load    ==>

DB2 Libraries (required if not included in LNKLST):
   DB2 User Load Library   ==>
   DB2 Exit Load Library   ==>
   DB2 Load Library        ==>
   DB2 Template Library    ==>

Alternate Utility Libraries:
   Utility Load Library    ==>
   If BMC Utilities:
     SMS Service Routine    ==>

******************************* N O T E S *********************************
*          You must have authority to update the Skeleton Library        *
*          to complete this function successfully.                       *
**************************************************************************
```

Figure 40. Supply Additional Batch STEPLIB Libraries - Upgrade

The following prompts are presented:

Product Load Library DSN
> The data set name for the Product Load Library, specified on the Supply Batch Parameters panel, is entered automatically. This value is included in the batch JCL.

Site Column Map Load DSN
> The name of the data set to contain the Column Map Exit programs, if Column Map exit programs are to be used.

DB2 Libraries
> The names of the following load libraries when they are not included in the linklist:
> - DB2 User Load Library DSN
> - DB2 Exit Load Library DSN
> - DB2 Load Library DSN
> - DB2 Template Library DSN

Alternate Utility Libraries
> The data set name of the load library for a load or unload utility, other than DB2:
>
> Utility Load Library DSN
>
> The data set name of the following load library, if using a BMC utility (that is, LOADPLUS or UNLOAD PLUS):
>
> SMS Service Routine DSN

Executing Batch Jobs Saved before Release 5.5

Prior to Release 5.5, when Optim online processes generated batch jobs, they stored the job parameters in a SYSIN data set, using an Optim proprietary format. The only method available to modify the job parameters was the Optim override facility.

In Release 5.5 and later, Optim executes the Batch Utility for processes performed in batch. The Batch Utility SYSIN data set consists of a series of control statements defining the job function to be performed. Using these control statements, you can view and modify the specific job parameters directly. (The *Batch Utilities Guide* describes the Batch Utility control statements.)

Note: Input to the batch utility requires the entire batch job, in addition to the binary SYSIN data.

When you upgrade to Release 5.5 or later, you can execute batch jobs saved from previous releases in the following two ways:

1. Continue to submit the existing batch jobs, making minor changes to the JCL that might be required for a new STEPLIB or plan name. Parameter overrides in the PSDFOVRD data set will remain available in Release 5.5 and future releases.

2. Convert existing batch jobs to the new Batch Utility format. You can then submit the converted jobs for execution, as needed. Since the converted jobs use the Batch Utility control statements to perform the functions, you can change the statements without having to recreate jobs when you need to make parameter changes.

Converting Batch Jobs

To convert existing batch jobs to the new Batch Utility format, execute the Batch Utility, supplying one or more JCL_REFORMAT statements in the SYSIN data set. The JCL_REFORMAT statement converts batch jobs generated prior to Release 5.5 to the Batch Utility format. The *Batch Utilities Guide* describes the format of the control statements in the SYSIN data set.

If the Batch Utility finds errors in the JCL_REFORMAT keywords, it issues a return code of 8 and terminates the run immediately. If the conversion of a data set or member is skipped because of improper format, the program writes appropriate messages to the Batch Utility Summary Listing, continues processing with the next data set or member, and issues a return code of 4.

```
JCL_REFORMAT
 INDSN dsname
 [ MEMBER memname ]
 OUTDSN dsname
 [ OUTSUFFIX suffix ]
 [ REPLACE { YES | NO } ]
 [ VOLUME volser ]
 [ UNIT unit type ]
```

INDSN Specifies the name of a sequential or partitioned data set containing the batch job to be converted. For sequential data sets, use a wild card (* or %) at the end of the name to process a range of data sets.

Note: The Batch Utility skips a data set that is not in the format generated by Optim and writes a note in the Summary listing. (This is not considered an error condition.)

MEMBER This keyword is required if the input data set is partitioned. It specifies the member name(s) to be processed. You can use a wild card (* or %) at the end of the member name. If you specify * by itself without a member name, the Batch Utility processes all members in the data set.

Note: The Batch Utility skips a member that is not in the format generated by Optim and writes a note in the Summary listing. (This is not considered an error condition.)

OUTDSN Specifies the name of the output data set. This keyword is present only if the INDSN keyword does not include a wild card.

If the input data set is partitioned, the OUTDSN data set must exist. If the member name already exists, you must specify the keyword **REPLACE** YES.

If the input data set is sequential, the OUTDSN data set may exist. If the OUTDSN data set does exist, you must specify the keyword **REPLACE** YES and the data set must have a RECFM parameter of FB and an LRECL parameter of 80. If the OUTDSN data set does not exist, the Batch Utility allocates a new data set on the same volume as the input data set (unless the **VOLUME** keyword is also specified.)

OUTSUFFIX
Specifies the suffix that is to be appended to each input data set name to create the corresponding output data set name. This keyword is required if the INDSN keyword includes a wild card. The

Batch Utility inserts a period automatically between the input name and the suffix. If the length of the output data set name after the suffix has been appended is greater than 44 characters, the Batch Utility issues an error message and skips the data set. If the data set already exists, you must specify the keyword **REPLACE** YES.

REPLACE

Specifies the action to be taken if an output data set or member already exists. When the output data set or member does not exist, the Batch Utility ignores this keyword. Once you specify this keyword, it applies to all subsequent JCL_REFORMAT statements in the same Batch Utility execution until the Batch Utility finds another REPLACE keyword.

YES Replace the existing data set or member.

NO Skip processing the data set or member.

VOLUME This keyword is optional and specifies the volume on which output data sets are to be created. If you omit it, the Batch Utility creates each output data set on the same volume as the corresponding input data set. Once you specify this keyword, it applies to all subsequent JCL_REFORMAT statements in the same Batch Utility execution until the Batch Utility finds another VOLUME keyword.

UNIT This keyword is optional and specifies the unit type on which output data sets are to be created. If you omit it, the Batch Utility uses the site installation default. Once you specify this keyword, it applies to all subsequent JCL_REFORMAT statements in the same Batch Utility execution until the Batch Utility finds another UNIT keyword.

Example

The following is a simple example of a Batch Utility execution to convert a single saved batch job:

```
//jobname JOB job card parameters
 JCL for executing the Batch Utility
//SYSIN DD*
JCL_REFORMAT INDSN DEPT105.ARCHIVE.CUST
 OUTDSN DEPT105.ARCHIVE.CUST.REL71
/*
```

Users may add the following DD statements to the JCL, as required:

```
PSDFASUM DD SYSOUT = *
PSDFADIR DD SYSOUT = *
PSDFRPRT DD SYSOUT = *
CEEDUMP  DD SYSOUT = *
```

See the JCL Requirements section in the *Batch Utilities Guide* for a detailed description of the JCL requirements to execute the Batch Utility.

Chapter 6. Directory Maintenance

Use Option 3 to maintain the Optim Directory and sample database.

Directory maintenance can affect performance. You can improve system performance by creating shadow copies of DB2 relationships, which are used rather than explicit references to DB2 relationships. You can also provide faster access to data by creating primary keys for all tables that have a unique index but no primary key.

This option also populates the Optim Directory with sample object definitions and sample Legacy and IMS files to set up the sample database. A sample database is provided with Optim to demonstrate the product capabilities and to support training. The sample database consists of a set of tables with a number of typical RI rules defined. You can also refresh the sample database, as needed.

The Maintain Optim Directory and Sample Database menu lists the available options. You can select each option, or press ENTER to proceed through the options in the order listed on the menu. The options listed on the following panel can be re-executed as needed.

```
-----------  Maintain Optim Directory and Sample Database  ----------------
OPTION  ===>

Select the desired Directory Maintenance Option:

      1 - Create/Load the Sample Database
      2 - Load Sample Access Definitions and Column Maps
      3 - Create Copies of DB2 Relationships
      4 - Create OPTIM Primary Keys for all Unique Indices
      5 - Create Legacy VSAM Sample Files
      6 - Load Legacy VSAM Sample Data
      7 - Create Legacy IMS Sample Databases
      8 - Load Legacy IMS Sample Data

      9 - Drop Sample Database
     10 - Drop Legacy VSAM Sample Data
     11 - Drop Legacy IMS Sample Data

  DB2 Subsys for Install ===> DSNC
  Secondary SQLID        ===>
  Load Library DSN       ===> 'FOP.VBR3M0.SFOPLLIB'
  Install Library DSN    ===> 'FOP.VBR3M0.SFOPINST'
```

Figure 41. Maintain Directory and Sample Data

Options

The following options are available.

1 Create/Load the Sample Database
> Create the distributed sample database and load the sample data.

2 Load Sample Access Definitions and Column Maps
> Load the distributed sample Access Definitions, Column Maps, and primary keys into the Optim Directory.

3 Create Copies of DB2 Relationships
> Create shadow copies of DB2 relationships and load them into the Optim Directory. Rerun this step whenever a large number of relationships are added to your catalog.

4 Create Optim Primary Keys for all Unique Indices

Scan for tables that do not have DB2 primary keys. If any of these tables has a unique index, a primary key is loaded into the Optim Directory for that table. Rerun this step whenever a large number of tables are added to your catalog.

5 Create Legacy VSAM Sample Files

Create VSAM datasets for sample Legacy files, FOPNVEND and FOPNVENI.

6 Load Legacy VSAM Sample Data

Load sample files and objects (Legacy Tables, primary keys and relationships) for *Move* or *Compare for IMS, VSAM, and Sequential Data*.

7 Create Legacy IMS Sample Databases

Create the IMS datasets (FOPDEMO.FOPDEPDB, FOPDEMO.FOPDEPIX, FOPDEMO.FOPJOBDB, and FOPDEMO.FOPJOBIX) for *Move* or *Compare for IMS, VSAM, and Sequential Data*.

8 Load Legacy IMS Sample Data

Load IMS sample files (FOPNDEPT, FOPNDEIX, FOPNJOB, FOPNJOIX) and objects (Legacy Tables, Environment Definitions, and Retrieval Definitions) for *Move* or *Compare for IMS, VSAM, and Sequential Data*.

9 Drop Sample Database

Drop all sample tables, allowing you to refresh the sample database after your site completes training.

10 Drop Legacy VSAM Sample Data

Drop Legacy VSAM sample files and objects, allowing you to refresh the sample database after your site completes training.

11 Drop Legacy IMS Sample Data

Drop Legacy IMS sample data, allowing you to refresh the sample database after your site completes training.

Parameters

In addition to selecting an option, you are prompted to provide certain parameters.

DB2 Subsys for Install

The DB2 subsystem for which maintenance is desired.

Secondary SQLID

Optional name of a secondary SQLID to be used for the process. If this is blank, your primary SQLID will be used.

Load Library DSN

Fully qualified name of the load library created when Optim was installed using SMP/E.

Install Library DSN

Fully qualified name of the dataset created and loaded when Optim was installed using SMP/E.

These prompts will contain the names entered on the Install menu. You can modify the values on this panel; however, the user must have authority to perform these functions.

Several of the functions provided on this menu are also provided when installing or upgrading Optim. To simplify this manual, these functions are discussed only in this section. They provide the same functionality, regardless of how they are invoked.

Create/Load Sample Database

Use the Create Optim Sample Database panel to populate the database with the sample data distributed with Optim.

The sample database contains the following DB2 tables, Legacy Tables, and associated indexes:

Table 1. Contents of sample database

Table	DB2 Index
FOPDEMO.OPTIM_CUSTOMERS	FOPDEMO.XCUSTPK
FOPDEMO.OPTIM_SALES	FOPDEMO.XSALESPK
FOPDEMO.OPTIM_SHIP_TO	
FOPDEMO.OPTIM_ITEMS	FOPDEMO.XITEMPK
FOPDEMO.OPTIM_ORDERS	FOPDEMO.XORDERPK
FOPDEMO.OPTIM_SHIP_INSTR	FOPDEMO.XSHIPIPK
FOPDEMO.OPTIM_DETAILS	FOPDEMO.XDETLPK
FOPDEMO.OPTIM_FEMALE_RATES	
FOPDEMO.OPTIM_MALE_RATES	
FOPDEMO.OPTIM_STATE_LOOKUP	
FOPDEMO2.CUSTOMERS	FOPDEMO2.X2CUSTPK
FOPDEMO2.ORDERS	FOPDEMO2.X2ORDRPK
FOPDEMO2.DETAILS	FOPDEMO2.X2DETLPK
FOPDEMO2.ITEMS	FOPDEMO2.X2ITEMPK
FOPDEMO3.ACCOUNTS	FOPDEMO3.XACCTPK

For a detailed description of the sample database tables, see Appendix A. Sample Database Tables and Structure in the *Common Elements Manual*.

To create the sample database, the following panel is displayed.

```
 ----------    Create Optim Sample Database  ---------------------------------
 OPTION  ===>

 Creating the Sample Database involves the following steps:

  1 - Specification of the Storage Group
  2 - Specification of the Database
  3 - Creation of the Table Space
  4 - Creation of the Sample Tables
  5 - Loading of the Sample Tables
  6 - Grant User Authority to Access the Sample Tables

  DB2 Subsys for Install ===> DSNC
  Secondary SQLID        ===>
  Load Library DSN       ===> 'FOP.VBR3M0.SFOPLLIB'
  Install Library DSN    ===> 'FOP.VBR3M0.SFOPINST'



 ********************************* N O T E S *********************************
 *       The authorization level required to complete this process is       *
 *       dependent upon the specifications of the various components.        *
 *       The details are supplied on the individual specification panels.    *
 ****************************************************************************
```

Figure 42. Sample Database Creation

In addition to the list of options, you are prompted for the following information only when you are not currently performing a full install or upgrade. Then, these values are specified on the initial menus for install and upgrade. When displayed, you can modify the values on this panel.

DB2 Subsys for Install
>   The DB2 subsystem into which the sample database is to be created. The sample database must be created in each subsystem in which it will be used.

Secondary SQLID
>   Optional name of a secondary SQLID to be used for the install process. If blank, the primary SQLID will be used.

Load Library DSN
>   Fully qualified name of the load library created when Optim was installed using SMP/E.

Install Library DSN
>   Fully qualified name of the dataset created when Optim was installed using SMP/E.

The menu lists the steps required to create and load the training database. You can press ENTER each time this panel is redisplayed and you will be prompted automatically through the steps in the listed order. As a result, the sample database creation process can be suspended and restarted later without having to repeat any steps. The system records the last step successfully completed and resumes with the next step in the sequence. The process restarts where you suspended it, provided the same TSO User ID and subsystem name are used. The following sections discuss each step in the order listed on the menu.

Specify Storage Group

You are prompted for the information required to specify the storage group for the sample tables. If you specify a previously defined VCAT, you may want to select Option X and bypass this function.

```
    ----------   Specify Storage Group for Sample Tables -----------------------
OPTION  ===>

   Select type of storage group allocation desired:

               1 - Create new storage group
               2 - Use existing storage group
               3 - Use previously defined VCAT
               X - Bypass storage group specification

Storage Group Name ===>             (Options 1 or 2)
VCAT Name          ===>             (Options 1 or 3)

For OPTION 1:
         Volumes  ===>        ===>         ===>        ===>        ===>
         Password ===>

Review SQL prior to execution ===> YES (Y-Yes N-No)

******************************** N O T E S ********************************
*               You must have CREATESG or SYSADM authority               *
*               to complete this function successfully.                  *
**************************************************************************
```

Figure 43. Specify Storage Group for Sample Tables

Specify the option to perform the desired function and then supply the following information:

Storage Group Name
> Name of the storage group to be used. This is specified for Option 1 or 2 only. For Option 1, you cannot name a storage group already defined in the DB2 Catalog. If specified for Option 2, you must name a storage group defined in the DB2 Catalog.

VCAT Name
> Name or alias of the VCAT catalog. This is specified for Option 1 or 3 only. Use the alias if the name of the VCAT catalog is longer than 8 characters.

For OPTION 1 only:

Volumes
> Volumes on which the storage group is to be placed. At least one volume must be specified. A maximum of five volumes may be specified. If you are using DFSMS for storage management, you may enter an asterisk delimited by apostrophes as in `'*'` to direct DFSMS to manage the storage needed.

You can review the SQL that is generated in response to the prompts before it is executed by specifying Y for **Review SQL prior to execution**. If you wish to execute the SQL without reviewing it, specify N. For more information about this prompt, see Appendix B, "Review SQL Statements and DSN Subcommands," on page 197.

If you display the SQL and determine that it is accurate as displayed, press ENTER to execute it. However, if the SQL should be modified, press END. The Specify Storage Group for Sample Tables panel is redisplayed. You can modify any of your responses to the prompts, redisplay the generated SQL, and execute it.

Specify Database for Sample Tables

You are prompted for the information required to specify the database for the sample tables. You can create a new database or use an existing one. You can bypass specifying the database by selecting Option X on the Specify Database for Sample Tables panel.

```
------------------ Specify Database for Sample Tables ----------------------
OPTION  ===>

   Select type of database allocation desired:

               1 - Create new database
               2 - Use existing database
               X - Bypass database definition


Database Name      ===>

For OPTION 1:
 Storage Group Name ===>
 Buffer Pool        ===>              (BP0-BP49)
 Index Buffer Pool  ===>              (BP0-BP49)


Review SQL prior to execution ===> NO (Y-Yes, N-No)

******************************** N O T E S **********************************
*        You must have CREATEDBA, CREATEDBC, or SYSADM authority          *
*        in order to complete this function successfully.                 *
****************************************************************************
```

Figure 44. Specify Database for Sample Tables

Specify the option to perform the desired function and then supply the information for the following:

Database Name
> Name of the database to be used. When Option 1 is selected, you may not specify the name of a database already defined in the DB2 Catalog.

For OPTION 1 only:

Storage Group Name
> Name of the storage group as specified on the Specify Storage Group for Sample Tables panel. If a name is not specified on that panel or that step was bypassed, specify an existing storage group name. Leave blank to use the DB2 default, SYSDEFLT.

Buffer Pool
> Name of the default buffer pool for table spaces within the database. The panel shows the valid selections (BP0-BP49). If a value is not specified, the default is BP0.

Index Buffer Pool
> Name of the default buffer pool for indexes within the database. The panel shows the valid selections (BP0-BP49). If a value is not specified, the default is the database buffer pool.

Create Table Space for Sample Tables

You are prompted for the information required to allocate the table space for the sample tables. (All tables are created in the same table space.)

```
------------ Create Sample Tables Table Space ------------------------------
COMMAND ===>

Table Space Name    ===>             Database Name ===>
Storage Group Name  ===>                    PRIQTY ===>  100
      OR                                    SECQTY ===>  50
VCAT Name           ===>                    ERASE  ===>  YES  (Y-Yes, N-No)

FreeBlock: FREEPAGE ===>      (0-255)    PCTFREE ===>       (0-99)

Buffer Pool         ===>            (BP0-BP49)
Close               ===> NO         (Y-Yes, N-No)
Segment Size        ===> 4          (multiple of 4 between 4 and 64)
Lock Size           ===>            (ANY, PAGE, ROW, TABLESPACE, TABLE)
LockMax             ===>            (0 - 2147483647, SYSTEM)
MaxRows             ===>            (1 - 255)

Review SQL prior to execution ===> NO  (Y-Yes, N-No)

******************************** N O T E S ********************************
*   You need CREATETS, DBADM, DBCTRL, DBMAINT authority for the database   *
*   or SYSADM authority in order to complete this function successfully.   *
**************************************************************************
```

Figure 45. Specify Table Space for Sample Tables

Supply the information for the prompts:

Table Space Name
> Name of the table space to be created.

Database Name
> Name of the database in which the table space is to be created. The name specified on the Specify Database for Sample Tables panel is displayed.

Storage Group Name
> Name of the storage group as specified on the Specify Storage Group for Sample Tables panel. The named storage group must be described in the catalog.

VCAT Name
> Name of the VCAT catalog as specified on the Specify Storage Group for Sample Tables panel.
>
> **Note:** You can specify either a storage group name or a VCAT catalog name, but not both. The name specified on the Specify Storage Group for Sample Tables panel is displayed.

PRIQTY
> Primary space allocation. Optional for Storage Group; not allowed for VCAT.

SECQTY
> Secondary space allocation. Optional for Storage Group; not allowed for VCAT.

ERASE
> Setting for removing data from the DB2-defined datasets when the table space is dropped. Optional for Storage Group; not allowed for VCAT. Specify:
>
> **Y** Fill datasets with zeroes.
>
> **N** Do not erase datasets.

FREEPAGE
> When the table space is loaded or reorganized, controls the frequency for leaving a page of free space. Supply a value from 0 through 255. Optional.

PCTFREE
> Percentage of a page to leave as free space when the table space is loaded or reorganized. Supply a value from 0 through 99. Optional.

Buffer Pool

Name of the buffer pool. Optional. The panel shows the valid selections (BP0-BP49).

Close Specification for whether the files are to be opened once per session or once per logical unit of work. Optional. Specify:

Y Files are to be opened once per logical unit of work and then closed.

N Files are to be opened once per session and closed when the session terminates. Default.

Segment Size

Segment size if a segmented tablespace is desired for the training database. Required if **Lock Size** is TABLE.

Lock Size

Locking level. Optional. Specify:

ANY DB2 uses any locking level, usually page.

PAGE Page locking level.

TABLESPACE

Tablespace locking level.

TABLE Table locking level; this requires **Segment Size**.

ROW Row locking level.

If **Lock Size** is TABLE, **LockMax** must be 0.

LockMax

Maximum number of locks against a table at any time.

MaxRows

Maximum number of rows that can be placed into a tablespace.

Creating the Sample Tables and Indexes

This option creates the sample tables and their indexes. The database and tablespace names from any previously executed steps are supplied. Otherwise, enter the names to be used. (Note that these tables are created with type 2 indexes.)

The Table Creator ID, FOPDEMO, is displayed for information.

```
------------------ Create Optim Sample Tables  --------------------------
COMMAND ===>

 Table CreatorID: FOPDEMO       Database     ===>
                                Tablespace   ===>
 Index CreatorID    ===> FOPDEMO
 Storage Group Name  ===>                     PRIQTY  ===>
         OR                                   SECQTY  ===>
 VCAT Name           ===>                     ERASE   ===>      (Y-Yes, N-No)

 FREEPAGE ===>      (0-255)       PCTFREE        ===>      (0-99)
 Subpages ===>      (1,2,4,8,16)  Buffer Pool    ===>      (BP0-BP49)
 Close    ===> NO   (Y-Yes, N-No)

Press ENTER key to create tables, Press END key to cancel

********************************* N O T E S ********************************
*    You must have CREATETAB, DBADM, DBCTRL, or DBMAINT authority for the   *
*    specified database, or SYSADM to complete this function successfully.  *
*    In addition, specifying a BUFFER POOL or STORAGE GROUP requires the     *
*    USE privilege for that resource or SYSADM authority.                   *
***************************************************************************
```

Figure 46. Create Sample Tables

The following are included:

For the tables

Table CreatorID
> Default Creator ID for the sample database. This value cannot be modified. It is used to create the tables in the sample database. A variation of this value is used to create five additional tables in the sample database. (The variation consists of a "2" or "3" in the last or eighth character position.)

Database
> Name of the database as specified on the Specify Database for Sample Tables panel.

Tablespace
> Name of the table space created as named on the Create Sample Tables Table Space panel.

For the indexes

Index CreatorID
> The Creator ID for the indexes. The default is FOPDEMO. This value can be modified. If it is modified, the Creator ID for the indexes for the tables with the default Creator ID FOPDEMO2 and FOPDEMO3, is named using a variation of this modified value. The variation consists of a "2" or "3" in the last or eighth character position. (Note that Optim will overlay the eighth character of the Creator ID if you specify an eight-character value.)

Storage Group Name
> Name of the storage group as specified on the Specify Storage Group for Sample Tables panel. The named storage group must be described in the catalog. You can specify either a storage group name or a VCAT catalog name, but not both.

VCAT Name
> Name of the VCAT catalog as specified on the Specify Storage Group for Sample Tables panel.

PRIQTY
> Primary space allocation. Optional.

SECQTY
> Secondary space allocation. Optional.

ERASE

Specification for whether the DB2-defined datasets are to be filled with zeroes when the table space is dropped. Optional. Specify:

Y Erase datasets.

N Do not erase datasets.

FREEPAGE

Specification for how often to leave a page of free space when the index is loaded or reorganized. Supply a value from 0 through 255. Optional.

PCTFREE

Percentage of a page to leave as free space when the index is loaded or reorganized. Supply a value from 0 through 99. Optional.

Subpages

Unit in which a physical index page can be divided. Supply one of the following values: 1, 2, 4, 8, 16

Buffer Pool

The name of the storage pool. Optional. The panel shows the valid selections (BP0-BP49).

Close Indicator for the frequency of file openings. Optional. Specify:

Y Files are to be opened once per logical unit of work and then closed.

N Files are to be opened once per session and closed when the session terminates. This is the default.

Copy Indicator for performing an Image Copy of this index space. Specify:

Y Image Copy can be performed.

N Image Copy cannot be performed. This is the default.

Note: There is no prompt to review the SQL.

For a detailed description of the sample database tables and structures, see the *Common Elements Manual*.

Loading the Sample Tables

This procedure automatically populates the test database. Any data in the test database is deleted prior to loading the data from an extract file included in the installation library. After the test database is loaded, a message is displayed on the panel indicating that the step was successfully completed.

This same procedure is used for the original installation and for any refresh procedures. You can refresh the test database as needed by re-executing this step. At that time, you can choose to recreate the tables and load the data, or simply refresh the data in the existing tables. When you refresh the data, the existing rows are deleted and then the data from the member in the installation library is inserted.

You can use Option 9 to drop all of the tables. For additional information, see "Drop Sample Database" on page 107.

Granting Authority

The Grant Authority on Sample Database panel is used to assign individual users access to the sample database.

```
-------------------- Grant Authority on Sample Database  --------------------
COMMAND ===>

Grant ALL authority to:

              PUBLIC ===>     (Y-Yes, N-No)

              < OR >

The following AUTHID(s) ===>        ===>      ===>      ===>
                       ===>        ===>      ===>      ===>
                       ===>        ===>      ===>      ===>
                       ===>        ===>      ===>      ===>

Review SQL prior to execution ===>  NO (Y-Yes, N-No)

********************************* N O T E S *********************************
*       You must have authority to grant access to the tables           *
*       in order to complete this function successfully.                *
****************************************************************************
```

Figure 47. Grant Authority on Sample Database

The following prompts are presented:

PUBLIC

> Specify:
>
> **Y**   All users can access the sample database.
>
> **N**   You must provide the Authorization IDs for each user that can access the sample database.

AUTHID(s)

> If you specify N for **PUBLIC**, you can specify the individual Authorization IDs that are to have access to the sample database. A series of 8-character entry prompts is provided. Use the tab key to position to the next prompt. You may enter up to 16 AUTHIDs at once. The panel is redisplayed automatically to supply additional AUTHIDs. To enter additional AUTHIDs, use the CLEAR command to reset the AUTHID prompts and enter additional values.

Use END to exit the panel.

Load Samples

When you select the option to load the distributed sample Access Definitions, Column Maps, and primary keys, the Load Optim Samples panel is displayed.

```
----------- Load Optim Samples --------------------------------------------
COMMAND ===>


This process loads the distributed sample Access Definitions, Columns Maps,
and Primary Keys into the Optim Directory.  These samples are
required for the Optim supplied training classes and are of
general use in demonstrating the product capabilities.


Enter the Authorization ID of the Optim directory:

        Authorization ID ===> SYSIBM

Press ENTER without an AuthID for a selection list of Directory ID Candidates
Press ENTER with an AuthID to Load the Samples
Press END to bypass the Load


********************************* N O T E S *********************************
*            You must have SELECT privilege on the DB2 catalog            *
*            tables to build the "Candidate Directory ID" list.           *
****************************************************************************
```

Figure 48. Load Samples

The following prompt is provided:

Authorization ID
>   The Authorization ID used to create the Optim Directory. This is set to the Authorization ID
>   profiled during the original installation if the same TSO ID is being used for this execution of the
>   upgrade. If it is not the same, **Authorization ID** is blank.

However, if the value is not profiled, you can obtain a selection list by leaving **Authorization ID** blank
and pressing ENTER. The following is an example of the selection list of candidates that is displayed if
an Authorization ID is not entered or is invalid.

```
------------------ Directory CreatorID Candidate List ----------------------
Command ===>                                         Scroll ===> PAGE

Cmd  CreatorID    DBName    TSName                    1 of 3
---  ---------    --------  -------
***  **************** TOP ******************
___   FOPDEMO     IN TECHDB   INSST1
___   OPTUSR      IN TESTDB   TS1
___   IBMTECH2    IN SPRTDB   TS2
***  ************** BOTTOM ******************

Enter an 'S' to select the Optim Directory CreatorID
```

Figure 49. Directory Creator ID Candidate List

Type S before the Authorization ID you wish to select.

Once an Authorization ID is specified, the procedure is executed. The following are deleted and reloaded.

Access Definitions
>   **FOPDEMO.OPTIM.SAMPLE**--Access the sample database.
>   **FOPDEMO.SYSIBM.CATALOG**--Access the DB2 system catalog.

Column Maps
>   **FOPDEMO2.TWOCUSTS**--Define literal values for columns in the CUSTOMERS table.
>   **FOPDEMO2.CUSTEXIT**--Specify an exit routine for columns in the CUSTOMERS table.

The install library contains the source code for the sample exit routines. FOPPEXA and FOPP2K are also available as load modules in the product load library. The sample exit routines are:

FOPPEXA
> An exit routine, written in Assembler, that inserts a unique value in any numeric or character columns.

FOPPEXC
> An exit routine, written in COBOL, that selectively manipulates columns in the CUSTOMERS table.

FOPP2K
> An exit routine, written in Assembler, that sets the year portion of DATE and TIMESTAMP columns to 2000.

Primary Keys

The following are inserted into the PKDEF table:

SYSIBM.SYSFIELDS

SYSIBM.SYSTABLEPART

SYSIBM.SYSINDEXPART

FOPDEMO.OPTIM_FEMALE_RATES

FOPDEMO.OPTIM_MALE_RATES

FOPDEMO.OPTIM_SHIP_INSTR

FOPDEMO.OPTIM_SHIP_TO

FOPDEMO.OPTIM_STATE_LOOKUP

Create Copies of DB2 Relationships

Use the Copy DB2 Relationship to Directory panel to update relationship information in the Optim Directory with the relationship information from the DB2 Catalog.

When you request the utility to scan the DB2 Catalog for relationship information to create and update the Optim Directory table RELDEF, the following panel is displayed.

```
--------------------- Copy DB2 Relationship to Directory --------------------
COMMAND ===>


This process scans the DB2 Catalog for relationship data and saves a compiled
form in the Optim Directory.

To improve performance, Optim maintains copies of the DB2 Catalog
relationship tables compiled and indexed in the Optim Directory.
These copies are automatically synchronized with the DB2 Catalog by compiling
new and/or changed relationship data as it is encountered.

Upon initial installation or after major alterations of the DB2 defined
relationship data, use this process to load the Directory Tables to
provide immediate, responsive access to the DB2 defined relationships.


Press ENTER to copy the DB2 Catalog Relationships
Press END to bypass this copy

********************************* N O T E S *********************************
*            You must have authority to execute the Optim plan            *
*            to complete this function successfully.                      *
****************************************************************************
```

Figure 50. Copy DB2 Relationship to Directory

The Directory is a set of DB2 tables that act as an extension to the DB2 Catalog. To improve performance when accessing the database, the Directory maintains copies of the relationship tables with indexes defined. The Directory is updated automatically to reflect any changes to the DB2 relationships in the DB2 Catalog each time a change is encountered when accessing a table. However, if many new relationships are added to the DB2 Catalog, you might want to execute this function to resynchronize the Directory with the DB2 Catalog. When this step is completed, a message will be issued indicating how many DB2 relationships were extracted from the DB2 Catalog and copied to the Directory.

This function may also be invoked in a batch job. In volatile environments where new relationships are added or updated regularly, a batch job may be scheduled to run at regular intervals to keep the Directory synchronized with the catalog, ensuring good performance. Sample JCL to run this utility is provided in the installation library in the member FOPJCLUT.

Create Optim Primary Keys

As part of the installation process, you can request Optim to define primary keys to the Optim Directory for all tables in the subsystem that have a unique index but do not have a primary key defined to the DB2 Catalog.

This utility is executed at installation because primary keys are required for some processes, as well as to simplify creating relationships. This utility is also provided as an upgrade option and as a maintenance option and should be re-executed to create additional primary keys for newly created tables and/or unique indexes.

When this option is selected, the following panel is displayed.

```
------------ Create OPTIM Primary Key For Unique Indices --------------------
COMMAND ===>


This process scans the DB2 Catalog for unique indices on tables which do not
have DB2 Primary Keys defined and creates Primary Key Definitions for these
in the Optim Directory.


   Directory CreatorID ===> SYSOPT
   Optim Plan Name      : USRPLAN
   Report PDS Library  ===> 'OPTRT.PROD.INSTALL'

Press ENTER to create the Primary Keys
Press END to bypass this process

********************************* N O T E S *********************************
*          You must have EXECUTE authority for the Optim Plan and          *
*      update authority for the report PDS to complete this function.      *
****************************************************************************
```

Figure 51. Create Primary Key for Unique Indices

The following prompts are presented:

Directory CreatorID
> Creator ID of the Directory that is to receive the primary key values.

Optim Plan Name
> Name of the Optim Plan currently executing. This value cannot be modified. You must have EXECUTE authority for the named plan.

Report PDS Library
> Name of the partitioned dataset that is to contain the report generated by the conversion utility. The report will be stored in this partitioned dataset in a member named RPTPKEY. If report output is not desired, leave blank.

Press ENTER to proceed with creating primary keys, or use END to bypass the process.

Utility Processing

The utility proceeds as follows:
- The utility searches for a table that does not have a primary key.
- If there is no unique index for that table, the table is ignored.
- If there is a single, unique index, the utility creates a primary key for the table based on the index and stores this key in the Optim Directory.

If there are multiple unique indexes, the utility attempts to determine which has the fewest columns. If only one is located, it selects that index. If more than one, it selects the first index DB2 returns.

Utility Messages

The utility messages are written to the RPTPKEY member in the library named as the Report PDS Library. The default Report PDS Library is the installation library. The following is the format of the message generated for each primary key that is created. (The variable text is shown in Italics.)

Table Name:
> creator.tablename

Status:
> Created Optim Key from Unique Index *creator.indxname*

Key Columns:
>*column1 column2*

Regardless of whether you wish to write all of the messages to this file, a summary message is displayed at the terminal when the utility completes.

nnnn Tables Processed

nnnn Optim Primary Keys Created

nnnn Tables already had a Primary Key

where *nnnn* is a numeric value.

This function may also be invoked in a batch job. In volatile environments where primary keys are added or updated regularly, a batch job may be scheduled to run at regular intervals to keep the Directory synchronized with the catalog ensuring good performance. Sample JCL to run this utility is provided in the installation library in the member FOPJCLUT.

Create Legacy VSAM Sample Files

This procedure creates VSAM datasets for the Legacy sample files. These sample files are required only if you are installing *Move* or *Compare for IMS, VSAM, and Sequential Data*. Before you perform this step, the plan must be fully bound.

When you select the option to create the Optim Legacy sample files, the following panel is displayed.

```
-----------  Create Legacy VSAM Sample Files  --------------------------------
COMMAND ===>


The Legacy VSAM Sample Files consist of the following:

   FOPUSR.VBR3M0.FOPDEMO.FOPNBKOR
   FOPUSR.VBR3M0.FOPDEMO.FOPNVEND
   FOPUSR.VBR3M0.FOPDEMO.FOPNVENI

The FOPNBKOR file is sequential and was allocated and loaded when the
INSTALL member was executed.

The FOPNVEND and FOPNVENI files are VSAM.  This step defines the clusters
for these files.


Enter the volume name that the VSAM files should be created on:

            Volume Name ===>

Press ENTER to Create the Samples
Press END to bypass the Create


******************************** N O T E S ********************************
*            You must have RACF privilege to create the files           *
*            using the high level qualifiers listed above.              *
**************************************************************************
```

Figure 52. Create Legacy VSAM Sample Files

The following prompt is presented:

Volume Name
>Volume on which the files are to be placed.

This step creates two VSAM datasets, named:

```
qual1.qual2.FOPDEMO.FOPNVEND
qual1.qual2.FOPDEMO.FOPNVENI
```

The file qualifiers are obtained from the FOPCUST CLIST.

Press ENTER to create the Legacy VSAM sample files on the specified volume, or use END to bypass the process.

Load Legacy VSAM Sample Data

This procedure imports object definitions for the Optim Legacy sample files, created in the previous step. This step is required only if you are installing *Move* or *Compare for IMS, VSAM, and Sequential Data*.

The following object definitions are imported.

Legacy Tables
- FOPDEMO.VENDOR
- FOPDEMO.VENDITEM

Relationship

RVV

Primary Key

Fields in Legacy Table FOPDEMO.VENDITEM:
- VEND-ITEM-ID
- VEND-ID

A confirmation message is displayed on the panel indicating that the option was successfully completed.

Create Legacy IMS Sample Databases

This procedure uses the IMS DBDs and the associated PSBs to create IMS datasets. These IMS sample files are required only if you are installing *Move* or *Compare for IMS, VSAM, and Sequential Data*. Before you perform this step, the plan must be fully bound.

When you select the option to create the IMS sample databases, the following panel is displayed.

```
---------------  Create Legacy Sample IMS Database Files  -------------------
COMMAND ===>

The sample IMS Database consists of the following:
  DBDs:                        PSBs:
    FOPDEPDB   FOPDEPIX           FOPDEPPA   FOPDEPPR
    FOPJOBDB   FOPJOBIX           FOPJOBPA   FOPJOBPR

The following datasets will be created:
  FOP.VBR3M0.FOPDEMO.FOPDEPDB   FOP.VBR3M0.FOPDEMO2.FOPDEPDB
  FOP.VBR3M0.FOPDEMO.FOPDEPIX   FOP.VBR3M0.FOPDEMO2.FOPDEPIX
  FOP.VBR3M0.FOPDEMO.FOPJOBDB   FOP.VBR3M0.FOPDEMO2.FOPJOBDB
  FOP.VBR3M0.FOPDEMO.FOPJOBIX   FOP.VBR3M0.FOPDEMO2.FOPJOBIX

Enter the volume name that the IMS files should be created on:

            Volume Name ===>

Press ENTER to Create the Samples
Press END to bypass the Create

********************************* N O T E S *********************************
*          You must have RACF privileges to create these files.        *
****************************************************************************
```

Figure 53. Create Legacy IMS Sample Databases

The following prompt is presented:

Volume Name
        Volume on which the IMS files are to be placed.

This step creates the following IMS datasets, named:

qual1.qual2.FOPDEMO.FOPDEPDB
qual1.qual2.FOPDEMO.FOPDEPIX
qual1.qual2.FOPDEMO.FOPJOBDB
qual1.qual2.FOPDEMO.FOPJOBIX
qual1.qual2.FOPDEMO2.FOPDEPDB
qual1.qual2.FOPDEMO2.FOPDEPIX
qual1.qual2.FOPDEMO2.FOPJOBDB
qual1.qual2.FOPDEMO2.FOPJOBIX

The file qualifiers are obtained from the FOPCUST CLIST.

Press ENTER to create the Legacy IMS files on the specified volume, or use END to bypass the process.

Load Legacy IMS Sample Data

This procedure imports object definitions for the IMS datasets, created in the previous step. This step is required only if you are installing *Move* or *Compare for IMS, VSAM, and Sequential Data*.

The following object definitions are imported:

Legacy Tables
- FOPDEMO.DEPARTMENT
- FOPDEMO.EMPLOYEE
- FOPDEMO.POSITION
- FOPDEMO.JOBCODE

Environment Definitions

FOPDEMO

Retrieval Definitions

- FOPDEPDB
- FOPJOBDB

A confirmation message is displayed on the panel indicating that the option was successfully completed.

Drop Sample Database

Option 9 on the Maintain panel drops the existing sample database. This option is typically used to clean up the sample database after training. Additional training can then be provided on a fresh copy of the distributed sample database.

When you select Option 9, the following panel is displayed.

```
------------------- Drop Optim Sample Database ----------------------------
COMMAND ===>

 Select type of drop processing desired:
  1 - Drop Sample Tablespace - All Tables in Tablespace will be Dropped
  2 - Drop Individual Sample Database Tables

 Database ===>                    Tablespace ===>
 Individual Tables to Drop:
  FOPDEMO.SALES                     FOPDEMO2.CUSTOMERS
  FOPDEMO.CUSTOMERS                 FOPDEMO2.ITEMS
  FOPDEMO.SHIP_TO                   FOPDEMO2.ORDERS
  FOPDEMO.ITEMS                     FOPDEMO2.DETAILS
  FOPDEMO.ORDERS                    FOPDEMO2.BKORDER
  FOPDEMO.SHIP_INSTR                FOPDEMO3.ACCOUNTS
  FOPDEMO.DETAILS                   FOPDEMO.MALE_RATES
  FOPDEMO.ACTIONS                   FOPDEMO.FEMALE_RATES
                                    FOPDEMO.STATE_LOOKUP


********************************* N O T E S *********************************
*     You must be the owner,  have DBADM authority for the sample database, *
*     or have SYSADMN or SYSCTRL authority to complete this function        *
****************************************************************************
```

Figure 54. Drop Sample Database

Select one of the following options:

1. Drop Sample Tablespace

   Use this option to drop all tables in the sample database. Specify the names of the database and the tablespace to be dropped. The tablespace profiled during the install is automatically entered for **Tablespace**. If you specify another tablespace and select Option 1, the specified tablespace is dropped, regardless of whether it contains the sample database.

2. Drop Individual Sample Database Tables

   Use this option to drop only the listed sample database tables. The listed tables are the sample database tables that were created by the installation facility.

A member in the sample library named FOP9DROP contains the necessary statements to drop all tables in the sample database. If errors occur during the installation or upgrade process, you can use this member instead of this option to clean up the database prior to recreating it.

Drop Legacy VSAM Sample Data

Option 10 on the Maintain panel drops existing Optim Legacy VSAM sample data. This option is most often used to clean up the sample database after training. Additional training can then be provided using a fresh copy of the distributed sample database.

When you choose this option, the following files will be deleted:
- *qual1.qual2*.FOPDEMO.VENDOR
- *qual1.qual2*.FOPDEMO.VENDITEM

A confirmation message is displayed on the panel indicating that the option was successfully completed.

Drop Sample IMS Legacy Data

Option 11 on the Maintain panel drops existing Legacy IMS datasets. This option is most often used to clean up the sample database after training. Additional training can then be provided using a fresh copy of the distributed sample database.

When you choose this option, the following datasets will be deleted:
- *qual1.qual2*.FOPDEMO.FOPDEPDB
- *qual1.qual2*.FOPDEMO.FOPDEPIX
- *qual1.qual2*.FOPDEMO.FOPJOBDB
- *qual1.qual2*.FOPDEMO.FOPJOBIX
- *qual1.qual2*.FOPDEMO2.FOPDEPDB
- *qual1.qual2*.FOPDEMO2.FOPDEPIX
- *qual1.qual2*.FOPDEMO2.FOPJOBDB
- *qual1.qual2*.FOPDEMO2.FOPJOBIX

A confirmation message is displayed on the panel indicating that the option was successfully completed.

Chapter 7. Enable and Disable Products

After Optim is installed, you can enable or disable product use by selecting Option 4 on the Install main menu or, if you are a site administrator, Option P on the main menu.

The Product Licensing/Configuration panel is displayed. It is used to extend the trial period or permanently enable or disable Optim. When Optim is installed, all components are available for an initial trial period. Product names may vary, depending upon your site's license. For example, when Move includes the options for Delete and Data Privacy functions, its name appears as shown here:

```
---------------------- Product Licensing/Configuration -----------------------
Command ===>                                             Scroll ===> PAGE

Site CPUID        : 0000F2
Load Library DSN  : OPZ.REL113.OPTIONS.LOAD

     Product / Release          Current Product Status      Password
     ----------------------     ---------------------------  ---------   1 of 9
   ****************************** TOP *******************************
   Optim Access               In Evaluation: 30 Days Left
   Optim Move w/ Delete/DP     In Evaluation: 30 Days Left
   Optim Compare               In Evaluation: 30 Days Left
   Optim Archive               Permanently Licensed
   Optim Move for VSAM/Seq     In Evaluation: 30 Days Left
   Optim Move for IMS          In Evaluation: 30 Days Left
   Optim Compare for VSAM/Seq  Permanently Licensed
   Optim Compare for IMS       Permanently Licensed
   Optim Data privacy          In Evaluation: 30 Days Left
   *************************** BOTTOM ******************************

INSTRUCTIONS: Press ENTER to Continue, Enter END to Return or CANCEL to Cancel
              Enter Password to Enable, Extend Evaluation or License Product
              Enter DISABLE in Password field to Disable Product

*********************************** N O T E S ***********************************
*           You must have authority to update the Load Library          *
*           to complete this function successfully.                     *
********************************************************************************
```

Figure 55. Enable/Disable Products

Parameters

On the Product Licensing/Configuration panel, the following values are displayed automatically and cannot be modified:

Site CPUID
Value that identifies your site. Passwords are based on this value.

Load Library DSN
Fully qualified name of the load library created when the tape was unloaded. If installing from CD, it was created when you executed the FOPCUST CLIST.

Status

The **Current Product Status** is displayed as:

In Evaluation
The product is in a trial period. The number of days remaining in the trial period is displayed.

Expired

The trial period has expired. If you wish to extend the evaluation period or permanently license the product, contact your IBM Optim representative.

Permanently Licensed

The product is installed and permanently licensed.

Disabled

The product is installed, but has been disabled by site management.

Not Installed

The product is not installed.

Being Installed

The product is being installed and the final bind has not been completed.

Password

The **Password** is used to enable or disable the product. Type the password obtained from IBM to enable a disabled product, extend the evaluation period, or permanently license the product. Type DISABLE to disable the product.

The product is initially installed for a standard evaluation period. The trial period can be extended or the product licensed permanently using passwords supplied by IBM. A password, based on the CPUID as displayed on this panel, is required to permanently license each component of Optim.

Chapter 8. Bind or Rebind the DB2 Plan

You can bind or rebind the DB2 Plans, as needed, by selecting Option 5 on the main Install menu. The introduction of DB2 Package Versioning enables you to issue a PACKAGE BIND replace, without affecting the existing plan bind. Packages that have changed with current maintenance actually add a new Version of the Package identified by the PTF. Previously, before DB2 Package Versioning, when a specific Package was replaced, it caused the existing plan to be invalidated.

When you select Option 5 the following panel is displayed.

```
---------------- Provide Access to Optim at Remote/Local Subsystems -----------
OPTION  ===>

    1 - Create SYNONYMs to access the DB2 Catalog & Product Directory.
    2 - Setup Packages for Product Execution.
    3 - Setup Plan for Product Execution at the Current Location.
    4 - Setup Plans for Product Execution at Remote Locations.
    5 - Rebind Existing Package.
    6 - Rebind Existing Plan.

    Plan Owner         ===>
    Plan Qualifier     ===>
    Directory CreatorID ===> SYSIBM
    Table Library DSN   ===> 'FOP.VBR3M0.SFOPTENU'

Optim does not explicitly reference either the Product Directory
or the DB2 Catalog.  Synonyms for the Plan Qualifier must be established to
the directory and catalog (or an alternate set of tables/views) before binding

******************************** N O T E S ********************************
* Plan/Package owner must have SYSADM or BINDADD authority, SELECT privilege *
* on Catalog, ownership of Optim Directory Tables, and authority to change   *
* the SQLID.                                                                 *
****************************************************************************
```

Figure 56. Provide Remote/Local Access

Select the desired option.

1 Create SYNONYMS to access the DB2 Catalog and Product Directory
> The synonyms are for the DB2 Catalog and the Optim Directory tables. You will have the opportunity to specify the synonyms to be used for a shadow catalog. This allows you to run against tables other than the DB2 Catalog tables. If you do not modify the default values, the synonyms are created for the DB2 Catalog tables. For additional information, see "Option 1 - Create Synonyms" on page 31.

2 Setup Packages for Product Execution
> You are prompted for the specifications to bind the packages and grant user execute authority on the packages. For additional information, see "Bind Package for Current Location" on page 35.
>
> **Note:** You must specify a uniquely named package for each remote location that is to be accessed by the current location. The same package can be shared by multiple plans.

3 Setup Plan for Product Execution at the Current Location
> You are prompted for the specifications for the final bind and then to grant user execute authority on the plan. For additional information, see "Bind Plan" on page 38.

4 Setup Plans for Product Execution at Remote Locations
> Once the packages are created, you can setup plans to access the remote subsystems. Remote subsystems are available to users by specifying the location name on the Optim main menu. If

you intend to access remote subsystems from the current location, you must perform this step at the local subsystem. For additional information, see "Option 3 - Setup Plans for Execution at Remote Subsystems" on page 42.

5 Rebind Existing Package

After Optim is installed, you can use this option to submit an SQL statement to rebind an Optim package. If this option is chosen, the BIND parameters in effect at the last BIND, other than Collection ID prefix, member name, DBRM Library DSN, and message level, are used to rebind the existing package. Prompts are displayed with default Collection ID prefix and message level values. See "Option 5 - Rebind Existing Package" for additional information.

6 Rebind Existing Plan

After Optim is installed, you can use this option to submit an SQL statement to rebind the Optim plan. If this option is chosen, the BIND parameters in effect at the last BIND, other than plan name and message level, are used to rebind the existing plan. Prompts are displayed with default plan name and message level values. See "Option 6 - Rebind Existing Plan" on page 113 for additional information.

In addition to selecting an option, you are prompted for the plan owner and qualifier. The plan qualifier owns the synonyms created to bind the plan. The plan owner must have the authorization to change the SQLID to the plan qualifier. Specify the Directory for the bind in **Directory CreatorID**. You are also prompted to specify the dataset name of the table library. This table library dataset contains a list of the plans created for remote access.

You can select each step or press ENTER each time the panel is redisplayed to proceed automatically through the steps in the listed order.

Option 5 - Rebind Existing Package

When you select Option 5 on the Provide Access to Optim at Remote/Local Subsystems menu, the Rebind Existing Package panel is displayed. Use this panel to submit an SQL statement to issue a rebind of an Optim package.

```
--------- Rebind Existing Package for Execution at Current Location ---------
OPTION ===>

    Package Owner        :             Package Qualifier:

    Collection-ID Prefix  ===>
    Member Name          ===>
    DBRM Library DSN      ===>
    Rebind Message Level  ===>           (All, Warning, Error, Complete)
    Encoding Scheme      ===>           (Non Unicode Code Page)

    Review DSN subcommand prior to execution ===> YES  (Y-Yes, N-No)


******************************* N O T E S ********************************
*    The package owner must have BINDADD privilege or SYSADM authority   *
*    to complete this function successfully.                             *
*************************************************************************
```

Figure 57. Rebind Package for Execution at Current Location

Panel

The following prompts are provided:

Package Owner

The owner specified on the Provide Access to Optim at Remote/Local Subsystems panel.

Package Qualifier

The qualifier specified on the Provide Access to Optim at Remote/Local Subsystems panel.

Collection-ID Prefix

The prefix used to identify the collection of packages. By default, the Collection ID prefix specified on the Bind Package for Current Location panel is displayed.

Member Name

The name of the package.

DBRM Library DSN

The name of the library containing the DBRMs.

Rebind Message Level

The level of messages to be displayed. Specify:

All   All messages.

Warning

Warning, error, and completion messages.

Error   Only error and completion messages.

Complete

Completion messages.

Encoding Scheme

If your DB2 Default Encoding Scheme (DSNHDECM module ENSCHEME value) in the DSNHDECP system file is set to Unicode, enter a 4-digit EBCDIC code page value. The recommended value is 0037.

If you do not supply a value for this prompt, the DSNHDECM value is used.

Notes:
- While the Optim plan must be bound with an EBCDIC encoding scheme, Optim is still able to process data in ASCII and Unicode DB2 tables. When Optim is reading data from an ASCII or Unicode DB2 table, DB2 will convert characters from the table's encoding scheme to the target Optim EBCDIC encoding scheme. When Optim is writing data to an ASCII or Unicode DB2 table, DB2 will convert characters from the Optim EBCDIC encoding scheme to the target table's encoding scheme.
- The code page value is not validated. Use caution when entering a value.

Enter Yes for **Review DSN subcommand prior to execution** if you want to review the SQL before it is executed. A sample of the DSN subcommand display panel is shown in Appendix B, "Review SQL Statements and DSN Subcommands," on page 197.

Option 6 - Rebind Existing Plan

When you select Option 6 on the Provide Access to Optim at Remote/Local Subsystems menu, the Rebind Existing Plan panel is displayed.

Use this panel to submit an SQL statement to issue a rebind of the Optim plan. A **REBIND** differs from a **BIND REPLACE** command in that the existing parameters for the plan are carried forward and the rebind takes less overhead.

```
  ---------- Rebind Existing Plan for Execution at Current Location -----------
OPTION ===>

    Plan Owner              :              Plan Qualifier:

    Plan Name           ===>
    Rebind Message Level  ===>               (All, Warning, Error, Complete)
    Encoding Scheme       ===>               (Non Unicode Code Page)

    Review DSN subcommand prior to execution ===>  YES  (Y-Yes, N-No)


********************************* N O T E S *****************************
*      The plan owner must have BINDADD privilege or SYSADM authority     *
*      to complete this function successfully.                            *
**********************************************************************
```

Figure 58. Rebind Plan for Execution at Current Location

Panel

The following prompts are provided:

Plan Owner
> The owner specified on the Provide Access to Optim at Remote/Local Subsystems panel.

Plan Qualifier
> The plan qualifier specified on the Provide Access to Optim at Remote/Local Subsystems panel.

Plan Name
> The name of the current Optim plan. The name is obtained from the FOPCUST CLIST.

Rebind Message Level
> The level of messages to be displayed. Specify:
>
> **All**   All messages.
>
> **Warning**
> > Warning, error, and completion messages.
>
> **Error**   Only error and completion messages.
>
> **Complete**
> > Completion messages.

Encoding Scheme
> If your DB2 Default Encoding Scheme (DSNHDECM module ENSCHEME value) in the DSNHDECP system file is set to Unicode, enter a 4-digit EBCDIC code page value. The recommended value is 0037.
>
> If you do not supply a value for this prompt, the DSNHDECM value is used.
>
> **Notes:**
> - While the Optim plan must be bound with an EBCDIC encoding scheme, Optim is still able to process data in ASCII and Unicode DB2 tables. When Optim is reading data from an ASCII or Unicode DB2 table, DB2 will convert characters from the table's encoding scheme to the target Optim EBCDIC encoding scheme. When Optim is writing data to an ASCII or Unicode DB2 table, DB2 will convert characters from the Optim EBCDIC encoding scheme to the target table's encoding scheme.
> - The code page value is not validated. Use caution when entering a value.

Enter Yes for **Review DSN subcommand prior to execution** if you want to review the SQL before it is executed. A sample of the DSN subcommand display panel is shown in Chapter 8, "Bind or Rebind the DB2 Plan," on page 111.

Chapter 9. Create or Modify Site Defaults Modules

Optim includes a site defaults module named FOPMDFLT, which can be updated or used as a model to create individualized site defaults modules. This allows you to customize the Optim environment according to the needs of a subsystem, location, and/or SQLID.

After Optim is installed, you can create or modify the site defaults module. To do this, select Option 6 on the main Install menu to display the Create/Modify Site Defaults Modules panel.

```
--------------------- Create/Modify Site Defaults Modules ---------------------
 Option ===>                   (C-Create, U-Update)

 Target Load Library      :
 Target Member Suffix   ===> LT
 Target Member Password ===>

 If Option is Create, Specify Existing Member to Initialize New Member:

 Model Load Library  ===>
 Model Member Suffix ===> LT
```

Figure 59. Create/Modify Site Defaults Modules

Options

Select an option to create or update a site defaults module that is modeled on an existing site defaults module:

C-Create
> To create a new site defaults module, enter `C` at the **Option** prompt.

U-Update
> To update an existing site defaults module, enter `U` at the **Option** prompt.

Parameters

In addition to selecting an option, you must also supply the following:

Target Load Library
> Fully qualified name of the load library created when Optim was installed using SMP/E. This information is entered automatically.

Target Member Suffix
> The two-character suffix to append to FOPMDF to name a new site defaults module or to select an existing one.

Target Member Password
> The Administrator password, if necessary.

Model Load Library
> Fully qualified name of the load library in which the model member is located.

Model Member Suffix
> The two-character suffix of the site defaults module you wish to use as a model for the new module. The default value is LT. Note that after creating the first new module, you can use any existing module suffix.

After selecting an option and entering the required information, the site defaults module is displayed for you to modify and save. (See Figure 17 on page 49.)

Note: To implement a new site defaults module, you must also modify FOPCUST. See Chapter 2, "Set Up and Modify the Optim CLIST," on page 7 for information on modifying FOPCUST to assign site defaults modules based on subsystem, location, or SQLID.

Chapter 10. Integrate Into Your Environment

There are many ways in which you can integrate the solution into your environment. You can use methods for invoking the solution, creating customized menus of functions, customizing the CLISTs used to process SQL statements in ISPF EDIT, customizing the JCL skeletons used to execute batch processes, performance considerations, and other miscellaneous topics.

After the solution is installed, review these topics to customize the implementation and to ensure optimum performance at your site.

By default, the solution is invoked by the name of the CLIST, FOP. To invoke the products using a different command name, rename the FOP CLIST to the desired name. (Although you can rename FOPFAD2P, FOPD2EXP, and FOPD2EXE—CLISTs used to invoke Access during an ISPF EDIT session—you should not rename any other CLISTs supplied with the solution.)

Invoking Optim

A CLIST named FOP is used to invoke Optim. This CLIST, and all other CLISTs prefixed with FOP provided in the installation library is available to all potential users of Optim via copy to a CLIST library or ALTLIB.

See Chapter 2, "Set Up and Modify the Optim CLIST," on page 7.

The FOP CLIST may be invoked directly or included on an application selection menu.

Application Selection Menu

To include the FOP CLIST in an application selection menu, add a line to the &ZSEL statement in the)PROC section.

To make Optim accessible to users from the ISPF primary option panel or another application selection menu, edit the panel to include it as an option and add the following line to the &ZSEL statement in the)PROC section:

```
OP,'CMD(%FOP PARM(Z=&ZCMD) ) NOCHECK'
```

The option designator, OP, may be changed to any appropriate value.

A sample menu, FOP9PRIM, is included in the install library.

Using the FOP CLIST

Using the FOP CLIST, you can invoke the solution without installing it as an option on an application selection menu.

Since this CLIST supports a number of optional input parameters, you can write simple customized CLISTs that call Optim to:

- Provide a fast path to commonly used Access Definitions or functions.
- Provide less technical users with a way to browse and update selected data by predefining Access Definitions and then providing a fast path to the data in a controlled manner.

Four parameters can be passed as arguments to the FOP CLIST:

`'C=`*command*`'`
> Where *command* can be EDIT or BROWSE with an optional Access Definition name or a table name and keyword to invoke a panel to qualify the set of data for the table before beginning the session.

`SSN(`*subsystem*`)`
> Where *subsystem* is the name of the DB2 subsystem containing the data of interest. When a user displays the main menu for the solution, a prompt is provided for the name of the DB2 subsystem to be used. This value is stored in the ISPF profile member for the solution. As long as a valid subsystem name exists in the profile, the SSN argument is not required. If it is not present, an error will result.

`SQLID(`*sqlid*`)`
> Where *sqlid* is the authorization ID to be used. This argument is optional. If it is not provided, the SQLID to be used is taken from the ISPF profile, if present. If a profile is not present, DB2 derives the SQLID from the TSO sign-on.

`USER(`*input*`)`
> Where *input* is the TSO ID.

You can write a one-line CLIST to perform a task directly. The following are examples of customized CLISTs.

1. A CLIST can be used to enter an edit session of a particular set of tables from the ISPF command prompt. In the following sample statement, the tables that comprise the sample database are specified in the Access Definition FOPDEMO.SAMPLE:

   `FOP PARM('C=EDIT FOPDEMO.SAMPLE')`

2. A CLIST can be used to enter an edit session by naming a specific table. In the following, a table in the sample database, FOPDEMO.CUSTOMERS, is specified:

   `FOP PARM('C=EDIT TABLE FOPDEMO.CUSTOMERS')`

3. A CLIST can be used to display the Specify Selection Criteria panel before invoking an edit session. Name a specific table and include SEL in the statement. In the following, the panel is displayed before the edit session is invoked for the table FOPDEMO.CUSTOMERS.

   `FOP PARM('C=EDIT TABLE FOPDEMO.CUSTOMERS SEL')`

   In addition to "SEL", you can specify "COL" to display the Describe Columns panel or "SQL" to display the Specify SQL WHERE Clause panel.

You can enter these commands directly, but it is simpler to embed them in another CLIST. The install library contains a sample CLIST named FOP9CALL.

Customized Applications Using Optim

By predefining a set of Access Definitions and using the techniques described in the previous section, you can write a series of simple CLISTs and then create specialized menus for the Optim functions that are invoked by those CLISTs.

An Access Definition consists of all the information necessary to describe the set of data to be processed by Optim. The process of creating and modifying Access Definitions is described in detail in the *Common Elements Manual*. Additionally, each user manual focuses on how to use an Access Definition to perform the appropriate tasks.

- The *Access User Manual* discusses editing and browsing related data.
- The *Move User Manual* discusses data migration or copying data from one set of tables to another.
- The *Compare User Manual* discusses comparing two sets of related data.
- The *Archive User Manual* discusses archiving and restoring a set of related data.
- The *Compare for IMS, VSAM, Sequential File Data* discusses comparing legacy table data.

You can define a set of Access Definitions tailored to the requirements of a particular set of users and include them on a custom menu. The install library contains a sample menu, FOP9MENU, illustrating this technique.

Customizing the CLISTs to Process SQL Statements in ISPF EDIT

CLISTs distributed with Access can be used to process SQL statements from a source program during an ISPF EDIT session.

These CLISTs perform a DB2 PREPARE, EXPLAIN, or EXECUTE and have the following names:

FOPFAD2P
> Perform a DB2 PREPARE.

FOPD2EXP
> Perform a DB2 EXPLAIN.

FOPD2EXE
> Perform a DB2 EXECUTE.

The following user operands are available for these CLISTs:

SSN(*subsystem***)**
> Subsystem name

SQLID(*sqlid***)**
> SQL ID

LANG(*language***)**
> Language of the source program. Specify:
>
> **ASM**   For Assembly Language
>
> **C**   For C Language
>
> **COB** For COBOL
>
> **PL1**   For PL1
>
> **FOR** For Fortran

QUAL(*qualifier***)**
> Table qualifier used as a prefix for all unqualified tables, views, and aliases in SELECT, DELETE, INSERT, and UPDATE SQL statements.

USER(*input***)**
> Where *input* is the TSO ID.

APOST | QUOTE
> The input data string delimiter. If necessary, it will be transformed to the DB2 string delimiter. APOST is the default.

PERIOD | COMMA
> The character used in decimal data. It will be transformed, if necessary. PERIOD is the default.

The following operand is available for FOPD2EXE only:

WIDTH(*n***)**
> Width of the host variable prompt area. Specify a value from 10 through 70. This value defaults to the **Max Columnar Display Width** option on the User Options panel.

To customize these CLISTs, you can specify these operands with site-defined default values. For example, to define default values for all of the operands in the CLIST FOPFAD2P, modify the following statement in FOPFAD2P:

```
ISPEXEC    SELECT CMD(%FOPD2XSQ PARM('CHECK &PARM')) -
           NEWAPPL(PSAD) PASSLIB MODE(FSCR)
```

to include the operands as shown here:

```
ISPEXEC SELECT CMD(%FOPD2XSQ PARM('CHECK SSN(TAD2) SQLID(SAMPLE) -
        LANG(COB) &PARM')) NEWAPPL(PSAD) PASSLIB MODE(FSCR)
```

Each CLIST that is to include default values must be modified.

Note: One or more of these operands can be specified after &PARM to establish a site-controlled specification that overrides any user specification.

Customizing the JCL Skeletons

Several operations performed by the solution can be executed in batch. In addition to the panels that prompt for batch parameters, several ISPF skeletons are distributed on ISPSLIB, the dataset used to generate the JCL.

Of the skeletons, the following may require modifications, FOPDSTEP, FOPDDB2S, FOPDRGSZ, and FOPDBMCS. These skeletons are generated through Step 12 of a Full Install or Step 8 of an Upgrade. The panels provided in these steps allow you to modify the skeletons. The Supply Batch Parameters panel and the Supply Additional Batch STEPLIB Libraries panel are displayed for Step 12 of a Full Install and Step 8 of an Upgrade. For information about these panels, see "Customize the Batch Environment" on page 67 for installing and "Customize the Batch Execution Environment" on page 83 for upgrading.

FOPDSTEP

The skeleton FOPDSTEP must include the load library, any load libraries used for Column Map exit routines, and the DB2 DSNLOAD and DSN libraries, if these libraries are not in the system link list.

You can specify these libraries on the Supply Optim Batch Parameters panel or the Supply Additional Batch STEPLIB Libraries panel.

- The load library.
- The load libraries used for Column Map exit routines, if used.
- The DB2 DSNLOAD and DSN libraries, if not in the system link list. (The DB2 DSNEXIT library is necessary only if used.)

Note: To enable Centera and Tivoli support at your site, you must manually change FOPDSTEP by establishing the DSN to be added to the STEPLIB concatenation for batch jobs. Insert the following lines after the SEL &ADCMLIB statement.

```
)SEL &STORLIB ¬= &Z
//          DD DSN=&STORLIB,
//             DISP=SHR                 STORAGE INTERFACE LIBRARY
)ENDSEL
```

The following ISPF variables are included in the skeleton FOPDSTEP. Use the following prompts provided on the Supply Batch Parameters panel to review and adjust these variables. Each ISPF variable is shown with its panel prompt and description.

ISPF Variable: PSDVSIPU
>   **Panel Prompt:**Unit Type (Permanent)

>   **Description:**Generic unit name for permanent DB2 Load SYS* datasets.

ISPF Variable: PSDVSIPQ
>   **Panel Prompt:**Primary CYLS (Permanent)

>   **Description:**Primary and secondary allocation in cylinders for permanent DB2 Load datasets.

ISPF Variable: PSDVSITU
       Panel Prompt: Unit Type (Temporary)

       Description: Generic unit name for temporary DB2 Load datasets.

ISPF Variable: PSDVSITQ
       Panel Prompt: Primary CYLS (Temporary)

       Description: Primary and secondary allocation in cylinders for temporary DB2 Load datasets.

ISPF Variable: PSDVSILB
       Panel Prompt: Utility Block Sizes: Load Input File

       Description: Blocksize for the DB2 Load Data files.

ISPF Variable: PSDVSICB
       Panel Prompt: Image Copy File

       Description: Blocksize for DB2 Image Copy Data files.

ISPF Variable: PSDVLCL2
       Panel Prompt:COPYDDN Backup

       Description: Image Copy dataset name suffix for COPYDDN Backup.

ISPF Variable: PSDVRMT1
       Panel Prompt:RECOVERYDDN

       Description: Image Copy dataset name suffix for RECOVERYDDN.

ISPF Variable: PSDVRMT2
       Panel Prompt:RECOVERYDDN Backup

       Description: Image Copy dataset name suffix for RECOVERYDDN Backup.

FOPDDB2S

If the DB2 DSNLOAD and DSNEXIT libraries are not in the system link list, they should be specified in the skeleton FOPDDB2S. (The DB2 DSNEXIT library is necessary only if it is used.) You can specify these libraries in **DB2 Libraries** on the Supply Additional Batch STEPLIB Libraries panel.

FOPDRGSZ

The skeleton FOPDRGSZ contains the Region Size Value common to FOPDSTEP and FOPDPLUS. You can specify this value in the **Batch Region Size** prompt on the Supply Batch Parameters panel.

FOPDBMCS

The skeleton FOPDBMCS is used to generate the JCL to invoke the BMC LOADPLUS and UNLOAD PLUS Utilities. If these utilities are to be used and are not in the system link list, the STEPLIB must include a DD statement for the LOADPLUS and UNLOAD PLUS Utilities.

If your LOADPLUS job uses SMS-managed datasets, you must include SYS1.CSSLIB in your system link list or in your STEPLIB concatenation. The SYS1.CSSLIB dataset contains the SMS service routine invoked by LOADPLUS.

All load libraries used by LOADPLUS and UNLOAD PLUS must be APF-authorized. This includes the following:
- The LOADPLUS load libraries
- The load libraries containing DB2 user exits
- The DB2 load libraries
- The SMS service routine load library (SYS1.CSSLIB)

If these are not included in the link list, they will be included automatically in your STEPLIB if the batch customization was performed properly. To modify these specifications, use the Supply Additional Batch STEPLIB Libraries panel.

If UNLOAD PLUS is to be used, the load library and any Column Map load library must also be APF-authorized.

PSDFERRM Variable

The variable PSDFERRM is available in all skeletons to enable sites to impose their own restrictions on job submission. At the beginning of the skeleton expansion, this 78-character variable is cleared to blanks. If the value of the variable is not blanks at the end of the expansion, the value is displayed on the long message line on the current panel and job submission is aborted.

Due to ISPF skeleton handling, a variable can not be assigned a string containing blanks. Since blanks are important for a message to be meaningful, you can use hyphens to represent blanks in the message. All hyphens are replaced with blanks before the message is displayed on the panel.

DB2 Upgrade Considerations

Upgrading your release of DB2 can affect the Optim Directory tables. The Directory tables contain the Optim object definitions.

If a DB2 upgrade changes the CODE PAGE, you must use the Extract Process to extract all Directory tables and their object definitions prior to performing the upgrade. The Extract Process copies the specified data and object definitions to an Extract File. Use the following guidelines to create the Extract File:

- Create an Access Definition that lists all Directory tables:
  - ADB2AUDIT
  - ADDEF
  - AFCOLLCT
  - ARCHDEF
  - ARCHIDX
  - ARCHLOG
  - CDDEF
  - CMDEF
  - ENVDEF
  - LTDEF
  - MDB2PROCESS
  - PKDEF
  - RELDEF
  - RDDEF
  - TMDEF
  - CPMETER
  - CMPROC

  Indicate all tables, except the Start Table (any table can be the Start Table), as reference tables. For **Default Creator ID**, specify the Authorization ID used to create the tables (SYSFOP is the default).
- Perform the Extract for both the data and object definitions.

Once you upgrade the DB2 release, use the Insert Process to update the Directory tables with the data from the Extract File. The Insert Process copies the data and object definitions to the destination tables

specified in the Table Map. When the Table Map for the Insert Process is displayed, copy the **Src CID** value to **Dest CID**. The tables listed under **Destination Table Name** are the same as those listed under **Extract Tables**. Once you complete the Table Map, perform the Insert using Update as the processing method.

Note: If you are not certain whether a DB2 upgrade changes the CODE PAGE, it is recommended that you perform these tasks.

For details about the Extract and Insert Processes, see Data Migration in the *Move User Manual*.

Refreshing the Internal Buffer

When accessing DB2 objects, Optim loads the definitions from the catalog into an internal buffer to reduce overhead from accessing the catalog. The definitions of DB2 objects might change after the buffer has been loaded, however. Depending on the type of processing you are performing, you might need to ensure that the most current set of definitions is available.

To refresh the buffer, use the REFRESH DIRECTORY command.

```
REFRESH DIRECTORY
```

You can enter the REFRESH DIRECTORY (or its short form, REFRESH DIR) command on any panel with a command line. A message indicates "Refresh Successful."

Data Space Considerations

Optim builds certain index structures in a data space, or in multiple data spaces, as needed. Using data spaces, instead of private storage, the products can process an unlimited number of rows. However, to run large jobs, you must provide adequate resources for central, expanded, and auxiliary storage. Otherwise, you must adjust the size of your job to fit the resources you have allocated.

Each data space consists of 524,288 blocks of storage, with each block as 4K in size. As these blocks are filled, you will notice an increase in the number of real page frames being used. As central storage fills up, the real storage manager backs up the page frames in expanded storage to provide the necessary real storage. When expanded storage fills up, the real storage manager moves the page frames back into central storage, and invokes the auxiliary storage manager to move ("page out") these page frames into auxiliary storage.

You can use the following formula to estimate the number of page frames required for any given Optim process, where *rows* is the number of DB2 rows and *keysize* is the key size.

```
frames = (((rows / (4K - 28 / keysize + 4)) * 2) +
(((rows / 4K - 28 / keysize + 4)) * 2) / (4K - 28 / keysize + 4)))
```

Note: This formula assumes that your page data sets are at the desired maximum usage and that the DB2 rows are presented in random order. (Ordered data takes approximately half the calculated space as DB2 rows presented in random order.)

Performance Considerations

To ensure optimum performance, remove duplicate relationships, review cycles, and perform RUNSTATS as necessary.

Perform RUNSTATS

Optim uses the statistics provided by RUNSTATS to determine the best strategy to extract the data.

Perform RUNSTATS to ensure these statistics are current. If these statistics are not current, the most effective strategy may not be selected. For example, if you notice an extract performing a tablespace scan, and you have an index defined, RUNSTATS may not be current.

Remove Duplicate Relationships

Remove duplicate relationships and create indexes on the relationship columns.

You can identify the duplicate relationships on the Specify Relationship Usage panel. An asterisk is displayed next to the names of the duplicate relationships.

Review Cycles

Review cycles, especially those that include large tables.

You can review cycles via the Specify Relationship Usage panel. On this panel, you can control which relationships are used to extract data and specify how RI rules influence which rows are extracted.

After reviewing the Specify Relationship Usage panel, you can request an analysis of the relationship traversal path using the SHOW STEPS command. Based on the information provided by the SHOW STEPS command, you can re-specify the relationship usage. In some instances, changing the Start Table can improve efficiency.

You can also change the **Q1** and **Q2** settings on the Specify Relationship Usage panel to control whether parent rows are extracted to satisfy RI rules and whether the children of those extracted parent rows are also extracted. For further information about relationship usage, see Relationships in the *Common Elements Manual*.

Chapter 11. Establish Security

Optim allows you to establish security for objects and Archive Files or provide functional security. That is, you can select the users that have access to objects, Archive Files, and to specific Optim functions.

Object Security

Use object security to control access to objects.

Use object security to control access to the following objects.

- Access Definitions
- Compare Definitions
- Column Maps
- Table Maps
- Optim Primary Keys
- Optim Relationships
- Archive Collections
- Legacy Tables
- IMS Environment Definitions
- IMS Retrieval Definitions
- Column Map Procedures

Each object has an owner, identified by a TSO User ID. The owner is the person who last modified the object. If the object has never been modified, the owner is the person who created the object.

A security exit enables security for your site. The exit is called when the following actions are attempted:

- Create a new object
- Access an existing object
- Update an existing object
- Delete an existing object
- Change the security status of an object
- Place an object in a selection list

The exit determines whether the operation can proceed and provides a return code to indicate its determination. A site may use an exit distributed with Optim or may write a custom security exit.

Note: Security exits created with earlier releases of the Optim solutions must be recompiled to be usable in Release 5.5 and later releases.

The following discussion covers the implementation of object security, the supplied security exits (FOP2OS00 and FOP2OS01), and provides information needed to write a custom security exit.

Enabling Object Security

To enable object security, you must indicate the security exit Optim is to call and whether a user can explicitly assign security status for an object. If a user can assign security status, you can indicate the default status used when it is not assigned explicitly.

You enable object security by providing the following parameters on the Site Options panel.

Object Security Suffix

The object security exit is a load module named FOP2OS*nn*, where *nn* is a two-digit suffix. The exits distributed with Optim use 00 and 01. Custom security exits may use any two-digit number from 02 through 99 for the suffix.

Specify the two-digit suffix, to be appended to FOP2OS to indicate the name of the exit called for object security. To use a different exit, change the number appropriately or, to disable object security, clear **Object Security Suffix**.

If object security is disabled, any established security status for an object is retained and becomes effective if security status is enabled again.

Object Security Mode

An indicator for the display and user assignment of the security status. (See "Security Status").

Y   Display the security status. The user can assign the status explicitly. (Required setting if using either distributed exit.)

N   Do not display the security status. The user cannot assign the status. (However, the site administrator can assign the status.)

Default Security Status

The default security status assigned to a new object. (This setting is relevant only if **Object Security Mode** is Y.)

P   PUBLIC

R   READONLY

V   PRIVATE

For more information about the Site Options panel, see "Customizing the Site Options" on page 49.

Security Status

A user can assign security status to an object if Y is specified for **Object Security Mode** on the Site Options panel. To do this when creating or editing an object, a user must display the Object Attributes panel, using the ATTRIBUTES primary command. (Use the AT line command to display this panel from a selection list.)

From the Object Attributes panel, the user can edit the setting for **Security Status**. **Security Status** settings are:

* PUBLIC
* READONLY
* PRIVATE

The enabled security exit determines the significance of the security status setting (that is, PUBLIC, READONLY, or PRIVATE). Optim allows you to change the status setting, display the setting on various information panels, and pass the status value to the security exit, but do not use the assigned status in any other way.

The security status for objects is displayed on selection lists if the User Option for **Selection List Format** is "F" for FULL. Also, a user can display the security status for an object by using the I line command on a selection list. These selection list displays cannot be edited, however.

Distributed Security Exits

The security exits, FOP2OS00 and FOP2OS01, are distributed with Optim. FOP2OS00 determines access privilege on the basis of the owner and the security status for the object. FOP2OS01 uses System Authorization Facility (SAF) calls for security authorization and, if access privilege cannot be determined

by the SAF call, the same security as provided by FOP2OS00 applies. If neither distributed exit meets the needs of a site, the site must write its own exit routine.

Distributed Exit FOP2OS00

The distributed exit FOP2OS00 determines access privilege for an object on the basis of the owner and security status for the object. The owner is the user that last modified the object or, if the object has never been modified, the user that created the object.

A user can assign security for an object on the Object Attributes panel or the default security setting applies.

PUBLIC
> Any user can access the object for any purpose, and any user can modify the status to READONLY or PRIVATE. The object name appears on selection lists.

READONLY
> Any user can access the object, but only the owner or site administrator can edit the object, delete the object, or change its security status. The object name appears on selection lists.

PRIVATE
> Only the owner or site administrator can access the object for any purpose, including changing its security status. The object name appears on selection lists only if the requestor is the owner or site administrator.

FOP2OS00 is distributed as a load module in the SFOPLLIB library. To use the supplied load module, specify 00 as the **Object Security Suffix** on the Site Options panel.

Distributed Exit FOP2OS01

FOP2OS01 uses SAF calls for security authorization.

(It is assumed that the user reading this section has a working knowledge of SAF facilities; a detailed explanation of SAF processing is beyond the scope of this *Customization Guide*.)

FOP2OS01 is distributed in source format in the Optim sample library. To use FOP2OS01, you must assemble and link the exit into the Optim load library. To use the unmodified exit, name the load module FOP2OS01 and specify 01 as the **Object Security Suffix** on the Site Options panel. If you customize the exit to satisfy site requirements, rename the load module to FOP2OS*nn*, where *nn* is any two-digit number from 02 through 99. You can then enable the new exit by specifying the two-digit number as the **Object Security Suffix** on the Site Options panel.

The FOP2OS01 security exit establishes access privilege in three steps. Privilege is established as follows:

- If the user requesting access to the object is the owner of the object, access is granted automatically and no SAF call is made; the owner of an object always has complete access to the object.
- If the user requesting access to the object is not the owner, Optim executes an SAF call using a parameter list. (A description of the parameter list follows.)
- If the SAF call cannot establish access privilege (that is, none of the conditions for which it checks apply – see "SAF Access Rules" on page 128), the exit determines access privilege on the basis of the security status of the object and whether the user is the site administrator. The PUBLIC, READONLY, and PRIVATE rules described earlier apply in this case.

SAF Parameter List

When you use the FOP2OS01 exit for security, and a user who requests access to an object is not the owner of the object, Optim executes an SAF call using a parameter list.

The SAF call used by FOP2OS01 uses the following parameter list.

PSTRT.*objtype.ssn.objname*

objtype The type of object, which can be:

        `AD` for Access Definition

        `CD` for Compare Definition

        `CM` for Column Map

        `TM` for Table Map

        `AC` for Archive Collection

        `PK` for Optim Primary Key

        `RL` for Optim Relationship

        `LT` for Legacy Table

        `ED` for IMS Environment Definition

        `RD` for IMS Retrieval Definition

        `PR` for Column Map Procedure

ssn The AttachID (either the DB2 SubsysID or a Group AttachID).

objname

The fully qualified name of the object, consisting of two or three qualifiers separated by periods.

For example, FOPDEMO.TM1 for a Table Map, or FOPDEMO.SAMPLE.AD1 for an Access Definition. See Naming Conventions in the *Common Elements Manual* for more information about object names.

Note: Standard SAF classes protect access to datasets, for which qualifiers are limited to 8 characters. Qualifiers for Optim objects may exceed 8 characters. Thus, SAF calls require a user-defined RACF® Class, PSTRT40. The sample definition for this class is in the FOPCDTU member of the install library. This class, with a maximum length of 246 characters, can be defined by assembling and linking the sample provided. For other user-defined RACF classes, you must add the sample to your existing class definitions and assemble and link it.

SAF Access Rules

Using SAF, access to an object is denied if the user has no authority.

Access is granted according to the following rules:

User Authority
    Type of Request

Control or Alter
    CREATE

    DELETE

    GET

    INDEX

    STATUS

    UPDATE

Update
    GET

    INDEX

    STATUS

    UPDATE

Read GET

INDEX

Writing Custom Exits for Object Security

If the standard exits for object security do not meet the needs of a site, the site can implement a custom security exit.

The custom security exit must be written in Assembly language and must follow standard z/OS conventions for called subroutines. The security exit is invoked using the standard BALR R14,R15 linkage. On entrance to the exit, R1 points to a parameter list with the following two pointers:

0(R1) Pointer to a fullword reserved for use by the exit. The contents of the fullword are initialized to 0 before the first call within each execution of Optim. The contents are then preserved between calls to the exit.

4(R1) Pointer to a parameter area.

The first parameter in the list, indicated by R1, may be used to pass user information between successive calls to the exit. A special call (that is, the TERM call) to the exit at the end of a session with Optim allows the exit to perform any final processing that might be needed.

Parameter Area

When you implement a custom security exit, ensure that the exit can correctly parse and use the parameters that are passed by Optim.

The following describes the parameter area passed to the exit by Optim. Any character values that are shorter than the length are left-justified and padded with trailing blanks.

Offset: 0
Length: 6

Reason for the call. Allowable values are:

CREATE
create new object

UPDATE
update existing object

DELETE
delete existing object

GET fetch existing object

INDEX
place object on selection list

STATUS
change status of object

TERM Optim terminating (any final processing may be performed here)

Offset: 6
Length: 2

Object Type. Allowable values are:

AD Access Definition

CD Compare Definition

TM Table Map

CM Column Map

PK	Optim Primary Key
RL	Optim Relationship
AC	Archive Collection
LT	Legacy Table
ED	IMS Environment Definition
RD	IMS Retrieval Definition
PR	Column Map Procedure

Offset: 8
Length: 128

First qualifier for object name

Offset: 136
Length: 128

Second qualifier for object name

Offset: 264
Length: 128

Third qualifier for object name (blank if two-level name)

Offset: 392
Length: 3

Current security status of the object:

PUB	PUBLIC
PVT	PRIVATE
RON	READONLY

Offset: 395
Length: 3

For status change, new security status specified by the user.

Offset: 398
Length: 8

The TSO ID of the owner.

Offset: 406
Length: 1

Indicator for site administrator:

| **T** | User is site administrator |
| **F** | User is not site administrator |

Offset: 407
Length: 40

Object description.

Offset: 447
Length: 8

The TSO ID of the current user.

Offset: 455
Length: 128

The SQL ID of the current user.

Offset: 583
Length: 19

The timestamp when object was last modified.

Offset: 602
Length: 2
>   The binary abend code used with return code 16.

Offset: 604
Length: 72
>   The error message text used with return code 8.

Offset: 676
Length: 4
>   The DB2 Subsystem or Group Attach Name.

To use the custom security exit, you must assemble and link the exit, and name the load module FOP2OS*nn*, where *nn* is any two-digit number from 02 through 99. You can then enable the new exit by specifying the two-digit number as the **Object Security Suffix** on the Site Options panel.

Return Codes

When the security exit returns to Optim after each call, R15 must contain a valid return code. A system error results if the exit returns any value other than that of a valid return code.

Valid return codes are as follows:

0   Operation authorized.

8   Operation not authorized.

16   Terminate the Optim products immediately.

Return Code 8

For an INDEX call, a return code of 8 does not indicate an error condition, but simply omits that entry from the selection list. For all other calls, a return code of 8 indicates that the operation is suppressed.

For calls other than an INDEX call, Optim displays the following short TSO error message:

```
AUTHORIZATION ERROR
```

You can supply an error message for the custom security exit that becomes the long TSO error message. This message must start in position 154 of the parameter area and must be padded with trailing blanks to total 72 characters. If the first character of the error message area is a blank (which is the default) Optim provides a long error message describing the type of authorization error that occurred.

Return Code 16

For a return code of 16, Optim creates a system error and a message states that the authorization exit has terminated execution of the session. The exit may optionally place a two-byte binary abend code in the parameter area at offset 152. This code is displayed on the termination panel in AuxInfo=rc=*nn*, and may be used for diagnostic purposes.

Considerations for the Batch Utilities - Object Security

An object security exit that is enabled on the Site Options panel is also called, when appropriate, during execution of the batch utilities. For example, a request to delete an object will result in a call to the object security exit to determine if the user has the authority to perform the delete operation.

The following rules apply to the object security exit when invoked for the batch utilities.

1. The value in the exit parameter area for the TSO ID of the current user will always be the TSO ID of the user that submits the job. Note that the TSO ID specified as the last parameter in the PARM field of the EXEC card is ignored for security purposes.
2. The batch utilities do not distinguish between a normal user and a site administrator. All users are considered normal, even a user that would be site administrator during an online session. The

indicator in the exit parameter area for site administrator will always be "F" (current user is not site administrator). Therefore, if you use the FOP2OS00 exit distributed with Optim, objects designated as PRIVATE cannot be accessed for any purpose and objects designated as READONLY cannot be modified or deleted using the batch utilities.

For more information on batch utilities, see Processing Utilities in the *Batch Utilities Guide*.

Archive Security

You can establish security to control access to Archive Files.

A security exit enables archive security. The exit is called when the following actions are attempted.
- Create a new Archive File
- Browse an Archive File
- Delete an Archive File
- Restore an Archive File
- Search an Archive or Index File
- Modify or delete an Index
- Edit an Archive File description
- Edit the group name
- Place an Archive File on a selection list
- Edit the security status for an Archive File

The exit determines whether the operation can proceed and provides a return code to indicate its determination. A site may use the exit distributed with Archive or may write a custom security exit.

The following discussion covers the implementation of archive security, the supplied security exit (FOP2AS00), and provides information needed to write a custom security exit.

Enabling Archive Security

To enable archive security, you must indicate the security exit Archive is to call and whether a user can explicitly assign security status for an Archive File.

If a user can assign security status, you can indicate the default status used when it is not assigned explicitly. You enable archive security by providing the following parameters on the Site Options panel.

Archive Security Suffix

The archive security exit is a load module named FOP2AS*nn*, where *nn* is a two-digit suffix. The exit distributed with Archive uses 00; 01 is reserved for future use. Custom security exits may use any two-digit number from 02 through 99 for the suffix.

Specify the two-digit suffix to be appended to FOP2AS to indicate the name of the exit called for archive security. To use a different exit, change the number appropriately or, to disable archive security, clear **Archive Security Suffix**.

If archive security is disabled, any established security status for an Archive File is retained and becomes effective if security status is enabled again.

Archive Security Mode

An indicator for the display and user assignment of the security status. (See "Security Status" on page 133.)

Y   Display the security status. The user can assign the status explicitly. (Required setting if using either distributed exit.)

N   Do not display the security status. The user cannot assign the status.

Archive Default Security
> The default security status assigned to a new Archive File. (This setting is relevant only if **Archive Security Mode** is Y.)

> **P**   PUBLIC

> **R**   READONLY

> **V**   PRIVATE

For more information about the Site Options panel, see "Customizing the Site Options" on page 49.

Security Status

A user can assign security status to an Archive File if Y is specified for **Archive Security Mode** on the Site Options panel.

To do this, display the Archive Attributes panel by specifying the AT line command for the appropriate Archive entry on the Archive File selection list. From the Archive Attributes panel, the user can edit the setting for **Security Status**. **Security Status** settings are:

- PUBLIC
- READONLY
- PRIVATE

The enabled security exit determines the significance of the security status setting (that is, PUBLIC, READONLY, or PRIVATE). Archive allows the status setting to be changed, displays the setting on various information panels, and passes the status value to the security exit, but does not use the assigned status in any other way.

The security status is displayed on the Archive File selection list if "detail" mode is in effect. Also, a user can display the security status for an Archive File by entering the I line command on the selection list. These selection list displays cannot be edited, however.

Writing Custom Exits for Archive Security

If the standard exit for archive security does not meet the needs of a site, the site can implement a custom security exit.

The custom security exit must be written in Assembly language and must follow standard z/OS conventions for called subroutines. The security exit is invoked using the standard BALR R14,R15 linkage. On entrance to the exit, R1 points to a parameter list with the following two pointers:

0(R1)   Pointer to a fullword reserved for use by the exit. The contents of the fullword are initialized to 0 before the first call within each execution of Archive. The contents are then preserved between calls to the exit.

4(R1)   Pointer to a parameter area.

Note that the first parameter in the list indicated by R1 may be used to pass user information between successive calls to the exit. A special call (that is, the TERM call) to the exit at the end of an Archive session allows the exit to perform any final processing that might be needed.

Parameter Area

When you implement a custom security exit, ensure that the exit can correctly parse and use the parameters that are passed by Archive.

The following describes the parameter area passed by Archive to the exit. If any character values are shorter than the allowable length, spaces are added to the end of the value.

Offset: 0
Length: 6

> **CREATE**
> > create new Archive File

> **BROWSE**
> > browse Archive File

> **DELETE**
> > delete Archive File

> **RESTOR**
> > restore an Archive File

> **SEARCH**
> > search an Archive File or index

> **UPDATE**
> > edit or delete index

> **DESC** edit description

> **GROUP**
> > edit group

> **INDEX**
> > place Archive File on selection list

> **STATUS**
> > edit status

> **TERM** archive session terminating

Offset: 6
Length: 2
> Constant of AR

Offset: 8
Length: 44
> Archive File name

Offset: 52
Length: 2
> Unused

Offset: 54
Length: 8
> Group Name

Offset: 62
Length: 3
> Current security status as:

> > **PUB** PUBLIC

> > **PVT** PRIVATE

> > **RON** READONLY

Offset: 65
Length: 3
> New security status, if specified

Offset: 68
Length: 8
> TSO ID of user who created Archive File

Offset: 76
Length: 1
> Indicates whether user is the site administrator:

> **T**   current user is site administrator

> **F**   current user is not site administrator

Offset: 77
Length: 40
> Archive File description

Offset: 117
Length: 8
> TSO ID of the current user

Offset: 125
Length: 128
> SQL ID of the current user

Offset: 253
Length: 19
> Date Archive File created, as a timestamp

Offset: 272
Length: 2
> Binary abend code used with return code 16

Offset: 274
Length: 72
> Error message text used with return code 8

Offset: 346
Length: 4
> DB2 Subsystem

To use the custom security exit, you must assemble and link the exit, and name the load module FOP2AS*nn*, where *nn* is any two-digit number from 02 through 99. You can then enable the new exit by specifying the two-digit number as the **Archive Security Suffix** on the Site Options panel.

Return Codes

When the security exit returns to Archive after each call, R15 must contain a valid return code. A system error results if the exit returns any value other than that of a valid return code.

Valid return codes are as follows:

0   Operation authorized.

8   Operation not authorized.

16   Terminate Archive immediately.

Return Code 8

For an INDEX call, a return code of 8 does not indicate an error condition, but simply omits that entry from the selection list. For all other calls, a return code of 8 indicates that the operation is suppressed.

For calls other than an INDEX call, Archive displays the following short TSO error message:

```
AUTHORIZATION ERROR
```

You can supply an error message for the custom security exit that becomes the long TSO error message. This message must start in position 154 of the parameter area and must be padded with trailing blanks to total 72 characters. If the first character of the error message area is a blank (which is the default) Archive provides a long error message describing the type of authorization error that occurred.

Return Code 16

For a return code of 16, Archive creates a system error and a message states that the authorization exit has terminated execution of the session. The exit may optionally place a two-byte binary abend code in the parameter area at offset 152. This code is displayed on the termination panel in AuxInfo=rc=*nn*, and may be used for diagnostic purposes.

Considerations for the Batch Utilities - Archive Security

An archive security exit that is enabled on the Site Options panel is also called, when appropriate, during execution of the batch utilities. For example, a request to delete an Archive File will result in a call to the archive security exit to determine if the user has the authority to perform the delete operation.

The following rules apply to the archive security exit when invoked for the batch utilities:

1. The value in the exit parameter area for the TSO ID of the current user will always be the TSO ID of the user that submits the job. Note that the TSO ID specified as the last parameter in the PARM field of the EXEC card is ignored for security purposes.

2. The batch utilities do not distinguish between a normal user and a site administrator. All users are considered normal, even a user that would be site administrator during an online Archive session. The indicator in the exit parameter area for site administrator will always be "F" (current user is not site administrator).

   Therefore, if you use the exit distributed with Archive, Archive Files designated as PRIVATE cannot be accessed for any purpose and Archive Files designated as READONLY cannot be modified or deleted using the batch utilities.

For more information on batch utilities, see Processing Utilities in the *Batch Utilities Guide*.

Functional Security

Optim provides functional security that allows you to designate the users that have access to individual functions. This security exit can be written to use a third-party security system such as RACF or some other security package.

To institute functional security, perform the following steps:
1. Define resources to the security package.
2. Define user access rights for the resources as wanted.
3. Customize the security distributed functional security, if necessary.
4. Assemble the security exit, FOP2FS00, and copy it to the Optim load library.
5. Enable the security exit by specifying the Functional Security Suffix keyword in Site Options.

Define Resources

To institute functional security, the first step is to define the resources to the security package.

Define the following resources to your security package using RACF class DATASET.

Resource Name: PSTRT.OPT.*subsys***.ALLFUNC**
> **Function:** Indicates access to all functions. If a user is assigned to this resource, access to all options is automatic. However, any user not assigned to this resource must be assigned individually to the appropriate resources in the following lists. (If the function is activated, no users have access by default. If the function is not activated, all users have access to all options.)

Resource Name: PSTRT.OPT.*subsys***.OPTION**
        Function: Options (By default, all users have access to Options)

Resource Name: PSTRT.OPT.*subsys***.BROWSE**
        Function: Browse Table

Resource Name: PSTRT.OPT.*subsys***.EDIT**
        Function: Edit Table

Resource Name: PSTRT.OPT.*subsys***.BROWSEAD**
        Function: Browse using Access Definition

Resource Name: PSTRT.OPT.*subsys***.EDITAD**
        Function: Edit using Access Definition

Resource Name: PSTRT.OPT.*subsys***.ADS**
        Function: Access Definition Editor

Resource Name: PSTRT.OPT.*subsys***.DEFPK**
        Function: Primary Key Editor

Resource Name: PSTRT.OPT.*subsys***.DEFREL**
        Function: Relationship Editor

Resource Name: PSTRT.OPT.*subsys***.DEFCM**
        Function: Column Map Editor

Resource Name: PSTRT.OPT.*subsys***.DEFTM**
        Function: Table Map Editor

Resource Name: PSTRT.OPT.*subsys***.DEFADS**
        Function: Access Definition Editor

Resource Name: PSTRT.OPT.*subsys***.DEFLRM**
        Function: Legacy Record Map Editor

Resource Name: PSTRT.OPT.*subsys***.DEFENV**
        Function: IMS Environment Definition

Resource Name: PSTRT.OPT.*subsys***.DEFRD**
        Function: IMS Retrieval Definition

Resource Name: PSTRT.OPT.*subsys***.DEFEXP**
        Function: Export Process

Resource Name: PSTRT.OPT.*subsys***.DEFPR**
        Function: Column Map procedure Definition

Resource Name: PSTRT.OPT.*subsys***.DEFIMP**
        Function: Import Process

Resource Name: PSTRT.OPT.*subsys***.MIGEXT**
        Function: Extract Process

Resource Name: PSTRT.OPT.*subsys***.MIGIMP**
        Function: Import Extract File entries into the Directory

Resource Name: PSTRT.OPT.*subsys***.MIGLIST**
        Function: List Extract Files

Resource Name: PSTRT.OPT.*subsys***.MIGBROW**
        Function: Browse Extract or Control File

Resource Name: PSTRT.OPT.*subsys***.MIGRETRY**
        Function: Retry Process

Resource Name: PSTRT.OPT._subsys_**.MIGCONV**
      Function: Convert Process

Resource Name: PSTRT.OPT._subsys_**.MIGDEL**
      Function: Delete Process

Resource Name: PSTRT.OPT._subsys_**.MIGLOAD**
      Function: Load Process

Resource Name: PSTRT.OPT._subsys_**.MIGINS**
      Function: Insert Process

Resource Name: PSTRT.OPT._subsys_**.MIGCREAT**
      Function: Create Process

Resource Name: PSTRT.OPT._subsys_**.COMPSNGL**
      Function: Define single pair of tables Compare Process

Resource Name: PSTRT.OPT._subsys_**.COMPMULT**
      Function: Define multiple table Compare Process

Resource Name: PSTRT.OPT._subsys_**.COMPREPT**
      Function: Generate a Compare Process Report

Resource Name: PSTRT.OPT._subsys_**.COMPPREF**
      Function: Perform Compare Process

Resource Name: PSTRT.OPT._subsys_**.COMPBROW**
      Function: Browse Compare Process Results

Archive Menu

Resource Name: PSTRT.OPT._subsys_**.ARCHIVE**
      Function: Archive Process

Resource Name: PSTRT.OPT._subsys_**.ARCREST**
      Function: Restore Process

Resource Name: PSTRT.OPT._subsys_**.ARCDEL**
      Function: Delete After Archive Process

Resource Name: PSTRT.OPT._subsys_**.ARCLIST**
      Function: List processing

Resource Name: PSTRT.OPT._subsys_**.ARCIMP**
      Function: Import Archive Files and Populate Directory

Resource Name: PSTRT.OPT._subsys_**.ARCUP**
      Function: Update Archive File indexes

Resource Name: PSTRT.OPT._subsys_**.ARCLOAD**
      Function: DB2 LOAD for Restore Process

Resource Name: PSTRT.OPT._subsys_**.ARCSUBS**
      Function: Create Subset of an Archive File

Resource Name: PSTRT.OPT._subsys_**.ARCCREAT**
      Function: Create objects from Archive File

Resource Name: PSTRT.OPT._subsys_**.ARCCONV**
      Function: Convert Archive File

Resource Name: PSTRT.OPT._subsys_**.ARCRET**
      Function: Retry/Restart Delete after Archive or Restore Process

Resource Name: PSTRT.OPT._subsys_**.ARCBROW**
> **Function:** Browse Archive File

Resource Name: PSTRT.OPT._subsys_**.ARCLOG**
> **Function:** Browse log of Archive Processes

If a user is given authorization to compare a single pair of tables or multiple tables (COMPSNGL and COMPMULT), they are automatically given authorization to perform the Compare Process (COMPPREF), browse the compare results (COMPBROW), and generate a Compare Process report (COMPREPT).

The names listed here are used in the supplied exit FOP2FS00. If there is a naming conflict, you must modify the resource names in the exit appropriately.

Also, replace _subsys_ with the DB2 subsystem. That way, you can specify different access rights for each DB2 subsystem. The subsystem name is passed to the exit which uses the name to construct the resource names.

Assign Users

After defining the resource names to the exit, specify the user access.

Although the TSO User ID is used in the exit, both the TSO User ID and the SQLID for the user are available to the exit. You can use either. However, if you use the SQLID, you must customize the exit. The resource names, as listed in the preceding section, are predefined in the exit. These resource names must be defined to the external security product and users granted access to them.

You can assign users to PSTRT.OPT._subsys_.ALLFUNC to provide access to all functions. Users not assigned to PSTRT.OPT._subsys_.ALLFUNC must be assigned to each function individually.

Customize the functional security exit

The exit, FOP2FS00, is provided to support external security. This exit is written in Assembly language and distributed in the sample library.

Although it can be assembled and used as is, you may need to customize this exit if:
- You encountered a resource name conflict.
- You need to change the user specification from TSO User ID in the supplied exit to SQLID.
- You need to change the call to the security package. The exit is defined to use the SAF interface. If you are using some other interface, you must customize.

Note: If LOG=YES in RACCHECK parameter list, RACF requires this exit to be APF-authorized.

Writing Custom Security Exits - Functional Security

If the distributed exit does not meet the needs of a site, the site can implement a custom security exit. The custom security exit must be written in Assembly language and must follow standard z/OS conventions for called subroutines. The security exit is invoked using the standard BALR R14,R15 linkage.

On entrance to the exit, R1 points to a parameter list with the following two pointers:

0(R1)   Pointer to a fullword reserved for use by the exit. The contents of the fullword are initialized to 0 before the first call within each execution of the solution. The contents are then preserved between calls to the exit.

4(R1)   Pointer to a parameter area.

The first parameter in the list, indicated by R1, may be used to pass user information between successive calls to the exit. A special call (that is, the TERM call) to the exit at the end of an Optim session allows the exit to perform any final processing that might be needed.

Parameter Area

When you implement a custom security exit, ensure that the exit can correctly parse and use the parameters that are passed by the solution.

The following describes the parameter area passed to the exit by the solution. Any character values that are shorter than the length are left-justified and padded with trailing blanks.

Offset: 0
Length: 4
> Address of options array.

Offset: 4
Length: 128
> SQL ID of current user.

Offset: 132
Length: 4
> Current DB2 subsystem.

Offset: 136
Length: 16
> Current DB2 location.

Assemble and Copy to Load

Assemble the exit, FOP2FS00, and copy it to the load library.

Enable Exit

After you assemble the exit and copy the exit to the load library, enable the exit.

Enable the exit by specifying this keyword in the Site Options:

Functional Security Suffix
> The functional security exit is a load module named FOP2FS*nn*, where *nn* is a two-digit suffix. The exit distributed with Optim uses 00; 01 is reserved for future use. Custom functional exits may use any two-digit number from 02 through 99 for the suffix.

> Specify the two-digit suffix to be appended to FOP2FS to indicate the name of the exit called for functional security. To use a different exit, change the number appropriately or, to disable functional security, clear **Functional Security Suffix**.

> If functional security is disabled, any established security status for a function is retained and becomes effective if security status is enabled again.

For more information about the Site Options panel, see "Customizing the Site Options" on page 49.

Return Codes

When the security exit returns to the solution after each call, R15 must contain a valid return code. A system error results if the exit returns any value other than that of a valid return code.

Valid return codes are as follows:

0   Operation authorized.

8   Operation not authorized.

16   Terminate the solution immediately.

Return Code 8

For an INDEX call, a return code of 8 does not indicate an error condition, but simply omits that entry from the selection list. For all other calls, a return code of 8 indicates that the operation is suppressed.

For calls other than an INDEX call, the solution displays the following short TSO error message:

```
AUTHORIZATION ERROR
```

You can supply an error message for the custom security exit that becomes the long TSO error message. This message must start in position 154 of the parameter area and must be padded with trailing blanks to total 72 characters. If the first character of the error message area is a blank (which is the default) Optim provides a long error message describing the type of authorization error that occurred.

Return Code 16

For a return code of 16, the solution creates a system error and a message states that the authorization exit has terminated execution of the session. The exit may optionally place a two-byte binary abend code in the parameter area at offset 152. This code is displayed on the termination panel in AuxInfo=rc=*nn*, and may be used for diagnostic purposes.

Chapter 12. Use SQL DDL to Populate Directory

Some sites maintain primary and foreign key definitions in third party data dictionary products. Also, some application generators produce primary and foreign key definitions as part of the code generation process. Optim provides the Import facility to populate the Optim Directory with primary and foreign key definitions produced by such tools.

The key information is extracted from SQL DDL statements, which may be generated from a dictionary other than DB2 or may be manually created. The Import facility also generates a comprehensive summary of the Import Process upon completion.

The following steps are used to populate the Optim Directory with primary and foreign key definitions produced by an external data modeling tool.

1. Generate the SQL DDL statements in SPUFI format in a partitioned or sequential dataset.

   Note: The Import facility processes only the PRIMARY KEY and FOREIGN KEY clauses of the CREATE TABLE or ALTER TABLE statements. It does not process any DROP clauses.

2. Invoke the Import facility from the main menu by selecting Option 6 DEFINITIONS. Then select Option I IMPORT on the Choose a Definition Option menu.

3. Supply the required information on the IMPORT Process menu, and execute the Import Process by pressing ENTER.

4. An IMPORT Summary panel is displayed upon completion of the Import Process. The contents of the report file can be displayed using the REPORT command.

Using the Import Facility

To use the import facility after installing Optim, select Option 6 DEFINITIONS on the main menu.

On the Choose a Definition Option menu, select Option I to display the following panel prompting for the required information.

```
----------------------------- IMPORT Process -----------------------------
Command ===>

Definition Type ===> ALL              (See List of Types Below)

   ALL - Import All Definitions    PK  - Primary Keys
   R   - Relationships             AD  - Access Definitions
   CM  - Column Maps               TM  - Table Maps
   CD  - Compare Definitions       LT  - Legacy Tables
   ED  - Environment Definitions   RD  - Retrieval Definitions
   AC  - Archive Collections       PR  - Column Map Procedure Definitions

Input DSN   ===> EXPO

Processing Options:
  Overwrite Existing Defs      ===> NO        (Y-Yes,  N-No)
  Continue Import on Error      ===> NO        (Y-Yes.  N-No)
  Create Generic OPTIM Rels     ===> NO        (Y-Yes,  N-No)
  Create Generic OPTIM Keys     ===> NO        (Y-Yes,  N-No)
    If YES  Specify CID         ===>           (* for All)
  Object Owner Processing       ===> USER      (U-User, S-Source, E-Explicit)
    If Explicit, Specify Owner  ===>           (1-8 Alphanumeric Characters)

Enter Report File Name or Leave Blank for a Temporary File:
  Report DSN ===>
```

Figure 60. IMPORT Process

The prompts on the IMPORT Process menu are:

Definition Type

    The type of definition to process. Specify:

ALL	For All Types
AC	For Archive Collection
AD	For Access Definitions
CD	For Compare Definitions
CM	For Column Maps
ED	For Environment Definitions
LT	For Legacy Tables
PK	For Primary Keys
R	For Relationships
RD	For Retrieval Definitions
TM	For Table Maps
PR	For Column Map Procedure Definitions

    To import definitions stored in an external data modeling tool, specify ALL for **Definition Type**.

Input DSN

    The name of the sequential or partitioned data set containing the SQL DDL statements for the object definitions that are to be imported. You can specify the name of an existing data set or request a selection list by specifying an asterisk as the last character of the name.

Overwrite Existing Defs

    Indicator for what action should be taken if an imported object definition has the same name as an object definition already in the current Optim Directory. Specify YES to overwrite existing definitions in the Directory; NO to ignore the conflicting imported object definition and proceed with the next.

Definitions in the DB2 Catalog are never overwritten. If a primary key or relationship name conflicts with a name in the DB2 Catalog, an error message is written to the report file. Processing continues based on the specification for **Continue Import on Error**.

Continue Import on Error

Indicator for whether the Import Process should continue if an error is encountered when importing an object definition. If you specify YES, the object definition in error is bypassed, and the Import Process continues with the next definition. The error messages are written to the report file.

Create Generic OPTIM Rels

Indicator for converting all relationships in the input file to generic relationships.

Create Generic OPTIM Keys

Indicator for converting all Primary Keys or all Primary Keys for a specific Creator ID in the input file to generic keys during the Import Process.

If YES, Specify Creator ID or *

To genericize all Primary Keys in the input file for a specific Creator ID, enter that Creator ID at this prompt. To convert all Primary Keys in the input file, enter *.

Object Owner Processing

Determine the owner of all imported objects, as well as the date and time last modified. Specify:

U   The owner of all imported objects is the user performing the Import Process, and the date and time last modified is the current date and time. Default.

S   The owner and the date and time last modified are set to the original values from the source Export File. This is useful for creating a mirror image of an Optim Directory.

E   The owner of all imported objects is the value specified for **If Explicit, Specify Owner**. The date and time last modified is set to the original value from the source Export File.

Note: An error occurs if you specify S or E for **Object Owner Processing** and the source Export File was created using release 5.1 or earlier of the Optim products.

If Explicit, Specify Owner

If **Object Owner Processing** is E, specify the owner of all imported objects. You can specify 1 to 8 alphanumeric characters; the first character must be alphabetic. (If **Object Owner Processing** is U or S, this value is ignored.)

Report DSN

The name of the sequential data set to contain the report resulting from the current Import Process. You can specify the name of an existing data set or request a selection list by specifying an asterisk as the last character of the name.

You can specify the name of a new data set. You will be prompted for allocation information and the data set will be allocated for you.

If you do not specify a name, the report is written to a temporary file. If the report is written to a temporary file, you must use the REPORT command when the IMPORT Summary panel is displayed.

Use ENTER to execute the Import Process. Use END or CANCEL to abandon the process. If you use END, your specifications will be profiled.

Import Summary

When the Import Process is completed, the IMPORT Summary panel is displayed. The IMPORT Summary panel indicates the name of the input file and the name of the report file that contains the results of the Import Process.

```
------------------------------ IMPORT Summary -----------------------
COMMAND ===>

Input File DSN:   FOPDEMO.TEST(DDL)
Report File DSN:  Temporary

Summary of Results

   Primary Keys Processed          :  1
   Relationships Processed         :  1
   Column Maps Processed           :  0
   Table Maps Processed            :  0
   Access Definitions Processed    :  0
   Collection Definitions Processed :  0
   Compare Definitions Processed   :  0
   Legacy Tables Processed         :  0
   Environment Definitions Processed:  0
   Retrieval Definitions Processed :  0
   Column Map procedure Defintions

No error conditions were found
Enter REPORT Command to View the Report File
Press ENTER Key to Continue Processing
Enter END Command to Exit
```

Figure 61. IMPORT Summary

The IMPORT Summary panel also displays a summary of the results of the Import Process, including a list of the object definitions and the total number of each that were added to the Directory. A message on the panel indicates whether any error conditions were encountered during the process.

REPORT Command

Use the REPORT command on the IMPORT Summary panel to display the contents of the report file in the ISPF browse facility. The report lists the imported object definitions by category and the names of each imported object, as well as any error messages.

```
******************************** Top of Data **************************
                        IMPORT Process Report
             Created on Thursday, Nov 18, 2010 at 11:03 AM
                         File: FOPDEMO.TEST(DDL)
RELATIONSHIP IMPORTED     : FOPDEMO.ORDERS.CUSORD
PRIMARY KEY IMPORTED      : FOPDEMO.CUSTOMERS
****************************** Bottom of Data ************************
```

Figure 62. IMPORT Process Report

Use END to terminate the ISPF browse session and return to the IMPORT Summary panel. From the IMPORT Summary panel, use ENTER to return to the IMPORT Process menu or use END to return to the Choose a Definition Option menu.

Chapter 13. Interface with the BMC Catalog Manager

If the BMC Catalog Manager is installed at your site, you may establish an interface between Access and the Catalog Manager. The interface allows you to use Access to browse and edit the tables, views, synonyms, and local aliases from the lists displayed by the Catalog Manager.

Notes:

- In order to edit or browse a synonym, the user must have SYSADM authorization or the synonym owner must be an authorization ID of the process.
- Access does not support remote aliases.

Modify the Command Table

The BMC Catalog Manager Command Table, ACTCOMND, must contain appropriate entries for the BR, BRS, ED, and EDS commands. (BR and ED display the data. BRS and EDS display a prompt for selection criteria prior to the data.) Review the $xxxCOMD member in the BMC install library which is used to assemble and link ACTCOMND, and modify these entries if necessary.

The entries for the commands should match the following:

```
*BROWSE TB OR VW (ACCESS DB2 TABLE/VIEW BROWSE)
      $ACTCMD BR,#BR,HELP=FOP7BB,LOAD=FOPD2BMC,CLIST=YES,       X
           OBJECTS=(TB+VW+SY),ADDOBJ=(AL)
*BROWSE TB OR VW (ACCESS DB2 TABLE/VIEW BROWSE)
      $ACTCMD BRS,#BRS,HELP=FOP7BC,LOAD=FOPD2BMC,CLIST=YES,     X
           OBJECTS=(TB+VW+SY),ADDOBJ=(AL)
*EDIT TB OR VW (ADB2 TABLE EDIT)
      $ACTCMD ED,#ED,HELP=FOP7BE,LOAD=FOPD2BMC,CLIST=YES,       X
           OBJECTS=(TB+VW+SY),ADDOBJ=(AL)
*EDIT TB OR VW (ADB2 TABLE EDIT)
       $ACTCMD EDS,#EDS,HELP=FOP7BD,LOAD=FOPD2BMC,CLIST=YES,    X
            OBJECTS=(TB+VW+SY),ADDOBJ=(AL)
```

If you are creating these entries manually, be sure to include the continuation indicator in column 72.

If your version of the Catalog Manager Command Table does not include these entries, insert them into the table in alphabetical order by command name. Ensure that the $ACTCEQU copy member contains equates for the #BR, #BRS, #ED, and #EDS command numbers or replace these symbols with the appropriate numeric constants before assembling. (See member FOP$BMCO in the Optim sample library for sample entries.)

Assemble and linkedit the updated $xxxCOMD member.

In addition to identifying the commands in the member FOP$BMCO, you need to perform the following two steps to use the interface.

- Ensure that the following CLISTs are in the CLIST library, which is accessible when the Catalog Manager is running.

  ```
  FOPD2BMC
  FOP
  FOPCUST
  FOPAM00
  FOPGPLAN
  ```

  If these CLISTs are not accessible, copy them to the appropriate CLIST library.

- Copy the help panels, FOP7BB, FOP7BC, FOP7BD, and FOP7BE from the panel library into the Catalog Manager panel library. Depending on the release of the Catalog Manager that you are using, these may already be present.

Once these steps are completed, the interface is available. Any time a list of tables, views, synonyms, or aliases is displayed by the Catalog Manager, you can enter BR, BRS, ED, or EDS as line commands to browse or edit the selected object.

Chapter 14. Convert an Extract File

Extract files can be converted to Archive Files and used in any Archive processes. Use the following JCL to convert an Extract File.

```
//              EXEC PGM=FOPMMAIN,REGION=0M,
//              PARM=' CON UTILITY subsys planname sqlid userid '
//SYSPRINT      DD SYSOUT=*
//SYSTERM       DD SYSOUT=*

//PSDFDFLT      DD DSN=your.loadlib(FOPMDFLT)      Default Site Options
//PSDFEXTR      DD DSN=dsname,DISP=SHR             Input Extract File
//PSDFARCH      DD DSN=dsname,DISP=SHR             Output Archive File

//SYSIN
              DD *

EXT_TO_ARC AD adname
/*
```

Notes:

- Specify fully qualified names for both the input Extract File and output Archive File datasets.
- The keyword AD in the EXT_TO_ARC control statement is optional. You can use this keyword to specify the name of an existing Access Definition. Archive checks internally to ensure that the Access Definition matches the contents of the input Extract File. If the Access Definition and input Extract File match, the Access Definition is copied into the output Archive File. If the Access Definition and input Extract File do not match, or if the Access Definition does not exist, the job will terminate with a message indicating the error.
- If you do not include the AD keyword, Archive will create a default Access Definition, named EXTRACT.TO.ARCHIVE. The generated Access Definition reflects the contents of the input Extract File, but does not include the original selection criteria.
- The process does not modify the Extract File used as input; the file remains unchanged in its original location.
- Object Definitions for tables are included in all Extract Files. If definitions for additional objects, such as keys, indexes, and so on, are not stored in the Extract File, you cannot use the converted Archive File to create them. These objects must be explicitly created in a database.
- You must execute this job for each Extract File you wish to convert.

Chapter 15. Customize Aging Rules

A member, FOP2RUSA, is distributed with Optim to define special days and aging rules.

The distributed tables contain common default holidays and weekend specifications. The tables can be customized to suit individual countries and specific sites.

Sample rule tables are distributed in the Optim install library for a number of other countries, as shown in the following section. These tables are designed to specify standard holidays for various countries. Modify these tables to suit your site.

Table 2. Aging rule tables by country

Rule Table	Country
FOP2RAUS	Austria
FOP2RDNK	Denmark
FOP2RFRA	France
FOP2RDEU	Germany
FOP2RIRL	Ireland
FOP2RSWE	Sweden
FOP2RSWE	United Kingdom
FOP2RUSA	United States

The RULEDEF entries build a table that is loaded by Optim. These aging rules can be specified for the columns that are to be aged, either individually within a Column Map or globally for a process. When aging is performed, Optim checks the table to determine the attributes of the aged day. Based on the aging rules, it then adjusts the date if needed.

This section documents the steps required to customize the tables by defining special business days and aging rules using the RULEDEF macro.

Distributed Aging Rules

Each time a date is aged, Optim evaluates the result and, based on a set of rules, adjusts the aged date. For example, rules can be defined such that if an aged date falls on Saturday, it is adjusted to a working day, such as Friday or Monday. To provide this semantic aging capacity, Optim provides the RULEDEF macro. Use this macro to define a date table that identifies categories of dates and to specify the rules to be applied to these categories.

When an aged date falls on a specified category (for example, holidays, weekend, and so on), the aging rules determine how to adjust the date (for example, forward, backward, and so on).

The RULEDEF entries in FOP2RUSA adjust the dates for many common situations such as the date falling on a Saturday, Christmas, and so on.

Date Table

The date table defined by FOP2RUSA in the RULEDEF macro identifies the dates and types of dates that require adjustment.

The following is a list of the specifications in the distributed table to adjust dates based on standard United States holidays, weekend days, and payday dates:

WEEKEND
>Saturday and Sunday.

HOLIDAY
>- New Years Day
>- Presidents Day
>- Good Friday
>- Memorial Day
>- Independence Day
>- Labor Day
>- Thanksgiving Day
>- Christmas Day
>
>Since Easter and the related holidays such as Good Friday vary each year, Optim automatically calculates these dates.
>
>For those dates that occur at irregular intervals, Optim allows you to specify explicit dates. Commented definitions for Rosh Hashanah are included in FOP2RUSA as an example.

PAYDAY
>The fifteenth (15) and last work day of each month.

QUARTER
>The first work day of every quarter.

Rules

The rules to apply to the dates in the date table are also defined in FOP2RUSA.

The following rules are defined using the date table built for standard United States holiday, weekends, and payday dates.

NEXTWORKDAY
>When the date falls on a weekend day or a holiday, adjust the date to the next workday. For example, if the date is a Saturday or Sunday, adjust the date to Monday.

PREVWORKDAY
>When the date is a weekend or a holiday, adjust the date to the previous workday. For example, if the date is a Saturday or a Sunday, adjust the date to the previous Friday.

CLOSWORKDAY
>When the date is a weekend or a holiday, adjust the date to the closest workday. If the date is a Saturday, adjust the date to the previous Friday; if the date is a Sunday, adjust the date to Monday.

NEXTWEEKDAY
>When the date falls on a weekend day, adjust the date to the next weekday. For example, if the date is a Saturday or Sunday, adjust the date to Monday.

PREVWEEKDAY
>When the date is a weekend, adjust the date to the previous weekday. For example, if the date is a Saturday or a Sunday, adjust the date to the previous Friday.

CLOSWEEKDAY
>When the date is a weekend, adjust the date to the closest weekday. If the date is a Saturday, adjust the date to the previous Friday; if the date is a Sunday, adjust the date to Monday.

NEXTSAMEDOW

When the date does not fall on the same day of the week as the original date, adjust the date to fall on the next same day of the week.

PREVSAMEDOW

When the date does not fall on the same day of the week as the original date, adjust the date to fall on the previous same day of the week.

CLOSSAMEDOW

When the date does not fall on the same day of the week as the original date, adjust the date to fall on the closest same day of the week.

NEXTSTRTQTR

When the date does not fall on a date defined as the start-of-quarter day, adjust the date to the next quarter. For example, if the date is March 15th, adjust the date to April 1 (the start of the next quarter). Note that this is further adjusted if the first day of the quarter is not a workday.

PREVSTRTQTR

When the date does not fall on a date defined as the start-of-quarter day, adjust the date to the previous quarter. For example, if the date is March 15th, adjust the date to January 1 (the start of the previous quarter.) Note that this is further adjusted if the first day of the quarter is not a workday.

CLOSSTRTQTR

When the date does not fall on a date defined as the start-of-quarter day, adjust the date to the closest quarter. For example, if the date is March 15th, adjust the date to April 1 and if the date is January 15th, adjust the date to January 1. Note that this is further adjusted if the first day of the quarter is not a workday.

NEXTPAYDAY

When the date does not fall on a date that is defined as a payday, adjust the date to the next payday. For example, if the date is March 5th and payday is defined as the first and the fifteenth of every month, adjust the date to March 15th.

PREVPAYDAY

When the date does not fall on a date that is defined as a payday, adjust the date to the previous payday. For example, if the date is March 5th and payday is defined as the first and fifteenth of every month, adjust the date to March 1st.

CLOSPAYDAY

When the date does not fall on a date that is defined as a payday, adjust the date to the closest payday. For example, if the date is March 5th and payday is defined as the first and fifteenth of every month, adjust the date to March 1st, the closest payday.

To use these rules, you specify the rule name in **Aging Rule** on the Aging Specifications or Aging Parameters panels. Optim checks the table built by the RULEDEF macros to determine whether an aged date falls on a date in the date table and if so, use the rule to adjust date.

Modifying Aging Rule

The Optim rules table is an assembler module. It contains the set of specifications for the date table and aging rules. You can modify the date table and any distributed aging rules; however, your modifications must conform to the expected format.

The basic structure of the distributed rule table, FOP2RUSA, demonstrates the format.

```
FOP2RUSA RULEDEF TYPE=START                    (Start
Statement)
     RULEDEF TYPE=              (Date definitions)
         . . .
```

```
     RULEDEF RULE=              (Rule definitions)
       . . .
       RULEDEF TYPE=END          (End Statement)
```

The contents of a rule table are divided into two sections.

- Date definition statements (for holidays and other special days), to create the table identifying the dates that require special handling.
- Rules for determining if an aged date meets the semantic requirements and, if not, the specifications for adjusting the aged date.

Date and Rule definitions can be intermixed as long as all dates used in a rule are defined prior to the rule.

FOP2RUSA Format

The format complies with S/390® Assembler standards.

- Separate operands with a comma. If you omit the comma, the remaining specifications are handled as comments. Blanks are allowed after the comma only for the last operand on the line and there is a continuation.
- Place any non-blank character in position 72 for continuation.
- Use an asterisk in the first position of the line to indicate a comment.
- Lists of operands must be enclosed in parentheses.

As an example of the format, the following discusses some of the sample table, FOP2RUSA, distributed for standard United States holidays with Saturday and Sunday as the weekend.

Date Table

You identify each date in the table by type and various other parameters. The parameters can be specified in any order.

The syntax for an entry into the date table is:

```
RULEDEF  TYPE=type,
               DESC=desc,
               DOM=dom,
               DOW=dow,
               MONTH=month,
               YEAR=year,
               ADJUST=adjust,
               ABSORB=absorb,
               DURATION=duration,
               RESOLVE=resolve,
               REOCCURS=reoccurs
```

type   Name assigned to the date type. Specify:

START   Identify the beginning of the RULEDEF specifications.

A label is required on the TYPE=START statement. It identifies the rules table name.

MATCH is an optional operand for START only. Use it to indicate the adjustment direction when CLOSEST is specified in a rule and the amount to adjust forward and backward are equal. Specify:

MATCH=NEXT to adjust forward (default)

MATCH=PREVIOUS to adjust backward

END   Identify the end of the RULEDEF specifications. No other clauses are used.

HOLIDAY
   Defined type. Any number of holidays can be defined.

WEEKEND
   Defined type. A maximum of two entries can be assigned as weekend. They need not be consecutive.

name   Required. 1 to 11-character user-defined name. You can specify up to 14 different user-defined names. (To define more than 14 different names, use multiple rule tables.)

These names enable you to categorize types of dates that require the same handling. For example, all paydays should be processed the same. You can specify any number of entries for a single user-defined name.

desc   Optional. 1 to 40-character description of the date. The description must be delimited by single quotes. This is truncated if the length is greater than 40 characters.

dom   Required. A value indicating the day of the month. This can be specified as:

   n   A numeric value from 1 through 31. Note that the value must be appropriate for the month if one is specified using the `MONTH` parameter.

   LAST   Last instance of a specific day.

   EVERY   Every instance of a specific day.

dow   A value indicating the day of the week.

   DAY   Indicates that the value specified for `DOM` is a specific date. For example, `DOM=4` and `DOW=DAY` indicates the fourth day of the month.

   day   A specific day of the week as: `MONday`, `TUEsday`, `WEDnesday`, `THUrsday`, `FRIday`, `SATurday`, `SUNday`.

   The full name or the abbreviation (indicated in uppercase) can be specified.

   Indicates that the value specified with `DOM` represents the occurrence of that day of the week. For example, `DOM=4` and `DOW=MON` indicates the fourth Monday of the month.

   WORKday
      Any day not specifically defined as `WEEKEND` or `HOLIDAY`.

      Indicates that the value specified with `DOM` represents the occurrence of a workday. For example, DOM=4 and DOW=WORK indicates the fourth workday of the month.

   EASter   Easter day. (Use this to calculate Good Friday and Easter Monday.)

   Note: Day of the week (day), `WORKday` and `EASter` are optional. If not specified, `DAY` is assumed. `DOW` is invalid if you specify `REOCCURS`.

month   Name of the month. Specify any month in the year. Either the full name or the first three characters are valid.

This is only required if REOCCURS is specified. If not specified all months are assumed. You can also specify MONTH=MONTH to indicate all months.

year A specific year. Specify any four-digit value within the range 1582 through 3999. This can be used to specify an explicit date.

This is only required if REOCCURS is specified. If not specified, all years are assumed.

adjust Number of days by which to adjust a date. Specify a value from 1 through 366. Use + or - to indicate whether the date is to be incremented or decremented (for example, +10 or -10).

This value must not adjust the date to a different year. Therefore, the value is compared to the number of days remaining in the year (if incremented) or the number of days that have passed (if decremented). The specified value must be less than or equal to the number of days with which it is compared.

absorb Optional. Specifies handling when the defined date falls on a Tuesday or a Thursday. Specify:

YES If the date falls on Tuesday, "absorb" the previous Monday. (Monday should also be included.) Similarly, if the date falls on Thursday, "absorb" the following Friday. (Friday should also be included.)

NO Do not "absorb" contiguous days. Default.

For example, if Christmas falls on Thursday, this indicates whether Friday is considered a holiday, too.

duration
Optional. Number of days the specification spans. Entries cannot span years. An error occurs if the value specified for DURATION results in a date beyond December 31.

resolve
Optional. Specifies how to handle adjusting the date in the table if it falls on a defined WEEKEND or HOLIDAY. For example, if you define a type as PAYDAY, use resolve to ensure that the dates defined as PAYDAY exclude weekends and holidays. Specify:

CLOSWD Adjust to closest workday.

PREVWD Adjust to closest previous workday.

NEXTWD Adjust to closest next workday.

reoccurs
Specifies that the type should be calculated as a "recurring" or repeating date. For example, at some sites payday is every two weeks regardless of the date.

Specify the frequency of the recurrence as one of the following where *n* is the number of the unit:

*n***,YEAR** Number of years.

*n***,MONTH**
Number of months.

*n***,WEEK** Number of weeks.

*n***,DAY** Number of days. Default.

The start date is specified using the MONTH, DOM, and YEAR parameters. These parameters are required for REOCCURS.

You can also specify an end date. If you do, you must supply all of the following:

EMONTH Ending month. Specify any month in the year. Either the full name or the first three characters are valid.

EDOM Ending day of the month in the range 1 through 31. The value must be appropriate for the EMONTH value.

If you do not specify an end date, Optim assumes it to be 12/31/3999. If you specify an end date along with the recurrence value, supply a comma to separate the values and delimit them with parentheses.

If REOCCURS is specified, DOW is invalid.

If RESOLVE is specified along with REOCCURS, the date may be adjusted. However, the next instance of the REOCCURS date is computed from the original REOCCURS date, not the RESOLVE date. (If REOCCURS=(2,DAY),RESOLVE=PREVWD is specified and the REOCCURS date falls on Saturday, it is adjusted to the previous Friday. The next occurrence is determined by adding 2 days to Saturday, the original REOCCURS date.)

RULEDEF Examples

The following examples demonstrate using RULEDEF to define several date types and provide examples of each of the operands.

WEEKEND

To define Saturday and Sunday as WEEKEND, you specify each day with a separate statement.

```
RULEDEF TYPE=WEEKEND,DOW=SATURDAY
RULEDEF TYPE=WEEKEND,DOW=SUNDAY
```

By default, Saturday and Sunday are defined as WEEKEND. If you intend to use these days, you need not re-specify them. However, if you want to designate other days as WEEKEND, you can select any two days by specifying the name of the day as the DOW operand. The days need not be consecutive and you can specify only two.

HOLIDAY and others

To define days as HOLIDAY or other types, you specify each with a separate statement.

The following discusses the specifications for holidays and other types according to a specific date, specific year, calculated date, and user-defined date.

Specific Date

Some holidays occur on the same day every year. For example, to designate New Year's Day (January 1) as HOLIDAY, you specify the month and the day in the statement.

```
RULEDEF TYPE=HOLIDAY,MONTH=JANUARY,DOM=1
```

Frequently, holidays require additional specifications. If New Year's Day falls on a Tuesday or Thursday, you may want to extend the holiday to include either Monday or Friday appropriately. Add ABSORB to the statement.

```
RULEDEF TYPE=HOLIDAY,MONTH=JANUARY,DOM=1,ABSORB=YES
```

Specific Year

For holidays that are recalculated each year, you can specify a specific date.

For example, Rosh Hashanah occurs on a different date every year. You specify each date by year individually.

```
RULEDEF TYPE=HOLIDAY,MONTH=OCT,DOM=2,DOW=DAY,YEAR=1997
RULEDEF TYPE=HOLIDAY,MONTH=SEP,DOM=21,DOW=DAY,YEAR=1998
```

Calculated Date

Some holidays fall on a different date every year. For example, to designate Memorial Day as HOLIDAY, you specify the day by identifying the occurrence of the day in a specific month.

The following statement specifies the last Monday in May.

```
RULEDEF TYPE=HOLIDAY,MONTH=MAY,DOM=LAST,DOW=MONDAY
```

Some holidays are dependent on others. The following statement specifies that Good Friday is a holiday.

```
RULEDEF TYPE=HOLIDAY,DOW=EASTER,ADJUST=-2
```

Sometimes the holidays extend beyond a single day. For example, Thanksgiving usually includes Friday as well as Thursday. Add `DURATION` to the statement.

```
RULEDEF TYPE=HOLIDAY,MONTH=NOV,DOM=4,DOW=THURSDAY,DURATION=2
```

User-Defined

User-defined types (those other than `WEEKEND` and `HOLIDAY`) use the same operands.

For example, you can use two statements to define payday as the 15th and the last day of every month. Additionally, if the 15th or the last day of the month falls on a `HOLIDAY` or a `WEEKEND`, you can specify whether your employees are paid on the previous workday (`PREVWD`) or next workday (`NEXTWD`).

The following statements establish that if payday falls on a holiday or a weekend, the date is adjusted to the previous workday before.

```
RULEDEF TYPE=PAYDAY,DOM=15,RESOLVE=PREVWD
RULEDEF TYPE=PAYDAY,DOM=LAST,RESOLVE=PREVWD
```

Some sites establish a specific day of the month as a payday. For example, the following statement establishes payday as the next to last Friday in any month.

```
RULEDEF TYPE=PAYDAY,DOM=LAST,DOW=FRIDAY,ADJUST=-7
```

If you want to include a span of several days, use `DURATION`. For example, some sites close for a week starting with Christmas Eve. You need to be able to handle dates falling on that week. To do so you can specify the type and define the dates. To specify a week beginning with Christmas Eve as `VACATION`, use the following statement.

```
RULEDEF TYPE=VACATION,DOM=24,DOW=DAY,MONTH=DEC,DURATION=7
```

Reoccurring Dates

For convenience, you can specify a rule to handle re-occurring dates.

For example, a single rule can define the first day of every quarter. To identify the first workday day of every third month, start with the first month of 1990 as the first day of each quarter. Use the `RESOLVE` parameter to adjust resulting date to a workday if it is not.

```
RULEDEF TYPE=QUARTER,
DOM=1,MONTH=JAN,YEAR=1990,REOCCURS=(3,MONTHS),RESOLVE=NEXTWD
```

The next `REOCCUR` date is calculated from the original date, not the `RESOLVE` date.

Defining Rules

After you have defined the entries in the table for each declared type, you define rules to be applied when an aged date must or must not fall on the day defined by a date type.

You can define any number of rules. For each rule, the parameters can be specified in any order.

The syntax for a rule statement is:

```
RULEDEF      RULE=rule,
    DESC='string'     AVOID=avoid,
    SEARCH=search,
    SDOWACTION=sdowaction      DIR=dir
```

rule   A 1- to 11-character user-assigned name for the rule. These names are specified in **Aging Rule** on the Aging Specifications and Aging Parameters panels.

desc   A 1- to 40-character string describing the rule. The string must be delimited by quotes. This is displayed on the rule name selection list. This is truncated if the length is greater than 40 characters.

avoid   List of date types. This list can include `HOLIDAY`, `WEEKEND`, and any user-defined date types.

        If the calculated date falls on a day defined by one of the listed date types, search for the next or previous day that is not defined by a listed type.

        (Next or previous is defined by the `DIR` parameter. If not specified, next is the default.)

        Use a comma, with no intervening spaces, to list multiple date types as in `HOLIDAY,WEEKEND`. Enclose the list in parentheses.

search   List of date types. This list can include `HOLIDAY`, `WEEKEND`, and any user-defined date types. Use the reserved word `SAMEDOW` to indicate same day of week processing. This processing is further governed by the specifications for `SDOWACTION` (see *sdowaction*).

        The calculated date must be defined by one of the listed date types. If `AVOID` is also specified, the calculated date must be included in the `SEARCH` list and not included in the `AVOID` list.

        Use a comma, with no intervening spaces, to list multiple date types as in `HOLIDAY,WEEKEND`. Enclose the list in parentheses.

sdowaction

        Action to be taken if `SAMEDOW` is specified on `SEARCH`. During execution of a rule with `SAMEDOW` specified on `SEARCH`, the initial step is to resolve the date forward or backward as determined by the `DIR` operand to the same day of the week as the original input date. Subsequent processing is determined by `SDOWACTION`. Specify:

        MATCH   Adjusted date must land on the same day of the week as the original date. If the initial target date does not satisfy the other `SEARCH` and `AVOID` criteria, it is adjusted a week at a time until it does.

        NEXTRULE
             After date is adjusted to the same day of the week, if it does not conform to the other criteria, it is adjusted forward to the next date that satisfies the other `SEARCH` and `AVOID` criteria.

        PREVRULE
             After date is adjusted to the same day of the week, if it does not conform to the other criteria, it is adjusted backward to the previous date that satisfies all of the other `SEARCH` and `AVOID` criteria.

        If not specified, `MATCH` is the default.

dir   Specify how the date is to be adjusted if the calculated date does not meet the criteria of `AVOID` and `SEARCH`. Specify:

        CLOSEST
             Date is adjusted to the closest date meeting the `SEARCH` criteria not included in the `AVOID` list. If the number of days from previous and next is the same, the next date is assumed. However, you can override this default by specifying the `MATCH` operand on the `RULEDEF=START` statement.

        PREVIOUS
             Date is adjusted to the closest previous date not included in the `AVOID` list and included in the `SEARCH` list, if defined.

        NEXT   Date is adjusted to the closest next date not included in the `AVOID` list and included in the `SEARCH` list, if defined.

        If not specified, `NEXT` is the default.

RULEDEF Examples

The following examples demonstrate using RULEDEF to define the rules.

They provide examples for using each of the operands.

- Avoid Non-Workdays

  To adjust a date to the next workday, use AVOID to list the date types and DIR to specify the direction. That is, using the following definition, if the aged date falls on a holiday or weekend, it is adjusted. For example, if Christmas falls on a Friday, the date is adjusted to the next workday, Monday.

  ```
  RULEDEF RULE=NEXTWORKDAY,AVOID=(WEEKEND,HOLIDAY),DIR=NEXT
  ```

- Ensure Payday is a Workday

  To ensure that the adjusted date is a payday and if not, adjust the date to the previous payday, specify SEARCH to ensure that only paydays are the target and DIR to specify the direction of the adjustment. For example, using the following definition, if the aged date is not a payday, the date is adjusted to the previous payday.

  ```
  RULEDEF RULE=PAYDAY,SEARCH=(PAYDAY),DIR=PREVIOUS
  ```

- Ensure Same Day of the Week

  To ensure that the adjusted date falls on the same day of the week as the original input date, specify SEARCH as SAMEDOW. For example, if the aged date does not fall on the same day of the week, it is adjusted (by default) to the next occurrence of the same day of the week.

  ```
  RULEDEF RULE=NEXTSAMEDOW,SEARCH=(SAMEDOW),DIR=NEXT
  ```

- Handle Start-of-Quarter

  To ensure that the adjusted date is the start of a quarter, you need to specify SEARCH to limit the date type to QUARTER.

  ```
  RULEDEF RULE=CLOSSTRTQTR,                        X
          SEARCH=QUARTER,                  X
          DIR=CLOSEST
  ```

Assemble and Link the Table

After you have modified the table, you must assemble and link the table to prepare it for use.

Modify the sample JCL provided in JCLASM in the install library to perform these steps. Here is a copy of the sample JCL.

```
//* JOB  )) PUT A VALID JOB CARD HERE                        <===
//*-------------------------------------------------------------*
//*   ASSEMBLE IBM CORPORATION   'FOP2RUSA' SEMANTIC RULE TABLE OR  *
//*                              'FOP9SAPF' DATE FORMAT TABLE      *
//*-------------------------------------------------------------*
//JCLASM  PROC MEMBER=ruletable,                            <===
//           QUAL1=qualifier1                               <===1
//           QUAL2=qualifier2                               <===2
//ASM      EXEC PGM=ASMA90,PARM='NODECK,OBJECT'
//SYSLIB   DD  DSN=&QUAL1..&QUAL2..SFOPINST,DISP=SHR
//SYSLIN   DD  DSN=&&OBJ,UNIT=VIO,SPACE=(TRK,(10,10)),DISP=(,PASS)
//SYSIN    DD  DSN=&QUAL1..&QUAL2..SFOPINST(&MEMBER),DISP=SHR
//SYSUT1   DD  UNIT=VIO,SPACE=(TRK,(10,10))
//SYSUT2   DD  UNIT=VIO,SPACE=(TRK,(10,10))
//SYSPRINT DD  SYSOUT=*
//SYSTERM  DD  SYSOUT=*                                     <===3
//*-------------------------------------------------------------*
//*         LINK THE TABLE, IF IT ASSEMBLED CLEANLY          *
//*-------------------------------------------------------------*
//LKED     EXEC PGM=HEWL,COND=(8,LT,ASM),PARM='AMODE=31,RMODE=ANY'
//SYSLIN   DD  DSN=&&OBJ,DISP=(OLD,DELETE)
//SYSLMOD  DD  DSN=&QUAL1..&QUAL2..SFOPLLIB(&MEMBER),DISP=SHR  <===4
//SYSUT1   DD  UNIT=SYSDA,SPACE=(TRK,(10,10))
//SYSUT2   DD  UNIT=SYSDA,SPACE=(TRK,(10,10))
```

```
//SYSPRINT DD  SYSOUT=*
//*--------------------------------------------------------------*
//       PEND
//*--------------------------------------------------------------*
//JCLASM  EXEC JCLASM
//
```

To use this JCL, supply a valid job card and then modify the JCL as follows:

1. Specify the name of the rule table for MEMBER. The default rule table is FOP2RUSA.

2. Specify the high-level qualifiers for the Optim libraries.

3. Specify the source library containing the rule table for SYSIN. If you use the default rule table, it is in the Optim install library.

4. Specify the load library for SYSLMOD.

Multiple Rule Tables

To use multiple rule tables, execute this JCL to assemble and link each rule table. Store the rule tables in the same library.

Chapter 16. Customize Date Format Tables

Date formats specify the format of a date field including the individual components (year, month, day) that make up the field. These are defined internally as a list of formats in a single format table, FOP9SAPF. New formats can be added easily to the table without requiring exit coding for each individual format.

The user specifies each format by its name in a Column Map, for an individual column, or on the Aging Parameters panel for a process. (Aging can be performed for Insert, Load, and Convert Processes.) Users can display a selection list of values appropriate for the specific field type by entering an asterisk in the field.

The name is qualified by the data type of the field. Thus, Optim supports a date format of MMDDYY as a character field and as a decimal field even though these are processed differently internally.

Optim provides an extensive list of date formats; however, you can augment them. For the most part, you can add formats to the list and need not write exit routines for unique data types.

Skipped Dates

Frequently, values that are not valid dates are inserted into date fields to indicate special handling or conditions.

Rather than treat these non-date values as invalid or as errors, you can direct Optim to skip them. That is, when a field contains the special value, the field is bypassed. Since there is no error, processing continues with the next date field. The user can indicate whether skipped values are noted in the Aging Report and if records with skipped dates are written to the output file.

By default, Optim automatically skips the field when it contains all spaces, hex zeros (low-values) or hex 'FF' (high-values). If the field does not contain those values, Optim parses the field based on the specified date format. To identify the values to be skipped either by individual format or globally, you also define them in FOP9SAPF. Thus, Optim interprets both format and skip values in a single definition member.

FOP9SAPF

The formats are defined by specifications in the member named FOP9SAPF distributed in the Optim install library.

These specifications are coded using a macro language to define the format information and the fields that make up the date field.

All statements and parameters must comply with S/390 Assembler standards.

- Separate operands with a comma. If you omit the comma, the remaining specifications are handled as comments. Many parameters are positional, therefore you must include a comma with no intervening blanks for parameters that are omitted.
- Blanks are allowed after the comma only for the last operand on the line and there is a continuation.
- Place any non-blank character in position 72 for continuation.
- Use an asterisk in the first position of the line to indicate a comment.
- Lists of operands must be enclosed in parentheses.

There are comments included in FOP9SAPF to show where site-specific entries should be added. The format specifications are defined separately for each data type of the underlying field. A data type is also included for DB2 dates and timestamps.

You define the format using FORMAT and PCODE statements and skip values using SKIP statements. You can specify OVERRIDE statements to establish default settings.

The format source is assembled and link edited using standard assembly procedures. Sample JCL for assembling the date formats is provided in member JCLASM in the install library and discussed in "Assemble and Link the Table" on page 172.

The remainder of this section discusses how to customize FOP9SAPF by specifying FORMAT, PCODE, SKIP, and OVERRIDE statements.

Date Components

The formats distributed with Optim are named with the following characters to represent the date component. This convention makes it easier for the user. Site-defined formats can be named as desired.

CC   Two-digit century.

YY   Two-digit year without century.

YYY   Three-digit year relative to 1900.

YYYY or CCYY
           Four-digit year.

MM   Two-digit month.

MMM   Three-character abbreviation for the name of the month, such as Jan or JAN.

DD   Two-digit day.

DDD   Three-digit Julian day.

DDDDDD Lilian date (number of days since Oct. 14, 1582).

/   Slash in date.

-   Dash in date.

*   Any delimiter in date.

U   Unsigned decimal field. It precedes the format.

Date Format Examples

Optim supports a variety of internal date storage schemes.

For instance, a YYMMDD date field may be stored as a:
- 6-byte character field
- 3-byte packed unsigned field
- 4-byte packed-decimal field
- 4-byte packed-decimal field with high-order bits indicating century

Character formats and numeric formats, as either packed decimal or binary fields, are allowed for any format without delimiters.

The following includes examples of date formats. You can display a list of valid formats for a specific field using the asterisk, "*", in the date format field.

- MMDDYY
- YYMMDD
- DDMMYY
- YYDDD
- MM*DD*YY
- YY*MM*DD
- DD*MM*YY
- YYDDD
- YY*DDD
- YYMM
- YY*MM
- DDMMMCCYY
- MMM*DD*CCYY
- CCYYMMDD
- CCYY*MM*DD
- UMMDDYY
- GYYDDD *(forced Gregorian date format)*

The following formats are provided for MLE COBOL date format.

```
YYXX
YYXXX
YYXXXX
YYYXXX
YYYXXXX
YYYYXXXX
```

Two formats are included in the distributed PSAPFMTS to describe dates used in the software applications distributed by Geac Computer Corporation. This is not used by Optim, but should be retained as is to simplify future upgrades. They are:

Field	Date Format	Example
3-byte decimal	D1900R1	January 1, 1900 is day 1
7-byte decimal	YYYMMDDX	First 3 ½ bytes are YYYMMDD

Customizing Date Formats

The format and field definitions are defined with the following macros:

FORMAT
> The format information. These are organized in FOP9SAPF by data type.

PCODE
> The fields that comprise the date.

The following describes how to modify the member FOP9SAPF to include additional formats. This requires two steps: specifying the FORMAT statements and specifying the PCODE statements.

FORMAT Statements

The syntax of the **FORMAT** specification is as follows:

```
FORMAT minlen,'name',pcodelabel,aux,EXIT=exit,
  UNITS=units,FLAG=flag,SKIP=skip
```

FORMAT Macro name for defining a format. It must be preceded by a space. Typically, it starts in column 10.

minlen Minimum length, in bytes, of fields which can use this format.

'name' Name used to identify this field. It must be delimited with apostrophes. Maximum length is 11 bytes.

pcodelabel

Label of a series of PCODE specifications that detail the fields used by this format.

aux Auxiliary information based on character type.

For character data: The delimiter character used in the format. The specified value can be the character (C'/') or the wild card indicator (DATEDELIM_WC).

For binary data: The number of decimal digits required to represent the date. Frequently, binary numbers must be converted into numeric characters to be parsed correctly. For example, December 31, 1998 formatted as YYYYMMDD would be 19981231 decimal or 0130E3AF in hex. To parse the value, it must be converted into unpacked decimal. In this case, the *aux* value should be set to 8. Once the value is converted to character, the standard PCODE mapping definitions for the equivalent character format date can be used.

If the binary number is saved in 2's complement, the number of characters should be increased by 128. For example, FFE1AE47 is the 2's complement of 1987001, a date in YYYYDDD format. In this case, *aux* should be set to 135 (128 + 7) to properly convert the date.

exit Optional. The 1-8 character name of the site-written exit routine.

The exit is called for GET, PUT and TERM processing.

For more information about the exit, see Chapter 17, "Define User Exits for Date Aging," on page 175. (Two sample exits written in COBOL, FOPHXIT1 and FOPHXIT2, are included in the install library.)

units An override to the units value used on the PCODE entries. Use this when it is necessary to override the processing by changing units required for a format. If more than one unit is needed, code them with a plus sign, "+", between them (for example, YEAR+JDAY).

flag Optional. Indicates special processing.

NOCLEAR

Specifies whether only part of a field containing a date is aged. The unused portion retains the original value. For example, with this operand you can direct Optim to process only the YYMMDDD value of a packed format such as:

YYMMDDHHMMSSF

NOLIST Do not include this format in the selection list.

GREGORIAN

When aging dates in Julian or Gregorian relative day number format by years or days, the results are determined by adding the aging amount. For example, when you age the Julian date 1999/200 (July 19, 1999) by one year, the result is 2000/200 (July 18, 2000). The day of the month is different because 2000 is a leap year. To maintain the same day of the month, age by 12 months or specify the Gregorian flag on the FORMAT being used. Examples are provided on the GYYDDD and GD1900R1 formats.

skip Optional. Specifies the SKIP statement(s) that defines values to be skipped for this format. If omitted and the format includes a base day or month number on the corresponding PCODE specifications, no skip processing is performed. If omitted and a base was not provided, the global list of skipped dates is checked.

Specify SKIP=NONE to bypass skipped date checking for this format. For more information and examples, see "Customizing Skip Values" on page 169.

Modifying FORMAT

The FORMAT statements are provided in the FOP9SAPF member and grouped by data type.

Within the member, comments indicate logical locations for inserting additional FORMAT statements to handle site-specific requirements. When you define a new format, you must define it for each corresponding data type.

Each FORMAT statement must have a unique name, however more than one can point to a specific PCODE.

Examples

The following examples demonstrate the FORMAT statement.

1. Adding an entry for a 3 byte decimal (COMP-3) field, which represents the number of days since 1900. (January 1 1900 is day 0.)

   Find the following comments in the FOP9SAPF member:

   ```
   *---------------------------------------------------------
   *   INSERT SITE SPECIFIC DECIMAL   FORMAT ENTRIES BELOW HERE
   *---------------------------------------------------------
   ```

   Insert the following entry after these comments:

   ```
       FORMAT 3,'D1900',D_D1900
   ```

   This will define a format named D1900 that can be used on decimal fields that are at least 3 bytes in length. The D_D1900 label refers to the PCODE entries that will be defined in the next section.

2. Adding an entry for a 2 byte (16 bit) binary (COMP) field, which represents the year, month, and day in the following format:

   ```
   YEAR    - 7 bits
   MONTH   - 4 bits
   DAY - 5 bits
   ```

   Find the following comments in the FOP9SAPF member:

   ```
   *---------------------------------------------------------
   *   INSERT SITE SPECIFIC BINARY    FORMAT ENTRIES BELOW HERE
   *---------------------------------------------------------
   ```

   Insert the following entry after these comments:

   ```
       FORMAT 2,'BINPACK',B_BINPACK
   ```

   This will define a format named BINPACK that can be used on binary fields that are at least 2 bytes in length. The B_BINPACK label refers to the PCODE entries that will be defined in the next section.

3. Adding an entry for a 4-byte binary number consisting of the year, month, and day (YYYYMMDD) in 9's complement format. For example, December 31, 1998, would be represented in memory by:

   ```
     99999999
    -19981231
    ---------
     80018768   =      4C4FD50   (hex)
   ```

   After the preceding example, insert the following entry:

   ```
       FORMAT 4,'YYYYMMDD9',B_YYYYMMDD9,8
   ```

   This will define a format to map fullword binary numbers. Because this format must be converted to its decimal equivalent before it can be broken into the individual date components, the number of decimal digits, 8, is specified at the end of the macro.

PCODE Statements

The syntax of the PCODE entries is as follows:

label PCODE *length,unit,offset,flag,base,type*

label   Identifier on the first PCODE entry for a format. This should be specified starting in column 1.

PCODE Macro name for defining a pseudo-code entry. It must be preceded by a space. Typically, it starts in column 10.

length Required. Length of the field represented in this entry specified in terms of the underlying data type. Character fields are in bytes, decimal fields are in half-bytes, binary fields are in bits. Free-form are indicated by 0.

unit Required. Unit of measure specified in the field. This must be CENTURY, YEAR, MONTH, MONTHSTR, DAY, or JDAY.

offset Optional. Offset for the start of the value. This is the number of digits to skip from the end of the previous component to the start of this one.

flag Special processing flag defined as one of the following:

WINDOW Force the use of Pivot Year window.

SIGN_FLAG
Number is signed (valid for decimal and binary data). The sign is treated as a flag, which is copied from the input to the output field.

SIGN_VALUE
Number is signed. The sign is recognized as the sign for the number. This is useful if the number represents a number of days since some base date and there are dates prior to the base date.

COMP2 Number is in 2's complement.

COMP9 Number is in 9's complement.

COMP10 Number is in 10's complement.

base If you need to specify multiple values, use the plus sign "+", between them (for example, WINDOW+SIGN).

Base year or Lilian day number. This value is added to the date prior to aging and subtracted from the aged date prior to writing the output.

The Lilian day for a specific date can be determined using the Optim Date Calculator.

type Override data type for BIN, CHAR, or DEC. This is useful when a component of the format is not the same type as the overall FORMAT.

Julian Dates
Note that a date format can have both a Julian day specification and a month and day specification.

On input, the Julian day takes precedence. On output, both are generated from the aged date. Therefore, a format such as YYYYMMDDJJJ is processed properly.

Explicit vs Relative
A distinction is made between processing an explicit date and a date relative to some base date.

When processing an explicit date, defaults are provided for the unspecified units to determine the date to be aged.

The defaults are:

Year: Current year

Month: *January*

Day: *1*

The value for year can be overridden with an absolute value in FOP9SAPF.

Modifying PCODE

The PCODE statements are provided in the FOP9SAPF member. Within the member, comments indicate logical locations for inserting additional PCODE statements to handle site-specific requirements.

Examples

The following examples demonstrate the PCODE statement.

Adding the field specifications for the D1900, BINPACK and YYYYMMDD9 formats.

Find the following comments in the FOP9SAPF member:

```
*------------------------------------------------------
*  INSERT SITE SPECIFIC PCODE ENTRIES BELOW HERE
*------------------------------------------------------
```

Insert the following entries after these comments:

```
D_D1900                 PCODE
5,DAY,,SIGN,115861
*
B_BINPACK                    PCODE 7,YEAR
                  PCODE 4,MONTH
                  PCODE 5,DAY
*
B_YYYYMMDD9                    PCODE  4,YEAR,,COMP
                               PCODE  2,MONTH,,COMP9
                  PCODE  2,DAY,,COMP9
```

1. The first PCODE will create an entry for a 5 decimal digit field that has a decimal sign after the number. The base date represented by this format is Lilian date 115861, which is January 1, 1900.

2. The next PCODE specifications define a date that is made up of three binary fields represented by 7, 4, and 5 bits respectively.

3. The last set of PCODE specifications map the three individual date components after the date has been converted to unpacked decimal digits. The COMP9 flag identifies that the components are in 9's complement.

Customizing Skip Values

The skip definitions are identified by the macros SKIP and SKIPEND.

SKIP   The skip value information. These are organized in FOP9SAPF directly after the OVERRIDE statements at the beginning of the definitions.

SKIPEND
>   Defines the end of a block of skipped dates.

The following describes how to modify the SKIP definitions and include specific definitions for individual formats in the member FOP9SAPF. This requires three steps:

1. Specify the SKIP statements.
2. Modify the OVERRIDE statement to include the desired global SKIP statement values.
3. Add the SKIP parameter to individual FORMAT statements as required to override the global specifications.

SKIP Specification

The syntax of the SKIP specification is as follows:

label SKIP YEAR=*year*,MONTH=*month*,DAY=*day*

label   Label for the SKIP statement. Use this label on the first statement in a set. This name is used to identify the group on an OVERRIDE or FORMAT statement.

SKIP Macro name for defining a skip value. It must be preceded by a space.

year Value in the year portion that identifies a field to skip.

month Value in the month portion that identifies a field to skip.

day Value in the day portion that identifies a field to skip.

Notes:

- You can specify any combination of YEAR, MONTH, and DAY. Each unit can be specified only once.
- At least one parameter (YEAR, MONTH, or DAY) must be specified.
- When a parameter is not specified for a unit, that statement does not apply to any field containing the unspecified unit. For example, if YEAR is omitted, the statement does not apply to any field containing a year value.
- Use an asterisk as the value for a unit to indicate any value is a valid skip value. For example, MONTH=* specifies that any month value is to be skipped. You can combine this with explicit values for YEAR and/or DAY for the SKIP statement.
- A SKIPEND entry is required to mark the end of a group of SKIP statements identified by a label.

Examples

The following SKIP statements are provided in the FOP9SAPF member.

These statements are grouped by the units that are included. Thus, a single SKIP statement can be defined for several FORMAT statements regardless of data type. (Examples for specifying the SKIP parameter on the FORMAT statements follow.)

```
*-----DEFINE THE SKIP DATES WITH: YEAR, MONTH, AND DAY
GBLSKIP SKIP  YEAR=1999,MONTH=99,DAY=99
        SKIP  YEAR=9999,MONTH=12,DAY=31
        SKIP  YEAR=1,MONTH=1,DAY=1
        SKIP  YEAR=0,MONTH=0,DAY=0
        SKIP  YEAR=1900,MONTH=0,DAY=0
*-----DEFINE THE SKIP DATES WITH YEAR AND DAY
        SKIP  YEAR=1999,DAY=999
        SKIP  YEAR=9999,DAY=999
        SKIP  YEAR=0,DAY=0
        SKIP  YEAR=1900,DAY=0
        SKIP  YEAR=2000,DAY=0
*-----DEFINE THE SKIP DATES WITH YEAR AND MONTH
        SKIP  YEAR=1999,MONTH=99
        SKIP  YEAR=9999,MONTH=99
        SKIP  YEAR=0,MONTH=0
        SKIP  YEAR=1900,MONTH=0
        SKIP  YEAR=2000,MONTH=0
*-----DEFINE THE SKIP DATES WITH MONTH AND DAY
        SKIP  MONTH=99,DAY=99
        SKIP  MONTH=0,DAY=0
*-----DEFINE THE SKIP DATES WITH YEAR ONLY
        SKIP  YEAR=0
        SKIP  YEAR=9999
*-----DEFINE THE SKIP DATES WITH MONTH ONLY
        SKIP  MONTH=0
        SKIP  MONTH=99
*-----DEFINE THE SKIP DATES WITH DAY ONLY
        SKIP  DAY=0
        SKIP  DAY=99
        SKIP  DAY=999
        SKIPEND
```

When coding a SKIP macro for a format with a base day or month, the macro must contain the Lilian day of the base date and the number to be skipped. For example, the following SKIP macro directs Optim to skip fields containing binary zeros or all nines formatted as D1900R1 (the number of days since January 1, 1900).

```
GLBSKIP  SKIP   DAY=115860      Value of zeros
         SKIP   DAY=215859      Value of all 9s
         SKIPEND
```

Global Skipped Dates

The list of global skipped values is defined using the OVERRIDE macro as described in the following section. The distributed version uses the values shown on the previous page.

Assigning on FORMAT

You use the label on the SKIP statement to assign them to the FORMAT statements.

The following SKIP statements demonstrate specifying two groups of labeled SKIP statements and using the asterisk.

```
*-----DEFINE THE SKIP DATES WITH: YEAR, MONTH, AND DAY
FULLFMT     SKIP  YEAR=*,MONTH=99,DAY=99
     SKIP  YEAR=*,MONTH=0,DAY=0
*-----DEFINE THE SKIP DATES WITH YEAR AND DAY
YRDAY       SKIP  YEAR=*,DAY=999
     SKIP  YEAR=9999,DAY=*
```

Using FORMAT statements provided in FOP9SAPF for the examples, you can assign these SKIP statements as shown:

For Character Data:

```
FORMAT 6,'MMDDYY',C_MMDDYY,SKIP=FULLFMT
FORMAT 8,'YYYYDDMM',C_YYYYDDMM,SKIP=FULLFMT
FORMAT 8,'YYYY/DDD',C'/',SKIP=YRDAY
```

For Decimal Data:

```
FORMAT 3,'YYDDD',D_YYDDD,SKIP=YRDAY
```

Note that you must assign SKIP to the format in each data type for which it is to be applied.

You can use the OVERRIDE statement to apply a set of SKIP statements globally and then specify other SKIP statements on individual FORMAT statements. The FORMAT specifications override any OVERRIDE specifications.

To override global specifications for an individual FORMAT, use SKIP=NONE. For example,

```
FORMAT 6,'MMDDYY',C_MMDDYY,SKIP=NONE
```

This is useful when fields defined for a specific format must contain a valid date and cannot be skipped.

Customizing OVERRIDE Macros

The distributed FOP9SAPF member contains several OVERRIDE macros.

Many of these are commented since they include default values. The syntax for these macros is as follows:

```
OVERRIDE MONTHS=months,DELIM=delim,SIGN=sign,YEAR=year,SKIP=skip
```

months The sets of string values to be used as the names of the months. By default, these values are set to the first three characters of the English language names of the months.

Each set of string values must be delimited by apostrophes and a comma used between strings. All sets of strings should be grouped together and bound by parentheses.

delim The default delimiter used for the output when a wildcard was used in the input. The character must be delimited by apostrophes.

The default delimiter defined in FOP9SAPF is:
    OVERRIDE DELIM='/'

sign The default decimal sign digit on output.

The default sign defined in FOP9SAPF is:
    OVERRIDE SIGN='F'

year A default year when the input date is supplied in months and/or days without a year value.

The default year defined in FOP9SAPF is:
    OVERRIDE YEAR=0

A year of 0 indicates the current year.

skip Indicates which skip values are to be applied globally when aging data.

The default defined in FOP9SAPF is:

OVERRIDE SKIP=GBLSKIP

The OVERRIDE specifications are provided in the beginning of FOP9SAPF. Edit them as desired.

Assemble and Link the Table

After you have modified the table, you must assemble and link the table to prepare it for use.

Modify the sample JCL provided in JCLASM in the install library to perform these steps. The following is a copy of the sample JCL.

```
       //* JOB  )) PUT A VALID JOB CARD HERE                        <===
       //*----------------------------------------------------------------*
       //*   ASSEMBLE IBM CORPORATION   'FOP2RUSA' SEMANTIC RULE TABLE OR  *
       //*                              'FOP9SAPF' DATE FORMAT TABLE       *
       //*----------------------------------------------------------------*
       //JCLASM  PROC MEMBER=FOP9SAPF,                               <===
1//         QUAL1=qualifier1                <===
2//         QUAL2=qualifier2                <===
       //ASM      EXEC PGM=ASMA90,PARM='NODECK,OBJECT'
       //SYSLIB   DD  DSN=&QUAL1..&QUAL2..SFOPINST,DISP=SHR
       //SYSLIN   DD  DSN=&&OBJ,UNIT=VIO,SPACE=(TRK,(10,10)),DISP=(,PASS)
       //SYSIN    DD  DSN=&QUAL1..&QUAL2..SFOPINST(&MEMBER),DISP=SHR
       //SYSUT1   DD  UNIT=VIO,SPACE=(TRK,(10,10))
       //SYSUT2   DD  UNIT=VIO,SPACE=(TRK,(10,10))
       //SYSPRINT DD  SYSOUT=*                                      <===
3//SYSTERM  DD  SYSOUT=*
       //*----------------------------------------------------------------*
       //*           LINK THE TABLE, IF IT ASSEMBLED CLEANLY              *
       //*----------------------------------------------------------------*
       //LKED     EXEC PGM=HEWL,COND=(8,LT,ASM),PARM='AMODE=31,RMODE=ANY'
       //SYSLIN   DD  DSN=&&OBJ,DISP=(OLD,DELETE)
       //SYSLMOD  DD  DSN=&QUAL1..&QUAL2..SFOPLLIB(&MEMBER),DISP=SHR
       //SYSUT1   DD  UNIT=SYSDA,SPACE=(TRK,(10,10))
       //SYSUT2   DD  UNIT=SYSDA,SPACE=(TRK,(10,10))
       //SYSPRINT DD  SYSOUT=*
       //*----------------------------------------------------------------*
       //        PEND
       //*----------------------------------------------------------------*
       //JCLASM  EXEC JCLASM
```

To use this JCL, supply a valid job card and then modify the JCL as follows:
1. Make sure MEMBER points to FOP9SAPF.
2. Specify the high-level qualifiers for the Optim libraries.
3. Specify the name of the source library containing FOP9SAPF.

Chapter 17. Define User Exits for Date Aging

Optim provides extensive facilities for aging date columns. To do so, the solutions support hundreds of date formats and allow you to augment the distributed date formats.

(See "Customizing Date Formats" on page 165.) Further, Optim provides a set of semantic rules that you can customize to ensure that aged dates adhere to the business rules embedded in your applications. The semantic rules as well as the date format can be extended. Optim provides user exits to handle any situation that cannot be handled through the product extensions.

This section discusses an exit to handle dates expressed in non-standard date formats when aging.

Exits can be written in assembler language, COBOL, or PL/I.

Overview

Two sample COBOL date format exit programs, FOPHXIT1 and FOPHXIT2, are included in the Optim install library.

These exits are useful for handling:
- Non-standard formatting.
- Values to be skipped other than those automatically handled.
- Dates for which the aging is dependent on another date.

To use a date format exit, specify the name of the exit on the EXIT parameter of the FORMAT statement in the member FOP9SAPF in the install library.

A date format exit consists of a single load module which is dynamically loaded and called to perform Get, Put and Term functions. A single entry point is used. The exit is called with an entry code indicating the desired function along with function-specific parameters. These parameters are passed to the exit using a standard z/OS-format parameter list.

Date Aging Functions and Parameters

The following are the date aging functions and their parameters.

Function: GET
> Parameters:
>> Entry Code
>>
>> Anchor
>>
>> Message
>>
>> In-Parms
>>
>> Out-Parms
>>
>> Aging Amount
>>
>> Pivot Year
>>
>> Date-Value

Function: PUT
> Parameters:
>> Entry Code

Anchor

Message

In-Parms

Out-Parms

Aging Amount

Pivot Year

Date-Value

Function: TERM

Parameters:

Entry Code

Anchor

Message

Parameters

Use the following data aging parameters. Each parameter is followed by its description, number of bytes and type:

Parameter: Entry Code

Description: Function indicator specified as GET, PUT or TERM:

4 GET - Indicates that a date value is to be retrieved from the input.

8 PUT - Indicates that a date value is to be formatted and written to the output by the exit.

12 TERM - Optim has completed processing. A termination call is made to clean up dynamically allocated storage. This call is made only when the anchor -INSTANCE is non-zero.

Number of Bytes: 4

Type: Binary

Parameter: Anchor-XXX

Description: The following fields are available to the exit to anchor dynamic storage or to communicate from one exit call to another. They are initialized to 0 when first used.

GLOBAL - Available to all format exits.

FIELD - Shared by GET and PUT.

INSTANCE - Single field for GET or PUT.

Number of Bytes: 4

Type: Binary

Parameter: Message

Description: A message to the user when the exit returns with RC=24. Initially set to blanks.

Number of Bytes: 80

Type: Character

Parameter: In-Parms

Description: Input field parameters including:

Address of the input field. **Number of Bytes:** 4 **Type:** Pointer

Name of the input field format (from the FORMAT statement). **Number of Bytes:**12 **Type:** Character

Data type of the column: **B** - Binary, **C** - Character, **D** - Decimal**Number of Bytes:**1 **Type:** Character

Length of the input field. **Number of Bytes:** 2 **Type:** Binary

Total Number of Bytes: 19 **Type:** Group

Parameter: Out-Parms

Description: Output field parameters. Passed on both GET and PUT calls. (Same as In-Parms)

Note that before processing the record, the input record is copied to the output area.

Number of Bytes: 19

Type: Group

Parameter: Aging Amount

Description: Amount by which the date is to be aged.

Year **Number of Bytes:** 2 **Type:** Binary

Month **Number of Bytes:** 2 **Type:** Binary

Day **Number of Bytes:** 4 **Type:** Binary

Total Number of Bytes: 8 **Type:** Group

Parameter: Pivot Year

Description: Year used to determine whether a two-digit year value is handled as occurring in the 20th century (1900) or the 21st century (2000).

Number of Bytes: 2

Type: Binary

Parameter: Date-Value

Description: On GET, the exit passes the parsed date to Optim (RC=4 only). On PUT, the products pass the aged date to the exit.

Year, as yyyy **Number of Bytes:** 2 **Type:** Binary

Month **Number of Bytes:** 2 **Type:** Binary

Day **Number of Bytes:** 4 **Type:** Binary

Total Number of Bytes: 8 **Type:** Group

Anchor

The parameter list for each entry code includes anchor pointers where you can save such information as status, control blocks, and so on. Use the anchor to handle communication between two exits. This is useful, for example, when a date is dependent on another (such as calculating "three days later"). The exits can perform the appropriate processing.

Return Codes List

Each time the exit returns to Optim, it should set a return code.

The following is the list of valid return codes.

Table 3. Return codes list

Return Code	Meaning
0	Process. The date required no special processing. Optim processes the date.
4	Parsed. The exit has parsed the input date into the DATE-VALUE parameter. Optim ages the date. This return code applies to the GET function only.
8	Complete. The exit has performed all processing and written the date to the output column. Optim continues with the next column. (See notes.)
12	Skip. Optim handles the date as "skipped." (See notes.)

Table 3. Return codes list (continued)

Return Code	Meaning
16	Invalid. Optim handles the date as "invalid." (See notes.)
24	Fatal. The exit has encountered a fatal error. Optim terminates with an error message in the message parameter area.

Notes:

- Return codes 8, 12, and 16 might be used for GET or PUT functions. When used on a GET function, the corresponding PUT function call for that column is omitted.
- For RC=8, the user must place the formatted date in the output column (pointed to by the first subfield of the Out-Parms parameter). The data type and length of the formatted value must be consistent with the third and fourth subfields of the Out-Parms parameter.
- For RC=12 and RC=16, the user may optionally place a formatted date in the output column.

Chapter 18. Writing and using I/O exits for processing VSAM or sequential data

The Optim solutions support the use of I/O exits to compress, decompress, encrypt, or decrypt VSAM or sequential data as it is processed.

The I/O exit program can be called as part of all I/O activity to a data file (Input, Output, or Discard). Each exit consists of a single load module that is loaded and invoked after the solution has dynamically allocated the file.

Using an Exit

The I/O exit will be invoked if the following conditions are met.
1. The Site Option, **Enable VSAM/Seq I/O Exit** is set to Y or U.
2. A load module (the I/O exit) exists and is available for the solution to load it.

If **Enable VSAM/Seq I/O Exit** is set to Y, the exit identified in the relevant Site Option **Enable VSAM/Seq I/O Exit** is called whenever a record is read or written. An exit referenced by the Legacy Table is ignored.

If **Enable VSAM/Seq I/O Exit** is set to U, an exit referenced in the Legacy Table is called. If the Legacy table does not specify an exit, the exit identified with the Site Option **Enable VSAM/Seq I/O Exit** is called.

FOP2DIOX

A sample exit program, FOP2DIOX, written in assembly language is included in the install library and is specified by default with the Site Option **Enable VSAM/Seq I/O Exit**. The sample illustrates the required functions, parameters, and return codes. It processes records in a sequential file with fixed length records.

Exit Conventions

The exit must be in RMODE ANY and AMODE 31. The exit is entered with a standard MVS™ parameter list. The first parameter is an entry or function code, the second is an anchor word and the remaining parameters vary depending on the function code. The anchor word is intended for use by the exit to address persistent storage acquired, used, and freed by the exit. The anchor word is not modified.

Functions

The functions and their parameters are:

Function	Parameters	Comments
OPEN	Entry code Anchor Message DDname MaxRecLength Mode Member Name	The first call to the exit is an Open request and subsequent calls are made for Get, Put, and Close requests as required for the particular file. The return code from the Open determines the level of processing that the exit will perform. The exit can perform the physical I/O to the file or the solution can perform the physical I/O after the exit processes the record. See the description of return codes that follows.

Function	Parameters	Comments
GET	Entry code Anchor Message Buffer Length	
PUT	Entry code Anchor Message Buffer Length	
CLOSE	Entry code Anchor Message	

Parameters

The following is a description of the parameters:

Table 4. Parameters

Parameters	Description	Number of Bytes	Type
Anchor	An anchor word available to the exit to save storage across multiple calls to the exit for the same file. This is initialized to 0 on the OPEN call.	4	Binary
Buffer	Pointer to the area into which the exit moves the next record prepared for solution processing or the location where the exit is passed the processed record. When the exit returns an 8 from the Open call, this area is used to pass the record read by the solution to the exit and return the record from the exit to the solution to be written. The number of bytes allocated for the buffer is the value specified for the MaxRecLength parameter on the OPEN call.	4	Binary

Table 4. Parameters (continued)

Parameters	Description	Number of Bytes	Type
DDname	An eight-byte DDNAME used by the solution to allocate the file. Use one of the following DDnames: PSAFIN Input dataset or output if update in place PSAFOUT Output dataset PSAFDISC Discard dataset COBOL or PL/I exits should use this parameter to ensure that they use the correct value in the SELECT statement.	8	Character
Entry Code	The function indicator specified as one of the following: **O O**PEN **G G**ET **P P**UT **C C**LOSE	1	Character
Length	The length of the input record in the buffer. This is filled in by the exit on GET and set by the solution on PUT.	4	Binary
MaxRecLength	The maximum length of the record from a file that can be returned from a GET call. The solution allocates an area based on this value.	4	Binary
Member Name	Name of a member in a partitioned data set to read from or write to.	8	Character
Message	An 80-byte area for messages used to return a description of errors.	80	Character

Table 4. Parameters (continued)

Parameters	Description	Number of Bytes	Type
Mode	The access mode as one of the following: I Input. Only GET calls will be made. O Output. Only PUT calls will be made. U Update. Both GET and PUT calls will be made. (This is passed on the OPEN call when the input file is to be aged in place.)	1	Character

Return Codes

Each time the exit returns control, it must set a return code. For return codes other than zero, the exit can provide a message in the Message area describing the error. The following is the list of valid return codes for each function.

Function	Return Code	Meaning	Comments
OPEN	0	File successfully opened.	If the I/O exit performs all file operations including issuing the physical Open, Get, Put, and Close requests, a useful condition if non-standard access methods are needed, the exit calls return 0 if processing was successful and 20 if there was a fatal error.
	8	The solution to open, read, write, and close.	Return code 8 indicates that the exit will perform logical record processing only. The solution will issue the physical I/O requests, calling the exit after records are retrieved and before records are written. Limiting the exit to record processing reduces the complexity of the exit and is useful in cases such as handling compressed records. For example, on a Get request, the exit simply calls a decompress routine to transform the physical record into the logical record used in the Record Layout.
	12	Open failed. Processing terminated.	
	20	Exit failed. The solution must terminate.	
GET	0	Record is successfully copied to buffer and length is set.	
	4	End of file.	
	8	Valid only if the OPEN return code is 8. Process the record returned from the exit, call the exit routine again with another Get request, without first reading another record from the Input file.	This return code is useful when the exit routine produces multiple records from a single Input file record.

Function	Return Code	Meaning	Comments
	9	Valid only if the OPEN return code is 8. Skip processing of the current record, read the next Input file record, and call the exit routine again with another Get request.	This return code is useful when the exit routine determines that the current Input record should be ignored.
	20	Exit failed. The solution must terminate.	
PUT	0	Record successfully written or updated.	
	8	Valid only if the OPEN return code is 8. Process another Input file record without writing the current record, and call the exit routine again with another Put request.	This return code is useful when the exit routine must produce a single Output file record from multiple records.
	9	Valid only if the OPEN return code is 8. Write the record returned from the exit to the Output file and call the exit routine again with another Put request, without first processing another Input file record.	This return code is useful when the exit routine must produce multiple Output file records from a single record.
	12	File is full. Processing cannot proceed.	
	20	Exit failed. The solution must terminate.	
CLOSE	0	File successfully closed.	
	12	File is full. Processing cannot proceed.	
	20	Exit failed. The solution must terminate.	

Chapter 19. Writing and using I/O exits for processing IMS data

The Optim solutions support the use of I/O exits to compress, decompress, encrypt, or decrypt IMS segment data as it is processed.

Optim automatically processes I/O exits that are specified in the DBD for an IMS segment but cannot invoke compression or encryption programs that are not specified in the DBD unless you take certain steps.

For IMS data, an I/O exit can be called during extract processing after a segment is fetched. When exit processing is complete, the extract process writes the segment data to the extract file. During insert processing, an I/O exit can be called after a row is read from the extract file and inserted into the IMS database when control is returned to the insert process.

Using an Exit

The I/O exit will be invoked if the following conditions are met.

1. The Site Option, **Enable IMS I/O Exit** is set to Y or U.
2. A load module (the I/O exit) exists and is available for the solution to load it.

If **Enable IMS I/O Exit** is set to Y, the exit identified with the Site Option **Enable IMS I/O Exit** is called whenever a record is read or written. An exit referenced by the Legacy Table is ignored.

If **Enable IMS I/O Exit** is set to U, the exit referenced in the Legacy Table is called. If the Legacy Table does not specify an exit, the exit identified with the Site Option **Enable IMS I/O Exit** is called.

FOPIMIOX and FOPIMIOC

Sample exit programs, FOPIMIOX, written in assembly language, and FOPIMIOC, written in COBOL, are included in the install library. FOPIMIOX is specified by default with the Site Option **Enable IMS I/O Exit**. The samples illustrate the required functions, parameters, and return codes. Each processes fixed length IMS segments.

Exit Conventions

The exit must be in RMODE ANY and AMODE 31. The exit is entered with a standard MVS parameter list. The first parameter is an entry or function code and the use of the remaining parameters depends on the function code.

Functions

The functions and their parameters are:

Function	Parameters	Comments
OPEN	FUNCCODE PROCMODE DBTYPE SEGNUM DBDNAME SEGNAME MAXSEGLN DBSEGMIN	The first call to the exit is an Open request and subsequent calls are made for Get, Put, and Close requests as required for the particular segment type. The return code from the Open determines the level of processing that the exit will perform. The exit must set MAXSEGLN to the size of the largest segment it will return. The solution allocates storage for the I/O Buffer on the basis of the value returned in MAXSEGLN.
GET	FUNCCODE PROCMODE DBTYPE SEGNUM DBDNAME SEGNAME MAXSEGLN DBSEGMIN IOBUFLEN IOBUFFER	The exit returns the decompressed or decrypted data in the buffer and updates the buffer length, which cannot exceed the maximum segment length (MAXSEGLEN) returned in the Open call. For variable length segments, the exit must update and return the segment length in the first two bytes of the buffer, as well as in buffer length. If the values do not match when returned, the solution issues an error message and terminates.
PUT	FUNCCODE PROCMODE DBTYPE SEGNUM DBDNAME SEGNAME MAXSEGLN DBSEGMIN IOBUFLEN IOBUFFER	The exit receives data in the buffer, which it can compress or encrypt in the buffer, and set the buffer length, which cannot exceed the maximum segment length returned in the Open call. For variable length segments, the exit must provide the length of the segment in the first two bytes of the buffer, as well as in buffer length. If the values do not match when returned, the solution issues an error message and terminates.
CLOSE	FUNCCODE PROCMODE DBTYPE SEGNUM DBDNAME SEGNAME MAXSEGLN DBSEGMIN	

Exit Parameters

The following is a description of the exit parameter list and parameters:

Parameter	Description	Number of Bytes	Type
FUNCCODE	Function code: **O** – **O**pen **C** – **C**lose **G** – **G**et **P** – **P**ut The solution sets this value before calling the exit.	1	Character
PROCMODE	Processing mode. **1**- BMP **2**- DLI Set in the Open call and not changed.	1	Character
DBTYPE	Supported database types are represented as decimal values as follows: **3** – HSAM **4** – SHSAM **5** or **13** – HDAM **6** or **14** – HIDAM **9** – HISAM **11** – SHISAM **34** – HALDB **37** – DEDB Set in the Open call and not changed.	1	Unsigned binary
SEGNUM	Segment number in the DBD.	1	Character
DBDNAME	DBD name.	8	Character
SEGNAME	Segment name.	8	Character
DBSEGMAX	Maximum segment length in DBD (for both fixed and variable length segments).	4	Binary
DBSEGMIN	Minimum segment length in DBD for variable length segments. This value is set to zero for fixed length segments.	4	Binary
MAXSEGLN	Maximum segment length returned by the exit. The solution sets this parameter to -1 before the Open call. The exit must set this value to the desired maximum segment length before returning control from OPEN.	4	Binary

Parameter	Description	Number of Bytes	Type
IOBUFLEN	The length of the buffer. This exit sets this value on GET calls and the solution sets it on PUT calls.	4	Binary
IOBUFFER	Address of the buffer to and from which the exit moves data during GET and PUT calls. The solution sets this value before calling the exit. The solution determines the size of the buffer from the MAXSEGLN returned from the Open call and allocates the required number of bytes.	4	Address
MESSAGE	Optional Error, Warning, or Status message set by the exit.	80	Character
RESERVED	Data area reserved for future use.	100	Character
USERDATA	Scratch pad area available to exit.	2048	Undefined

Return Codes

Each time the exit returns to the solution, it must set a return code. The exit can also provide an optional message in the Message area. The solution will display the message in the TRACE or as part of the System Error depending upon the return code. The following is the list of valid return codes for each function.

Function	Return Code	Parameters
OPEN	0	The Open call succeeded
	4	The segment type is not processed by the exit
	>4	The exit failed. The solution must terminate.
GET	0	The segment processed successfully and the buffer length is set
	4	The segment has been skipped by the exit.
	>4	The exit failed. The solution must terminate.
PUT	0	The segment processed successfully and the buffer length is set
	>4	The exit failed. The solution must terminate.
CLOSE	0	The Close call succeeded
	>4	The exit failed. The solution must terminate.

Chapter 20. Install the Optim Data privacy providers

The Optim Data Privacy Solution for z/OS (FMID HAI8B30) provides components to mask sensitive data propagated outside your production environment, such as national IDs, credit card numbers, and email addresses. You can mask sensitive data by replacing it with fictional, yet contextually accurate data.

Optim for z/OS includes the ability to call supported functions of the Optim Data Privacy Solution directly from a Column Map. This section contains instructions for configuring the Optim Data Privacy Solution for z/OS for use with Optim for z/OS.

You must first install and configure the Optim Data Privacy Solution for z/OS according to instructions in the Program Directory and Installation Guide. The installation and configuration instructions for Optim Data Privacy Solution for z/OS can be found at: http://www.ibm.com/support/knowledgecenter/ SSMLQ4_11.3.0/com.ibm.nex.mod.doc/topics/opmod-c-overview_Optim_data_privacy_components.html

Ensure that the TSO region size is a minimum of 220 MB.

Applying the Data Privacy license to Optim for z/OS

Before you can use the data privacy providers, you must first provide a valid Data Privacy license to Optim for z/OS.

Note: If you need an Optim for z/OS Data Privacy license, contact your IBM representative for information.

1. Select option 8 (Enable Optim Product(s) for Execution).
2. From the Product Licensing/Configuration panel, in the field for Optim Data Privacy, in the Password column, enter the provided Optim data privacy license key and press Enter. The Optim site options module (FOPMDFLT) will be updated and the status on the panel will change according to the type of license key entered (permanent or temporary), such as:

```
--------------------- Product Licensing/Configuration ------------------------
Command ===>                                              Scroll ===> PAGE
                                                       SUBSYS: DD9F
Site CPUID        : 0011FE
Load Library DSN   : OPTUSR.FOPnnn.SFOPLLIB


     Product / Release          Current Product Status     Password
  ----------------------     ----------------------------  ---------   1 of 9
** ***************************** TOP *******************************
  Optim Access             Permanently Licensed
  Optim Move w/ Delete/DP   Permanently Licensed           _____
  Optim Compare            Permanently Licensed
  Optim Archive            Permanently Licensed
  Optim Move for VSAM/Seq   Permanently Licensed
  Optim Move for IMS        Permanently Licensed
  Optim Compare for VSAM/Seq Permanently Licensed
  Optim Compare for IMS     Permanently Licensed
  Optim Data Privacy        Permanently Licensed           _____
** *************************** BOTTOM ******************************


INSTRUCTIONS: Press ENTER to Continue, Enter END to Return or CANCEL to Cancel
              Enter Password to Enable, Extend Evaluation or License Product
              Enter DISABLE in Password field to Disable Product


******************************** N O T E S ********************************
*            You must have authority to update the Load Library          *
*            to complete this function successfully.                     *
**************************************************************************
```

Configuring data privacy providers

After the data privacy license key has been applied to Optim for z/OS and the Optim data privacy license files have been generated, you must define required environment variables.

Creating the environment variable file

Optim provides sample job FOPODPPJ in the SFOPSAMP library that creates a copy of the sample environment variable file for your use with the Optim data privacy providers.

```
//*  SUPPLY A VALID JOB CARD
//*--------------------------------------------------------------*
//*                                                              *
//*   COPY SAMPLE ENVIRONMENT VARIABLE FILE TO PDS               *
//*                                                              *
//*   To execute this JCL:                                       *
//*   - replace qual1.qual2 to match your Optim environment      *
//*   - change OPTIM.ODPP.ENV and member V1130 to your desired   *
//*     destination of the environment file.                     *
//*--------------------------------------------------------------*
//*
//* CREATE PDS
//*    LRECL OF 256 BECAUSE NO LINE CONTINUATION ALLOWED
//*
//CRTENV   EXEC PGM=IEFBR14
//PDSENV   DD DSN=OPTIM.ODPP.ENV,DISP=(NEW,CATLG,DELETE),
//            SPACE=(TRK,(5,2,2)),UNIT=DISK,
//            LRECL=256,RECFM=V,BLKSIZE=260
//*
//* COPY ENVIRONMENT VARIABLE SAMPLE
//* EDIT PDS MEMBER FOR LOCAL PATHS
//*
//STEPJ    EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSIN    DD DUMMY
//SYSUT1   DD DSN=qual1.qual2.SFOPSAMP(FOPODPPE),DISP=SHR
//SYSUT2   DD DSN=OPTIM.ODPP.ENV(V1130),DISP=SHR
//*
```

This sample job:
- Creates a new PDS (named OPTIM.ODPP.ENV in this example).
- Copies the sample environment file into a new member (named V1130 in this example).

Specify Environment Variables for Data Privacy

Optim provides sample job FOPODPPE in the SFOPSAMP library that defines the required and optional environment variables for use with the Optim data privacy providers. This sample was used to populate your environment variables file in the previous step.

```
_CEE_ENVFILE_COMMENT=*
* This Environment Variable file is used by Optim/z
* Use Options,Site (0.0) Change Optim Data Privacy to enter the name of this
*    Environment File
* If Site Data Privacy Option is set to U, then enter the name of this
*    Environment File under Options,User (0.1)
* - Use one line per variable
* - No spaces
* - text is case sensitive
* - you cannot use existing environment variables in this file.
* - e.g., you cannot have: LIBPATH=$LIBPATH:/usr/local/odpp/odppbin
*
* LIBPATH is used to load Optim Data Privacy Library DLLs
LIBPATH=/usr/local/odpp/odppbin:/usr/local/odpp/odppext
*
* ODPPERRL location of Optim error messages
ODPPERRL=/usr/local/odpp/odppbin
*
* ODPPLL location of Optim license file
ODPPLL=/usr/local/odpp/license
*
* ODPPTRCL location to write trace information.  User must have write authority
ODPPTRCL=/usr/local/odpp/tracefiles
*
* ODPPTRC Y or N to turn trace on or off
ODPPTRC=N
```

Note the following z/OS Language Environment restrictions regarding the syntax of this file:

- z/OS Language Environment searches for an equal sign to delimit the environment variable from its value. If an equal sign is not found, the environment variable is skipped and the rest of the text is treated as comments.

- Each record of the file is processed independently from any other record in the file. Data within a record is used exactly as input with no substitution. A file containing:

```
FRED=WILMA
FRED=$FRED:BAMBAM
```

will result in the environment variable FRED being set to $FRED:BAMBAM, rather than to WILMA:BAMBAM as would be the case if the same statements were processed by a UNIX shell.

Environment Variables

Descriptions of the required and optional environment variables for use with the Optim data privacy providers are as follows:

LIBPATH

>This environment variable is used to specify where to find the Optim Provider program library. This variable is required.

>Do the following to set this variable:

>In z/OS, set LIBPATH to the full path where the Optim data privacy provider library exists. Here is an example:

>```
LIBPATH=/usr/local/odpp/odppbin:/usr/local/odpp/odppext
```

**ODPPERRL**

This environment variable is used to specify the path to the error message file. This variable is required.

The error message file named ODPPErrMsgs_EN.xml is shipped with the odppbin directory. If you would like to view the contents of this file, use the FTP command to transfer it to your workstation as a binary file. It is in UTF-8 format.

Do the following to set this variable:

In z/OS, set ODPPERRL to the full path to the error file. Here is an example:

```
ODPPERRL=/usr/local/odpp/odppbin
```

ODPPICUDIR

This optional environment variable is used to specify the path to the ICU (International Components for Unicode) files. To support database-specific code pages, ICU uses database-specific packages, such as `OPMICU*.DAT` and `OPMICU*.ALS`. By default, these files are placed in the odppbin folder during the installation process. In most circumstances, you do not need to set this environment variable unless directed to do so by IBM Optim Support.

Do the following to set this variable:

In z/OS, set ODPPICUDIR to the full path to the ICU file. Here is an example:

```
ODPPICUDIR=/usr/local/odpp/odppbin
```

ODPPLL

This environment variable is used to specify the path to the license files for data privacy. This variable is required.

Do the following to set this variable:

In z/OS, set ODPPLL to the full path to the license files for data privacy. Here is an example:

```
ODPPLL=/usr/local/odpp/license
```

ODPPTRCL

This environment variable is used to specify the path to the directory where the trace files are generated by the Optim data privacy providers. This variable is required.

Do the following to set this variable:

In z/OS, set ODPPTRCL to the full path to the trace files. Here is an example:

```
ODPPTRCL=/usr/local/odpp/tracefiles
```

ODPPTRC

This environment variable enables/disables the trace file generation irrespective of the trace level settings. By default tracing is always enabled with a trace level set to ODPP_TRACE_ERROR. In most circumstances, you do not need to set this environment variable unless directed to do so by IBM Optim Support.

To enable tracing set this value to one of the following:

```
{ Y | y | 1 }
```

To disable tracing set this value to one of the following:

```
{ N | n | 0 }
```

In z/OS, set ODPPTRC to enable or disable tracing by the Optim data privacy providers. Here is an example:

```
ODPPTRC=N
```

Defining the environment variables file to Optim

After you have created and modified the environment variable file, you must define it to Optim. Invoke the Optim ISPF interface (typically through the FOP CLIST). From the Main Menu, select option **0 Options**. Then select option **0 Site**.

Note: If the Site option is not displayed, contact your Optim administrator.

From the Site Options panel, scroll to the bottom and find the option for **Change Optim Data Privacy**.

```
-------------------------- FOPMDFLT Site Options -----------------------------
Command ===>                                              Scroll ===> PAGE
                                                 SUBSYS: DD9F
                                                      72 of 75


Register DASD Extract file ===> U        Y-Yes, N-No, U-User
Replace Extr Dir in Batch  ===> U        Y-Yes, N-No, U-User
Require IMS Data Set Names ===> Y        Y-Yes, N-No
Change Optim Data Privacy  ===> N        Y-Yes, N-No, U-User
******************************** BOTTOM ************************************
```

Setting a site default environment variable file

If you wish to set a site default environment variable file that will apply to all users, enter Y for the option **Change Optim Data Privacy** and press Enter.

The Site Data Privacy Options panel will be displayed:

```
+------------------------Site Data Privacy Options------------------------+
|                                                                         |
|  File Name ===>                                                         |
|                                                                         |
|     Enter the name of the data set that defines the Unix System Services |
|     environment variables required for the Optim data privacy providers.|
+-------------------------------------------------------------------------+
```

Enter the name of the environment variable file as previously created . For example:

```
+------------------------Site Data Privacy Options------------------------+
|                                                                         |
|  File Name ===> 'OPTIM.ODPP.ENV(V113)'                                  |
|                                                                         |
|     Enter the name of the data set that defines the Unix System Services |
|     environment variables required for the Optim data privacy providers.|
+-------------------------------------------------------------------------+
```

Optim will then attempt to perform the following validation:
- Verification that the entered data set exists
- Verification that Optim is able to load the Optim data privacy providers using the environment variables defined in the specified file

An error message will be produced if either of these validations failed. If validation is successful, message ENVIRONMENT SET will be displayed.

Note: Optim users will require appropriate read or write authority to the specified paths in the environment variable file to invoke the Optim data privacy providers.

Setting a user-defined environment variable file

If you want each Optim user to be able to specify their own environment variable file, set **Change Optim Data Privacy** to U on the Site Options panel and press Enter.

Each user will then be able to specify the name of their environment variable file through the User options interface in a similar manner as described above.

Note: Each user will be required to first define their environment variable file through the User options before they can invoke the Optim data privacy providers. They must also have appropriate read or write authority to the specified paths in the environment variable file.

Refer to the *Common Elements Manual* section on Data Privacy providers for details on using the individual providers to mask data.

Appendix A. Reuse DSN Subcommands and SQL

The DSN subcommands and SQL statements generated as part of the installation of Optim can be stored in a sequential file to be reviewed and executed using DB2I and SPUFI.

Since Optim must be installed in each subsystem in which it is to be used, this type of installation may be useful when, for example, installing in several subsystems. In this case, it may be more expedient to execute the FOPINSTL CLIST for one subsystem and then use DB2I and SPUFI to display, modify, and execute the SQL statements to install Optim in the other subsystems.

To store the SQL statements generated by the installation, enter `Yes` for the **Save SQL/DSN Output** on the Installation menu. You will be prompted for the name of a sequential file to be used to store the statements and subcommands. For more information see Chapter 3, "Installation Dialog," on page 11.

The DSN subcommands and SQL statements are stored in the sequential file in a SPUFI-compatible format. They are intermixed because they are interdependent. The Directory tables must be created prior to the Final Bind, which must be accomplished before Grant Authority, and so forth.

Each statement is formatted as follows:

```
---------------------------------------
-- date/timestamp
SQL statement
;

---------------------------------------
```

The date and timestamp document the time the statement was generated and are for information only.

The following is a sample of the stored SQL statements and DSN subcommands:

```
---------------------------- BEGIN DSN SUBCOMMAND ----------------------=

-- 2014-08-04-11:57:26
   BIND PLAN (FOP1130) ACTION(REPLACE)  -
       PKLIST(FOPPACKAGEMB.*,FOPPACKAGEME.*) CURRENTDATA(YES)  -
       VALIDATE(BIND) ISOLATION(CS) FLAG(I) ACQUIRE(USE)  -
       RELEASE(COMMIT) EXPLAIN(NO) ENCODING(EBCDIC) SQLRULES(STD)  -
       ;
---------------------------- END DSN SUBCOMMAND ------------------------=
```

You can use SPUFI to edit and execute the SQL statements. DSN subcommands must be commented out prior to execution. (To execute the DSN subcommands in batch you must first modify them.)

Any editing changes made to these statements have no effect when the FOPINSTL CLIST is used again to install Optim.

Additional Notes

The SET CURRENT SQLID statements executed to change a Secondary Authorization ID specified for the Install are not stored in the output dataset. You must insert these statements prior to executing SPUFI. The SQLID must be changed to an authorized SQLID at the beginning of the process and changed to the plan qualifier authorization ID prior to Dropping and Creating the Synonyms for the Final Bind.

When you use the saved statements to install into another subsystem you must execute the following options provided by FOPINSTL to prepare the product for execution.

For a Full Install, complete Steps 8 through 11 on the Full Installation menu:

1. 8 - Enable Product(s) for Execution
2. 9 - Setup Optim Directory and Sample Database
3. 10 - Customize Site Options
4. 11 - Customize the Batch Execution Environment

For an Upgrade, do Steps 7 and 8 on the Upgrade menu:

1. 7 - Copy Site Options from Previous Release
2. 8 - Customize the Batch Execution Environment

Appendix B. Review SQL Statements and DSN Subcommands

The SQL statement or DSN subcommand generated by the steps to install or upgrade Optim can be reviewed by specifying Y in response to the **Review SQL** or **Review DSN subcommand** prompt on any panel that displays the prompt.

The SQL or DSN subcommand for the current step is then displayed. Press ENTER to execute the SQL statement or DSN subcommand. Use END to cancel the request and redisplay the panel that generated the SQL statement or DSN subcommand.

For SQL Statements

If you press ENTER and an SQL error is encountered, the DB2 SQL Error panel is displayed. Type SQL at the command prompt to display the SQL statement or END to redisplay the panel that generated the SQL in error.

For DSN Subcommands

The DSN messages are echoed to the terminal. If an error is encountered, the panel is redisplayed so you can correct the error and re-execute. You can not directly modify the SQL statements or DSN subcommand for any step. The SQL statement and DSN subcommand display is for review only. You can modify your responses to the prompts on any panel to alter the SQL generated by the Install Facility.

The following figure shows a sample DSN subcommand for Option 1 Bind Plan for Product Install on the Full Installation menu. The current function is always included in the title.

```
-- Review DSN Subcommand: Bind Plan for Product Install -----------------
COMMAND ===>

DSN Subcommand to be executed:

BIND PLAN (FOPnnn) OWNER(SENTNEM) MEMBER(DSQLDB2)
LIBRARY('SFOPINST')


Press ENTER key to continue
Press END key to respecify
```

Figure 63. Review DSN for Bind Plan for Product Install

Appendix C. Run Multiple Releases

Sites may want to run more than one release of Optim at a given time. For example, you may wish to perform testing with the new release before switching your production system from the previous release.

Therefore, while you are testing the new release, the production system must continue to run with the previous release.

There is more than one way to handle this situation. The main issues to consider are the management of the ISPF and CLIST libraries and the management of the Directory and the product plan.

CLIST Libraries

If you intend to run more than one release of Optim, the CLISTs for each release must be stored in a separate library.

Chapter 2, "Set Up and Modify the Optim CLIST," on page 7 instructs you to copy all the CLISTs into a CLIST library before performing the install.

To execute the installation procedure without any further changes to the existing system, edit the CLIST member FOPCUST as documented in Chapter 2, "Set Up and Modify the Optim CLIST," on page 7, with the appropriate values for the libraries for the release being installed. All Optim ISPF libraries should be LIBDEF'd at execution time. You should also choose a plan name different than the one used for the current release in the SET statement for the plan name.

After FOPCUST is modified, you must identify the CLIST libraries for the installation procedure. Select ISPF Option 6 and enter:

```
ALTLIB ACTIVATE APPL(CLIST) DSN('SFOPCLST')
```

where '*dsn.of.install.lib*' is the dataset name of the installation CLIST library to be used for this execution of the installation procedure.

Note: When the installation procedure is complete, use ALTLIB RESET.

You can begin executing the installation dialog for the new release without interfering with the existing release. Also, from ISPF Option 6, enter:

```
FOPINSTL
```

Directory Considerations

You will then proceed through the steps in the installation process. Be sure to specify a unique plan name for each version of Optim. The plan name used during the BIND step should be the same as the name provided for the DEFPLAN variable in the FOPCUST CLIST. Also, you must use separate Directories for each release.

Separate Directories

After installing the new release and creating a new Directory, you can use the Optim Export and Import facilities to transport the Optim object definitions from the previous release to the new release.

The Directory objects are migrated in two steps:

1. Export

The Export facility copies the objects from the previous release to a data set, which may be used repeatedly by the Import facility to copy the objects to any Optim Directory.

Invoke the Export facility from the main menu by selecting Option 6 DEFINITIONS. Then select Option E EXPORT on the Choose a Definition Option menu to display the EXPORT Process panel. To export all definitions from a previous release, specify an output data set and indicate ALL for **Definition Type**.

2. Import

The Import facility copies the objects in the specified data set to the new release.

Invoke the Import facility by selecting Option I IMPORT on the Choose a Definition Option menu. On the IMPORT Process menu, specify the appropriate data set and indicate ALL for **Definition Type** to import all definitions from a previous release.

For details about the Export and Import facilities, see the *Common Elements Manual*.

Unique Qualifier and Collection-ID Prefix

In addition to using a unique plan name for each version, you should also specify a unique plan qualifier and a unique Collection-ID prefix.

For each plan that you bind, you are given the opportunity to create synonyms for the Optim Directory and the DB2 Catalog. Any existing synonyms are dropped before the "new" synonyms are created. If you specify the plan qualifier used for an existing plan, that plan is invalidated when the synonyms are dropped for the "new" plan.

Switching Versions

Once the versions are installed, you can switch from one version to the other.

Prior to the installation, you modified FOPCUST to identify the ISPF libraries for the release. This will handle the selection of the appropriate set of ISPF libraries during product execution. However to make sure that the correct version is invoked, use the ALTLIB command in the startup CLIST, FOP, to select the appropriate CLIST library. Perform the following steps:

- Edit the member FOP to include the ALTLIB commands to point to the CLIST library for the release. This member contains comment lines that show where these commands need to be inserted and the syntax. (The ALTLIB commands should be the same as entered from ISPF Option 6 prior to beginning the installation dialog.)
- Establish how the FOP CLIST is to be invoked. Use one of the following:
  - Invoke the CLIST from ISPF Option 6 by entering:

        EXEC 'hlq.slq.SFOPCLST(FOP)'

    where 'hlq.slq.SFOPCLST(FOP)' is the fully qualified dataset name of the CLIST.
  - Create a menu that executes the EXEC command as a result of a menu selection.
  - It may be more convenient to copy the FOP CLIST into a CLIST library allocated to SYSPROC at logon. If you have a previous version of Optim in your SYSPROC library to invoke one release, you can rename the new Optim release version (for example, FOP71) to be used to invoke another. Since the new release version uses ALTLIB to select the CLIST library, there are no conflicts with the current release.

You can run different releases of Optim in one DB2 subsystem. Execution is keyed to the FOPCUST CLIST, the plan name, and the Optim synonyms.

Extract Files and Archive Files

Extract Files or Archive Files created or used with previous releases of the Optim products can be used with the current release.

However, due to enhancements, any Extract File or Archive File created or used with the current release is not usable with previous releases of the Optim products.

Column Map Exits

Column map exits created with earlier releases of the Optim products must be recompiled to be usable in Release 7.2 and later releases.

Appendix D. Installation and Pre-training Checklist

The following is an installation checklist. Review this list to verify that the installation is successful and that you are ready for the training session.

1. Load the sample database. To load or refresh the sample database at any time, use Option 3 **Maintain Directory and Sample Database** on the Installation menu. If you are preparing for a training session, it is recommended that you refresh the sample database. (Chapter 6, "Directory Maintenance," on page 89 discusses loading the sample database.)
2. Invoke Optim using the site-chosen method.
3. On the main menu, specify the name of the subsystem containing the sample database and press ENTER. For the training, write the subsystem name on the Instructor Prep Sheet.
4. Verify that the SQLID is profiled.

The rest of this checklist is divided into information that is product specific. If you installed the Optim components listed, review all of the information, otherwise, review only the pertinent products:
- "Access"
- "Archive" on page 204
- "Compare" on page 205
- "MOVE" on page 206
- "Optim Legacy—IMS" on page 207
- "Optim Legacy—VSAM" on page 208.

Access

If you have installed Access or are preparing for a training session, review the following steps:

Edit Table
> On the main menu, select Option 2 and press ENTER.
>
> On the Choose a DB2 Table/View to EDIT panel, enter the following values for:
>
> **Creator ID**
>> Specify FOPDEMO.
>
> **Table Name**
>> Specify OPTIM_CUSTOMERS.
>
> **Begin Edit Session with**
>> Specify D.
>
> Press ENTER.

Review Data
> The data from the table FOPDEMO.OPTIM_CUSTOMERS should be displayed. Type JOIN OPTIM_ORDERS on the command line and press ENTER.
>
> One row from the table FOPDEMO.OPTIM_CUSTOMERS should be displayed along with related rows in the table FOPDEMO.OPTIM_ORDERS.
>
> If no orders are displayed, position the cursor on the row from FOPDEMO.OPTIM_CUSTOMERS and scroll the rows until joined OPTIM_ORDERS rows are displayed.

Test Complete
> The test is complete. Press PF3 until you return to the main menu. Press PF3 again to exit.

Archive

If you have installed Archive or are preparing for a training session, review the following steps:

Archive

On the main menu, select Option 9 and press ENTER. Select Option 1 on the Archive and Restore menu and press ENTER.

Archive Process

On the ARCHIVE Process menu specify `P` as **Type of Access Definition to Use for Archive**. Enter the following values:

GROUP

Specify `GRP`.

USER   Specify `FOPDEMO`.

NAME

Specify `ARCHIVE`.

Select Option 1 TABLES and press ENTER.

Select Tables/Views for AD

On the Select Tables/Views for AD panel, specify:

Default Creator ID

Specify `FOPDEMO`.

Start Table

Specify `OPTIM_ORDERS`.

Press ENTER. OPTIM_ORDERS should be displayed in the table list area of the panel.

Type `GET TABLES RELATED` on the command line and press ENTER. The list now includes OPTIM_CUSTOMERS and OPTIM_DETAILS, as well as OPTIM_ORDERS.

Type `ARC` in **Cmd** for the OPTIM_ORDERS table name and press ENTER. The Specify Archive Criteria for AD panel is displayed.

Specify Archive Criteria for AD

Type `DC` in **Cmd** for the OPTIM_ORDER_SHIP_DATE column and press ENTER.

On the Define Date Criteria pop-up, specify these values for:

Date Format

Specify `YY/MM/DD`.

Pivot Year

Specify `55`.

Specific Date

Specify `1998-03-01`.

Use PF3 until you are returned to the ARCHIVE Process menu.

Perform Archive

On the ARCHIVE Process menu select Option 3 PERFORM and press ENTER. The Specify ARCHIVE Parameters and Execute panel is displayed, specify these values for:

Archive File DSN

Specify *qual*`.FOPDEMO.ARC.FILE` (where *qual* is a valid high-level qualifier)

Archive Group

Specify `ARC1998`.

Archive Description

Specify `FEBRUARY`.

Press ENTER. A pop-up allocation window is displayed. It contains default allocation specifications. Press ENTER. A message verifies that the dataset has been allocated.

Type B at the prompt for online or batch to run the process in batch. The Job Card and Print Options panel is displayed. Specify the required parameters to generate the JCL and press ENTER.

The JCL necessary to archive the data in batch is displayed. Verify that the Optim load library name is correct. Type CANCEL on the command line and press ENTER.

Test Complete
> The test is complete, press PF3 until you are returned to the main menu. Press PF3 again to exit.

Compare

For Compare, perform the following steps:

Compare
> On the main menu, select Option 8 and press ENTER.

Compare Process
> On the COMPARE Process menu, select Option 2 MULTIPLE and specify T as **Type of Compare Definition to Use**. Press ENTER.

Specify Compare Source Types
> On the Specify COMPARE Source Types panel, specify these values for:

> **Source 1**
> > Specify 1.

> **Source 2**
> > Specify 3.

> Press ENTER. Specify the name of the Extract File you unloaded during the installation process on the panel prompting for Source 1. Press ENTER.

Compare Process Table Map
> On the Compare Process Table Map panel, make sure that the following values are displayed:

```
Src 1 CID     FOPPROD
Src 2 CID     FOPDEMO

table names   OPTIM_CUSTOMERS
              OPTIM_ORDERS
              OPTIM_DETAILS
              OPTIM_ITEMS
              OPTIM_SALES
```

> **Note:** If not already specified, type FOPDEMO at the prompt for **Src 2 CID**. Also, if OPTIM_SHIP_TO is displayed in the list of table names, clear the OPTIM_SHIP_TO table name by over-typing it with spaces.

> Press PF3.

Perform Comparison
> The Specify COMPARE Parameters and Execute panel is displayed, specify the following value for:

> **Compare File DSN**
> > Specify *qual*.FOPDEMO.SAMP.CMP (where *qual* is a valid high-level qualifier).

> Press ENTER. As the comparison is performed, status information should be displayed. Verify the results by referring to the Compare Summary Selection List panel in the *Compare Introduction* (see the Compare File Summary section in that document). The results from your execution may differ from the figure.

Batch Repeat the compare process in batch.

Type B at the prompt for online or batch to run the process in batch. The Job Card and Print Options panel is displayed. Specify the required parameters to generate the JCL and press ENTER.

The JCL necessary to extract the data in batch is displayed. Verify that the Optim load library name is correct. Type CANCEL on the command line and press ENTER.

Test Complete

The test is complete, press PF3 until you are returned to the main menu. Press PF3 again to exit.

MOVE

For Move, perform the following steps:

Migrate

On the main menu, select Option 7 and press ENTER.

Select Option 1 on the Data Migration menu and press ENTER.

Extract Data

On the EXTRACT Process menu, select Option 1 TABLES and specify T as **Type of Access Definition to Use**. Press ENTER.

Select Tables/Views

On the Select Tables/Views for AD panel, specify these values for:

Default Creator ID
Specify **FOPDEMO**.

Start Table
Specify **OPTIM_CUSTOMERS**.

Press ENTER. OPTIM_CUSTOMERS should be displayed in the table list area of the panel.

Type GET TABLES RELATED on the command line and press ENTER. The list now includes OPTIM_SALES and OPTIM_ORDERS, as well as OPTIM_CUSTOMERS.

Point-and-Shoot

Type POINT on the command line and press ENTER.

A pop-up should be displayed. It indicates that you are starting Row List Processing. Do not type anything. Press ENTER.

The data from the table FOPDEMO.OPTIM_CUSTOMERS should be displayed. On the first OPTIM_CUSTOMERS row, type SSR in the **Cmd** area. Then tab down several rows and type SSR in the **Cmd** area for another OPTIM_CUSTOMERS row. Press ENTER. A message displays reflecting the number of rows you selected. Also, an S is placed in the status flag (**F**) for each selected row.

Use PF3 to terminate Point-and-Shoot. A pop-up should be displayed. It indicates that you are terminating Row List Processing. Do not type anything. Press ENTER.

The Select Tables/Views for AD panel is redisplayed. Press PF3 until you return to the **EXTRACT Process** menu.

Perform Extract

On the **EXTRACT Process** menu, select Option 4 PERFORM and press ENTER.

The Specify EXTRACT Parameters and Execute panel is displayed, specify the following value for:

Extract File DSN
Specify *qual*.FOPDEMO.EXT.TST (where *qual* is a valid high-level qualifier).

Press ENTER. A pop-up allocation window is displayed. It contains default allocation specifications. Press ENTER. A message verifies that the dataset has been allocated.

Type B at the prompt for online or batch to run the process in batch. The Job Card and Print Options panel is displayed. Specify the required parameters to generate the JCL and press ENTER.

The JCL necessary to extract the data in batch is displayed. Verify that the Optim load library name is correct. Type CANCEL on the command line and press ENTER.

Test Complete
> The test is complete, press PF3 until you are returned to the main menu. Press PF3 again to exit.

Optim Legacy—IMS

For Optim Legacy, perform the following steps:

Define IMS Environment
> On the main menu, select Option 6 and press ENTER.
>
> Select Option 7 on the Choose a Definition Option menu and press ENTER.
>
> On the Choose an IMS Environment panel, make sure that FOPDEMO is displayed at the **Environment Name** prompt, and press ENTER.
>
> On the Modify IMS Environment panel, type in the dataset names of your IMS program libraries at the **IMS Program Libraries** prompt, and the name of the IMS Buffer Parameters member at the **DFSVSAMP DSN/Member Name** prompt (if required).
>
> Press PF3 until you are returned to the main menu.

Migrate
> On the main menu, select Option 7 and press ENTER.
>
> Select Option 1 on the **Data Migration** menu and press ENTER.

Extract Data
> On the **EXTRACT Process** menu, select Option 1 TABLES and specify T as **Type of Access Definition to Use**. Press ENTER.

Select Tables/Views
> On the Select Tables/Views for AD panel, specify these values for:
>
> **Default Creator ID**
> > Specify FOPDEMO.
>
> **Start Table**
> > Specify OPTIM_EMPLOYEE.
>
> Press ENTER. OPTIM_EMPLOYEE should be displayed in the table list area of the panel.
>
> Type GET TABLES RELATED on the command line and press ENTER. The list now includes OPTIM_DEPARTMENT, OPTIM_POSITION, and OPTIM_JOBCODE, as well as OPTIM_EMPLOYEE.
>
> Press PF3 until you return to the **EXTRACT Process** menu.
>
> **Note:** When you exit the Select Tables/Views for AD panel, the Associate Legacy Tables with Data Sources panel is displayed. Press PF3 to display the Associate IMS Segments with IMS Database Datasets panel. Press PF3 to return to the EXTRACT Process menu.

Perform Extract
> On the **EXTRACT Process** menu, select Option 4 PERFORM and press ENTER.

The Specify EXTRACT Parameters and Execute panel is displayed, specify the following value for:

Extract File DSN
> Specify `qual`.`FOPDEMO.EXT.TST` (where `qual` is a valid high-level qualifier).

Type `B` at the prompt for online or batch to run the process in batch. Press ENTER. The Job Card and Print Options panel is displayed. Specify the required parameters to generate the JCL and press ENTER.

The JCL necessary to extract the data in batch is displayed. Verify that the Optim load library name is correct. Type CANCEL on the command line and press ENTER.

Test Complete
> The test is complete, press PF3 until you are returned to the main menu. Press PF3 again to exit.

Optim Legacy—VSAM

For Optim Legacy, perform the following steps:

Migrate
> On the main menu, select Option 7 and press ENTER.
>
> Select Option 1 on the Data Migration menu and press ENTER.

Extract Data
> On the EXTRACT Process menu, select Option 1 TABLES and specify `T` as **Type of Access Definition to Use**. Press ENTER.

Select Tables/Views
> On the Select Tables/Views for AD panel, specify these values for:
>
> **Default Creator ID**
> > Specify `FOPDEMO`.
>
> **Start Table**
> > Specify `OPTIM_VENDOR`.
>
> Press ENTER. OPTIM_VENDOR should be displayed in the table list area of the panel.
>
> Type `GET TABLES RELATED` on the command line and press ENTER. The list now includes OPTIM_VENDITEM as well as OPTIM_VENDOR.
>
> Press PF3 until you return to the EXTRACT Process menu.
>
> **Note:** When you exit the Select Tables/Views for AD panel, the Associate Legacy Tables with Data Sources panel is displayed. You must associate the FOPDEMO.OPTIM_VENDOR and FOPDEMO.OPTIM_VENDITEM Legacy Tables with a specific data source. You can either use the default dataset names, type in the dataset names, or use wildcards to obtain a selection list. For example, *qual1.qual2*.FOPDEMO.OPTIM_VENDOR or *qual1.qual2*.FOPDEMO.OPTIM_VENDITEM.

Perform Extract
> On the EXTRACT Process menu, select Option 4 PERFORM and press ENTER.
>
> The Specify EXTRACT Parameters and Execute panel is displayed, specify the following value for:
>
> **Extract File DSN**
> > Specify `qual`.`FOPDEMO.EXT.TST` (where `qual` is a valid high-level qualifier).
>
> Type `B` at the prompt for online or batch to run the process in batch. Press ENTER. The Job Card and Print Options panel is displayed. Specify the required parameters to generate the JCL and press ENTER.

The JCL necessary to extract the data in batch is displayed. Verify that the Optim load library name is correct. Type CANCEL on the command line and press ENTER.

Test Complete

The test is complete. Press PF3 until you are returned to the main menu. Press PF3 again to exit.

Appendix E. Collecting Diagnostic Information

To diagnose and resolve problems with your Optim solution, IBM Technical Support might ask you to obtain diagnostic information.

This Appendix describes the following methods of collecting diagnostic information:
* Using the trace facility.
* Allocating and capturing a SYSMDUMP.
* Using the TSO TRANSMIT and RECEIVE commands.

Using the Trace Facility

IBM Technical Support may ask you to enable tracing to help analyze a problem.

Note: It is recommended that you enable tracing only when requested by Technical Support.

To enable tracing, enter the TRACE ON command on any screen that contains a command line. Once tracing is activated, you can run a process to obtain the information requested by Technical Support.

Once you have completed the process, enter the TRACE CLOSE command on the command line to disable tracing. (If you do not enter this command, tracing is disabled when you exit Optim.)

The trace output is routed to the location specified in **Trace SYSOUT Job Class** on the Site Options panel (see "Customizing the Site Options" on page 49). You can view the trace output through the SDSF Held Output panel. This information should be forwarded to Technical Support, when requested.

Alternatively, you can pre-allocate a sequential file for the trace output. Specifically, the DCB parameters should be:

```
RECFM=FB
DSORG=PS
LRECL=133
BLKSIZE=0
```

Specifying `BLKSIZE=0` is recommended. This allows the value to be calculated by the system at run time. After you preallocate the trace file, you must assign it to PSTRACE prior to executing the TRACE ON command, using the following command:

```
TSO ALLOC F(PSTRACE) DSN('dsn.trace.file')
```

Batch Execution

To enable tracing in batch, review the JCL prior to job submission.

For example:

```
PARM= 'CON UTILITY ssn db2plan sqlid tsoid =TRACE=*',
```

You can view the trace output through the SDSF System Log display. This information should be forwarded to Technical Support, when requested.

Allocating and Capturing a Diagnostic Dump

When an Optim process causes an abend, you can capture a SYSMDUMP for your TSO session or batch job.

You should first verify that you do not already have a SYSUDUMP, SYSABEND, or SYSMDUMP allocated to your TSO session using the TSO **ISRDDN** command. Browse the list of allocated data sets, and use the **F** line command to free an allocation.

Allocate a data set, *tsoprefix*.SYSMDUMP, by selecting ISPF Option 6 and entering the following command:

```
ALLOC F(SYSMDUMP) DA(SYSMDUMP) SPACE(50 50) CYL VOL(volser)
```

where 50 cylinders of primary and secondary space are allocated for the data set and *volser* is the volume serial number of the device on which the data set will reside. DCB parameters are not required.

Because Optim is an LE application, before a diagnostic dump is collected, allocate the following data set to the TSO session or in the batch JCL:

- DDNAME CEEOPTS in stream or allocated data set with the following contents:
- TRAP(OFF)

For example:

```
//CEEOPTS DD *
TRAP(OFF)
/*
```

Once both data sets are allocated, run the process that caused the abend. (If you receive TSOI line mode messages, press ENTER.) The SYSMDUMP is captured. To verify that the entire dump is captured, the standard Symptom Dump with registers should be listed, followed by the message:

```
IEA993I SYSMDUMP TAKEN TO tsoprefix.SYSMDUMP
```

Note: Once you confirm the dump is captured, you can use the TSO TRANSMIT and RECEIVE commands to transfer files to Technical Support.

An alternate method for collecting the diagnostic dump is to use the z/OS **SLIP** command. Consult your system administrator or refer to IBM publications for setting up and processing a SLIP DUMP.

Transferring Files Using TRANSMIT and RECEIVE

Technical Support may ask you to send files (for example, Extract Files, Archive Files, dumps, and so on) for review. You can send files using the TSO TRANSMIT (or XMIT) command.

Issue the TRANSMIT command from the source system to invoke the IEBCOPY utility and transmit the unloaded data set in sequential format. For example:

```
XMIT (A.B) DSN('input.dsn') [MEMBERS(mem1,mem2,...)] OUTDSN('output.dsn')
```

where A.B are fixed placeholders, *input.dsn* is the data set to be transmitted, *mem* specifies members of the PDS to be transmitted, and *output.dsn* is the data set to which output is directed. Note that the MEMBERS operand is optional and, if omitted, all members of the PDS are unloaded to the OUTDSN.

Notes:
- You must transfer the data set using a binary format (instead of EBCDIC to ASCII translation).
- To avoid a B37 abend, you should pre-allocate the OUTDSN on the destination system. The DCB parameters should be:
  ```
  RECFM=FB
  DSORG=PS
  LRECL=80
  BLKSIZE=3200
  ```

Technical Support may need to send information to your site in TSO TRANSMIT format. You can unload this information using the TSO RECEIVE command. Use "Send File to Host" on your 3270 Emulator to transfer the file to the destination system. Once the file is transferred, issue the following TSO command:

```
RECEIVE INDSN('dsname')
```

where *dsname* is the data set name used for the file transfer. When prompted for additional parameters, enter the following:

```
DSN('outdsname') [VOL(volser)]
```

where *outdsname* is the name of the data set to contain the file being received, and *volser* is the volume serial number of the device on which the output data set will reside. Note that the VOL operand is optional and can be omitted if the PDS exists.

Appendix F. ODM Optim Connect

ODM Optim Connect provides access to data in Optim Archive Files for applications that use the ODBC and JDBC APIs.

The Optim z/OS Solution must be installed prior to installing Optim Connect.

Optim Connect is implemented using a custom driver that provides access to Archive Files or Archive File Collections. Optim Connect is a rich peer-to-peer networking product. A full set of manuals is included with Optim Connect. This appendix describes how to install, configure, and use Optim Connect to provide access to Archive Files.

Deployment Strategy

To be accessed using Optim Connect, an Archive File must be registered in an Optim Directory on the z/OS Optim Connect Server machine.

A primary Optim Connect Server resides on the machine that contains the Optim Directory and includes one or more Optim Connect data sources defined for Archive Files or Archive File Collections.

If JDBC is the sole API used to access archived data, a direct connection is made to the Optim Connect Server; however, you must install the Optim Connect Server Thin Client on each Windows machine that will use JDBC to access archived data.

If ODBC is used to access archived data, the thin ODBC client can connect directly to an Optim Connect Server without the need of a secondary server.

You can use Optim Connect Studio, which is included with your Optim Connect installation, to administer the server from a single Windows machine. Documentation is located in the ODM/Optim Connect Documentation subdirectory of the Optim z/OS Solution documentation CD.

Installation

The Optim z/OS solution provides the installation files for the ODM server in the Optim installation library SFOPINST. Use option 1.9 of the Optim z/OS installation interface and select the names FOPODMKT and FOPODMLD and press Enter. The interface will create these files:
qual1.qual2.FOP*nnn*.TRANSMIT.KIT
and
qual1.qual2 .FOP*nnn*.TRANSMIT.LOAD
where *qual1* and *qual2* are the first two qualifiers of your Optim z/OS installation. During the unload of Optim Connect data sets, Optim uses both *qual1* and *qual2* only if their total length is less than or equal to 12 characters. Otherwise, only *qual1* will be used. Files are included for the Thin Client and Optim Connect Server.

Note: The DSN prefix for the TRANSMIT.LOAD and TRANSMIT.KIT files must not exceed a length of 20 bytes.

To install Optim Connect on a z/OS machine, refer to the z/OS Optim Connect Server Installation Guide, located on the documentation CD. After installation, you must edit the Optim Connect USER.PROCLIB library and register the Optim Connect driver.

To make an ODBC connection from a Windows machine, you must install an ODBC Thin Client (refer to "Thin Client Installation" on page 222) and use Connect Studio to create a shortcut to the data source on the Optim Connect Server (refer to "Using the Data Source Configuration Wizard" on page 220).

To make a JDBC connection from a Windows machine, you must install the JDBC Thin Client. For more information, see "Thin Client Installation" on page 222.

Applying Maintenance to Optim Connect

After installing ODM Optim Connect, use the XMIT files included with the installation to apply required maintenance.

From the install Main Menu, select option 7 **Apply Optim Connect Maintenance**. The following panel displays with options for unloading the datasets:

```
FOPIUNXM --------- Unload Optim Connect Maintenance --------------------------
OPTION  ===>

Optim Connect Maintenance has been packaged in XMIT format. You may optionally
specify a Unit, Volume and STORCLAS to unload the maintenance. The maintenance
will be unloaded to the following datasets :
    QUAL1.QUAL2.FOPxxxx.LOAD
    QUAL1.QUAL2.FOPxxxx.LOADAUT
    QUAL1.QUAL2.FOPxxxx.LOADCDCY

Optional attributes for unloaded data sets:
    UNIT               ===>
    VOLUME             ===>
    STORCLAS           ===>
```

After you receive the maintenance Optim displays this prompt:

```
OPTIM CONNECT MAINTENANCE HAS BEEN RECEIVED. PLEASE UPDATE
YOUR NAVCMD CLIST AND THE OPTIM CONNECT DAEMON AND SERVER
STARTED TASKS WITH THE NEW LIBRARIES UNDER THE QUALIFIER:

        FOP.VB30BASE.FOP5336.
```

Migrating to a New Version of ODM Optim Connect

Releases of Optim prior to v7.2 included versions of ODM Optim Connect (or Attunity) that are no longer supported. If you accessed data sources with older versions of ODM Optim Connect, and plan to access them with the current ODM Optim Connect version, you need to migrate the data source definitions. This section shows the steps to migrate the data source definitions.

1. License - You need a new license for the updated version of Optim Connect. Contact Optim Support to obtain the license and apply it.
2. Copy all data source definitions from the prior (before Optim v7.2) release.
3. Define the data sources, using NAVCMD utility or the Data Source Configuration wizard. Refer to "Define a Data Source" on page 219.

ODM Optim Connect Server Installation

To install ODM Optim Connect on a z/OS machine, refer to the z/OS Optim Connect Server Installation Guide, located in the Documentation subdirectory of the Optim z/OS Solution documentation CD.

Follow the procedures in the Installation Instructions and General Post Installation Procedures sections of the Installation Guide. In the section Preparing the Installation, omit Step 1 and begin with Step 2 (Execute the PREPARE member of the FOPnnn.TRANSMIT.LIB). In the General Post Installation Procedures section, perform only the following steps:

1. Perform the procedures described in the Setting APF Authorization and Started Tasks section.

2. Start the daemon using the following command:

   `S ATTDAEMN`

3. Perform the procedures described in the Registering Optim Connect Server section. Use the license file provided by IBM Optim. These procedures require Connect Studio. To configure Connect Studio, see "Connect Studio Configuration."

After completing these General Post Installation Procedures, the ODM Optim Connect Server installation is complete. To complete the ODM Optim Connect installation, you must edit the Optim Connect USER.PROCLIB library and register the ODM Optim Connect driver, as described in "Register ODM Optim Connect Driver."

Connect Studio Configuration

Connect Studio is used to configure the ODM Optim Connect Server environment from a Windows machine.

Install Connect Studio

To install Connect Studio on a Windows machine, run the ConnectStudio_*nnnn*_windows.exe installation file, located in the Optim Connect Studio subdirectory of the Optim z/OS Solution documentation CD. If installing the Server or Connect Studio in a Windows environment, you must remove any older versions before installation.

Adding an ODM Optim Connect Server to Connect Studio

To add an ODM Optim Connect Server to Connect Studio:

1. Click **Start** > **All Programs** > **IBM Optim Connect** > **IBM Optim Connect Studio**.
2. In the Configuration explorer, right-click **Machines** and click **Add Machine** to open the Add machine dialog.
3. In the Add machine dialog, enter the **Host name/IP address** or `localhost` for the machine hosting Connect Studio.
4. Use the default **Port**, 2551 (or the TCP/IP port number assigned to the z/OS Optim Connect daemon).
5. Enter the **User name** and **Password** needed to connect to the machine.

Edit Optim Connect USER.PROCLIB

After you install Connect Studio, add the following libraries to the STEPLIB DD in the Optim Connect Service Started Task member ATTSRVR which is available in the Optim Connect USER.PROCLIB library:

- The Optim SFOPLLIB library
- The DB2 load library

Register ODM Optim Connect Driver

To register the ODM Optim Connect driver, use the ADDON file, named FOPODMAO, located in the Optim install library (SFOPINST). Use the NAV_UTIL utility to register the driver.

To register the ODM Optim Connect driver from NAV_UTIL:

1. From the Optim Connect (NAVROOT) USERLIB, execute NAVCMD.

   `TSO exec 'qual.USERLIB(NAVCMD)'`

   Where *qual* represents the qualifier(s) of your Optim Connect library.
2. From the Local> prompt, enter:

   `ADDON 'qual1.qual2.SFOPINST(FOPODMAO)'`

   Where *qual1* and *qual2* are the first two qualifier(s) of your Optim installation.
3. Enter QUIT to exit NAV_UTIL.

ADDON Definition

The ADDON definition is a sequential file or PDS member that includes the following statement:

```
<PST_GDB>
INIT-FUNCTION = ENV_ATSP_Attach
SHAREABLE-NAME = FOPOMN
TYPE = DRIVER
```

If you create the ADDON definition file in an ISPF editor, ensure that number mode is off, using the NUM OFF command.

Note: The INIT-FUNCTION = entry is case sensitive and must be entered exactly as shown, with only the first letter of the word Attach capitalized (that is, ENV_ATSP_Attach). Do not enter the word Attach with any letter capitalized other than the first letter.

Define the ODM Optim Connect Data Source

A data source definition is needed to access archived data. You can define a data source for each Archive File or Collection, or specify the Archive File or Collection in an ODBC or JDBC connection string.

Conventions

The following conventions used to describe these statements:

KEYWORD
>   Keywords are shown in uppercase for emphasis, but can be specified in lowercase or mixed case.

text   Variable text is shown in lowercase italics.

[ ]   An optional keyword or argument is shown in brackets.

{ }   A choice of settings from which only one must be selected is shown in curved brackets.

| A vertical bar separates options.

Data Source Definition

A data source definition is expressed in XML as follows:

```
<DATASOURCE
    NAME='datasourcename' TYPE='PST_GDB' READONLY='true'>
    <CONFIG DB2SSN='db2subsystemname'
    PLAN='optimplanname'
    {ARCV_FILE='archivefilename' | ARCV_ID='n' |
     COLLECTION='archivefilecollection' }
      [PST_AF_SUBSET={ 'AF_IN(n,n,...)' |
      'AF_DATE_RANGE
      (yyyy-mm-dd-hh:mm:ss,yyyy-mm-dd-hh:mm:ss)' } ]
    [TRACEPARM='traceparms'
     TRACEOUT= 'datasetname' | 'SYSOUT=class' ] />
</DATASOURCE>
```

Syntax

Use the following syntax to define a data source:

<DATASOURCE>
>   Specifies the name and type of the data source and information required to connect to the data source.

NAME='*datasourcename*'
>   The data source name, which can be up to 32 characters and cannot include hyphens ("-").

TYPE='PST_GDB'
>   The data source driver. The value for the ODM driver is 'PST_GDB'.

READONLY='*true*'
> The value for an Archive File or Archive File Collection is 'true.'

<CONFIG>
> Specifies configuration properties of a data source.

DB2SSN='*db2subsystemname*'
> The name of the DB2 subsystem that includes the Optim Directory in which the Archive File or Archive File Collection is registered. The Optim Directory is read to validate Archive File names or Archive File Collection names. This parameter has a length of 4 bytes.

PLAN='*optimplanname*'
> The Optim plan name. This parameter has a length of 8 bytes.

ARCV_FILE='*archivefilename*'
> The fully qualified Archive File name. This parameter has a length of 44 bytes.

ARCV_ID='*nn*'
> The Archive File ID number.

COLLECTION='*archivefilecollection*'
> The Archive File Collection name.

PST_AF_SUBSET='*AF_IN*'|'*AF_DATE_RANGE*'
> Subsets an Archive File Collection to specific Archive Files. Use one of the following parameters:
>
> **'AF_IN(*n,n,...*)'**
> > Archive Files to include, where *n* is an Archive File name or Archive File ID. If a specified Archive File cannot be found, the process will fail.
>
> **'AF_DATE_RANGE(*yyyy-mm-dd-hh:mm:ss*,*yyyy-mm-dd-hh:mm:ss*)'**
> > A range of Archive File creation dates. You must include the time of day (hh:mm:ss). You can use zeros for the time (for example, 00:00:00).

TRACEPARM='*traceparameters*'
> An optional attribute for turning on the Optim Trace file. This attribute should be used at the direction of Optim support. Do not use commas in the TRACEPARM attribute.

TRACEOUT='*datasetname*'|'SYSOUT=*class*'
> If the Trace file is used, indicate a preallocated z/OS dataset name or a SYSOUT class for the Trace file. You can also indicate the SYSOUT class in parentheses, for example, 'SYSOUT(*class*)'.

Define a Data Source

To access archived data using ODM Optim Connect, you need to define the data source. Data sources can easily be defined using the Data Configuration Wizard or the NAVCMD utility. In addition, you can use the wizard to edit or test an existing data source definition.

You can define an ODM Optim Connect data source for an Archive File using the NAVCMD utility from the ODM Optim Connect Server or Connect Studio from a Windows machine.

NAVCMD

To define a data source from NAVCMD, edit the NAV binding.

1. From the Optim Connect (NAVROOT) USERLIB, execute NAVCMD.

   `TSO exec 'qual.USERLIB(NAVCMD)'`

   Where *qual* represents the qualifier(s) of your Optim Connect library.

2. From the Local> prompt, enter:

   `EDIT BINDING NAV`

   An ISPF edit session displays the NAV binding.

3. Edit the <datasource> statement to name the data source.

4. Edit the <config> statement to identify an Archive File or Archive File Collection and the Optim Directory in which it is registered.

5. Save the binding.

6. Enter QUIT to exit NAVCMD.

Using the Data Source Configuration Wizard

To access archived data using ODM Optim Connect, you need to define the data source. Data sources can easily be defined using the Data Configuration Wizard, described in this section. In addition, you can use the wizard to edit or test an existing data source definition.

1. After adding a machine to Optim Connect Studio, configure a datasource.

2. In the Connect Studio Configuration explorer, expand the **Binding** list, and expand the member you want to work with.

3. Right-click **Data Sources** and select **New Data Source**.

4. On the New data source panel, select **Optim Archive** and enter a value for **Name:** and press **Next**.

5. The Optim Archive Configuration Parameters panel displays. This panel has the following fields:

DB2 Subsystem
> This parameter is required. Name of the DB2 subsystem. This is the XML mapping value for the DATASOURCE/CONFIG/@DB2SSN attribute.

Optim Plan
> This parameter is required. Supply the unique plan name for this Optim release. This is the XML mapping value for the DATASOURCE/CONFIG/@PLAN attribute.

Type This parameter is required. Select one data source type from the following:
> - Archive File Collection - (Default) - creates the XML attribute DATASOURCE/CONFIG/ @COLLECTION
> - Archive File Name - creates the XML attribute DATASOURCE/CONFIG/@ARCV_FILE
> - Archive ID - creates the XML attribute DATASOURCE/CONFIG/@ARCV_ID

Source
> This parameter is required. There is no default. Supply the value to be used for the XML attribute created for the data source type:
> - DATASOURCE/CONFIG/@COLLECTION attribute (Archive File Collection)
> - DATASOURCE/CONFIG/@ARCV_FILE attribute (Archive File)
> - DATASOURCE/CONFIG/@ARCV_ID (Archive ID)

Archive Subset
> This parameter is optional. It is enabled only when **Type** is Archive File Collection. Allowable values are:
>
> **'AF_IN(*n,n,..*)'**
> > Archive Files to include, where *n* is an Archive File name or Archive File ID.
>
> **'AF_DATE_RANGE(*yyyy-mm-dd-hh:mm:ss,yyyy-mm-dd-hh:mm:ss*)'**
> > A range of Archive File creation dates. You must include the time of day (*hh:mm:ss*). You can use zeros to specify the time (for example, 00:00:00).

Trace Parameters
> This parameter is optional. It is the XML mapping to value for DATASOURCE/CONFIG/ @TRACEPARM. There is no default.

Trace Output
> This parameter is optional. It is the XML mapping to value for DATASOURCE/CONFIG/ @TRACEOUT. There is no default.

6. After you have supplied values for the fields, click **Finish**.

Using the Data Source Configuration Wizard to Edit a Definition

You may need to edit an existing data source definition, and this can be done using the Data Source Configuration Wizard.

1. From the Optim Connect Studio panel, right-click on a data source that you want to edit. The Data Source Context menu displays:
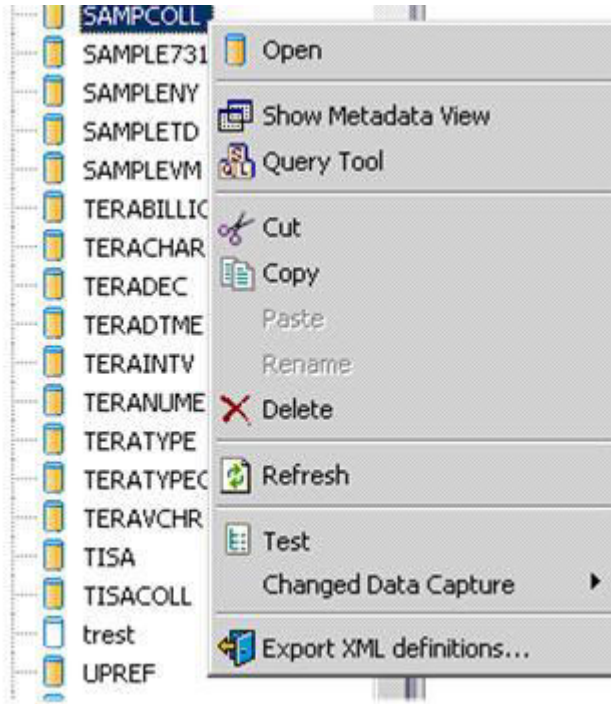
   Select **Open**.

2. A panel displays showing the data source name and type. The **Property** column lists all possible properties for this data type. The **Value** column displays the current value defined for each property. This area is modifiable and you can make any changes to the values, or you can click the **Restore Defaults** button to restore default values.

Note:

- In this panel, DB2SSN, PLAN, TRACEPARM and TRACEOUT are for Optim z/OS data sources only.
- DIRDB and ARCV_GUID do not apply to Optim/z.

Thin Client Installation

The Optim Connect Server Thin Client is required on each machine that uses JDBC to connect to ODM.

To install the Thin Client, run the ASTH5320-win32.exe installation file, located in the Attunity\Install Image\Attunity Thin Client subdirectory of the Optim z/OS Solution documentation CD.

Runtime Connection Information

The *Optim Connect Guide and Reference* manual, available with your installation, specifies the format and contents of the connection strings for ODBC and JDBC.

An Archive File or Collection defined in the connection string overrides the corresponding specification in the ODM data source definition. If an Archive File or Collection is not identified in the data source, you must specify an Archive File or Collection in the connection string.

Connection Parameters

The following are special connection parameters that can be used when connecting to an ODM data source from ODBC or JDBC:

```
DSNCONFIG=datasourcename1(
   [ ARCV_FILE=archivefilename | ARCV_ID=n |
      COLLECTION=archivefilecollection  ]
         [PST_AF_SUBSET={ 'AF_IN(n,n,...)' |
         'AF_DATE_RANGE
         (yyyy-mm-dd-hh:mm:ss,yyyy-mm-dd-hh:mm:ss)'|
          'AF_ID_RANGE(x,y'}]
   [ PSTTRACE=COMP (n n ...) [COMP (n n...)]]])
    &datasourcename2(...)
DSNPASSWORDS=datasourcename=domainname/userID/
     password & datasourcename2=...
QPTDPNAME=primaryservername:daemonportnumber
```

DSNCONFIG=*datasourcename*
> The ODM data source name.

ARCV_FILE=*archivefilename*
> The fully qualified Archive File name.

ARCV_ID=*n*
> The Archive File ID number.

COLLECTION=*archivefilecollection*
> The Archive File Collection name.

PST_AF_SUBSET='*AF_IN***' | '***AF_DATE_RANGE***'| '***AF_ID_RANGE***'**
> Subsets an Archive File Collection to specific Archive Files. Use one of the following parameters:

> **'AF_IN(***n,n,...***)'**
>> Archive Files to include, where *n* is an Archive File name or Archive File ID. If a specified Archive File cannot be found, the process will fail.

> **'AF_DATE_RANGE(***yyyy-mm-dd-hh:mm:ss***,***yyyy-mm-dd-hh:mm:ss***)'**
>> A range of Archive File creation dates. You must include the time of day (hh:mm:ss). You can use zeros for the time (for example, 00:00:00).

> **'AF_ID_RANGE(***x,y***)'**
>> A range of Archive File IDs, where *x* is the start and *y* is the end.

PSTTRACE=*COMP (n n ...)***'**
> An optional attribute for turning on the Optim Trace file. This attribute should be used at the direction of Optim support. Do not use commas in the PSTTRACE attribute.

TRACEOUT='*datasetname***' |**
'SYSOUT=*class***'**
> If the Trace file is used, indicate a preallocated z/OS dataset name or a SYSOUT class for the Trace file. You also can indicate the SYSOUT class in parentheses, for example, 'SYSOUT(*class*)'.

DSNPASSWORDS=*datasourcename=domainname/userID/password*
> The user ID and password for Archive File Security authentication. The *datasourcename* is the data source to which the credentials apply. The *domainname* is optional. You can use forward or backward slashes within the parameter. If this parameter is not used, the credentials of the user running the Optim Connect Server will be used for Archive File Security authentication.

QPTDPNAME=*primaryservername: daemonportnumber*
> The server name and port number of the Optim Connect daemon. Required when the connection string is passed from a secondary server.

Archive File Collections

An Archive File Collection is a list of Archive Files that can be logically unioned together as a single data source for ODM access.

For example, ODM uses an Archive File Collection to provide access to data in multiple Archive Files, even if all files do not include a specific table or column or if the attributes of data in a column vary from file to file. ODM processes Archive Files in the order they are listed in the Archive File Collection Editor. For information about creating Archive File Collections, see Definitions: Archive Collections in the *Archive User Manual*.

Unioned Tables

Tables with matching Creator IDs and names in separate Archive Files will be unioned.

To be processed, a table need not exist in every Archive File. ODM is not case-sensitive.

Matching tables are not required to have the same columns. The union will include all column names in the matching tables. Rows from a table that do not include a column found in another table will use a default value such as NULL, a default date specified in the Archive File Collection Editor, or an appropriate data type (spaces, zero, and so on).

Column Compatibility

All columns with the same name that are in tables with matching Creator IDs and names must have compatible attributes.

If columns have different but compatible attributes, a compatible attribute will be used for those columns. The column compatibility rules for the Compare Process apply to Archive File Collections. For information about comparison Compatibility Rules, see the *Common Elements Manual*.

For example, columns COLX DECIMAL(8,2) and COLX DECIMAL(10,0) will become COLX DECIMAL(12,2).

If a compatible attribute cannot be found (for example, COLX INTEGER and COLX TIMESTAMP), the Archive File Collection Editor will display an error message.

Unioned Indexes

Archive Indexes for unioned tables may also be unioned.

The following rules apply to unioned indexes:
- Each Archive File that includes the table must also include the index.
- ODM will use a unioned index until a column with a different name or attribute is found (compatible attributes are not used). The unique column and remaining columns in the index will not be processed.

Archive File Collection Subsets

You can specify a subset of files in an Archive File Collection using a data source definition, connection string, or SQL WHERE clause.

For more information about data source definitions, see "Define the ODM Optim Connect Data Source" on page 218. For more information about connections strings, see "Runtime Connection Information" on page 222.

The PST_AF_SUBSET pseudocolumn is logically added to each archived table during ODM processing and allows you to create a subset using an SQL WHERE clause. Use the following syntax:

```
PST_AF_SUBSET='AF_IN(n,n...)'  |  'AF_DATE_RANGE(yyyy-mm-dd-hh:mm:ss,yyyy-mm-dd-hh:mm:ss)'
```

'AF_IN(n,n,...)'
> Archive Files to include, where *n* is an Archive File name or Archive File ID.

'AF_DATE_RANGE
(*yyyy-mm-dd-hh:mm:ss***,***yyyy-mm-dd-hh:mm:ss***)'**
> A range of Archive File creation dates. You must include the time of day (hh:mm:ss). You can use zeros for the time (for example, 00:00:00).

The subset criteria need only be specified once in an SQL statement. If subset criteria is specified in the data source or connection string, the WHERE clause subset must be a subset of that criteria.

PST_ARCHIVE_ID Pseudocolumn

The PST_ARCHIVE_ID pseudocolumn is logically added to each archived table during ODM processing and contains the Archive File ID of the Archive File that includes the table.

Use PST_ARCHIVE_ID to specify the Archive File from which a row is selected.

For example, you can use PST_ARCHIVE_ID to control a join by avoiding duplicate rows from a table in multiple files in the Archive File Collection. If the DETAILS table is related to the ITEMS table, and only the DETAILS table is unique across all files in the collection, then a join between the tables would result in duplicate ITEMS rows joined with DETAILS rows.

Note: In the following example, the table names from the sample database are shown without the OPTIM_ prefix.

To avoid duplicate rows, use the following syntax:
```
SELECT * FROM FOP.DETAILS, FOP.ITEMS
WHERE
FOP.DETAILS.ITEM_ID=FOP.ITEMS.ITEM_ID
AND
FOP.DETAILS.PST_ARCHIVE_ID=FOP.ITEMS.PST_ARCHIVE_ID
```

PST_ARCHIVE_FILES Table

An Archive File Collection includes a table named PST_ARCHIVE_FILES, which contains a row for each Archive File in the collection.

PST_ARCHIVE_FILES has the following columns:

Column Name	Data Type	Description
ARCHIVE_ID	INTEGER	Archive File ID
ARCHIVE_FILE_NAME	VARCHAR(256)	Archive File name
ARCHIVE_DATETIME	TIMESTAMP	Archive File creation date and time

You can use this table to query context data from the Archive Files in the collection. For example, use the following syntax to find the latest Archive File ID:
```
SELECT MAX(ARCHIVE_ID) FROM PST_ARCHIVE_FILES
```

Appendix G. Metering

The Optim solutions meter all archive and extract requests and record the amount of data processed. These metering records are produced automatically and stored with the source file.

The byte count of the internal, uncompressed representation of the data is accumulated. Metering details for specific data types follow:

String Data Types

Data Type	Metering Description	Sample	Length Descriptor	Accumulated Byte Count and Explanation
Fixed Length Character	length of the defined column is accumulated	CHAR(24)	N/A	24 - length of defined fixed length column
Variable Length Character	length of data as provided in the length descriptor, plus max size of the field required to store the length descriptor is accumulated	VARCHAR(10)	6	7 - value of length descriptor plus 1 byte for field required to store the length descriptor (assuming a max value for the length descriptor of 255)
Fixed Length National Character	length (total bytes, not characters) of the defined column is accumulated	NCHAR(24)	N/A	48 - assuming a default length of 2 bytes per character in this example, a total length of 48 for the defined fixed column is accumulated
Variable Length National Character	length of data (total bytes, not characters) as provided in the length descriptor, plus max size of the field required to store the length descriptor is accumulated.	NVARCHAR(10)	6	13 - assuming a default length of 2 bytes per character in this example, the value of the length descriptor (in bytes) multiplied by 2 plus 1 byte for the field required to store the length descriptor (assuming a max value for the length descriptor of 4000) is accumulated

Numeric data types

The length of the internal representation of data is accumulated (System/390 format and IEEE floating-point format).

Data Type	Metering Description	Sample	Length Descriptor	Accumulated Byte Count and Explanation
Small Integer	2 bytes - length of the internal representation	SMALLINT	N/A	2 - internal representation is 2 bytes
Integer	4 bytes - length of the internal representation	INTEGER	N/A	4 - internal representation is 4 bytes
Big Integer	8 bytes - length of the internal representation	BIGINT	N/A	8 - internal representation is 8 bytes

Data Type	Metering Description	Sample	Length Descriptor	Accumulated Byte Count and Explanation
Single Precision Floating-point or REAL	4 bytes - length of the internal representation	FLOAT(21)	N/A	4 - internal representation of floating-point of precision 21 is 4 bytes
Double Precision Floating-point	8 bytes - length of the internal representation	FLOAT(53)	N/A	8 - internal representation of floating-point of precision 53 is 8 bytes
Decimal or Numeric	INTEGER(p/2) +1, where p is the precision	DECIMAL(31,0)	N/A	16 - internal representation of decimal with 31 digits of precision is 16 bytes (31/2) + 1 = 16
Decimal Floating-point	Internal representation of decimal floating point with 16 digits of precision is 8 bytes. Internal representation of decimal floating point with 34 digits of precision is 16 bytes.	DECFLOAT(16)	N/A	8 - internal representation of decimal floating point with 16 digits of precision is 8 bytes

Date, Time, Timestamp data types

Data Type	Metering Description	Sample	Length Descriptor	Accumulated Byte Count and Explanation
Date	4 bytes - length of the internal representation	DATE	N/A	4 - internal representation is 4 bytes
Time	3 bytes - length of the internal representation	TIME	N/A	3 - internal representation is 3 bytes
Timestamp	INTEGER($(p+1)$ /2) + 7 where p is the precision	TIMESTAMP(6)	N/A	10 - ((6+1)/2)+7 = 10
Timestamp with Time Zone	INTEGER($(p+1)$ /2) + 10 where p is the precision	TIMESTAMP(6) WITH TIME ZONE	N/A	13 - ((6+1)/2)+10 = 13

XML Data Type

Data Type	Metering Description	Sample	Length Descriptor	Accumulated Byte Count and Explanation
XML data	length of the internal representation of XML data is accumulated	XML	2000	2000 - internal representation of XML data

Large object data types

Data Type	Metering Description	Sample	Length Descriptor	Accumulated Byte Count and Explanation
Character Large Objects	length of data as provided in the length descriptor, plus 4 bytes to store the length descriptor is accumulated	CLOB(1000000)	500000	500004 - value of length descriptor plus 4 bytes for size of length descriptor
Binary Large Objects	length of data as provided in the length descriptor, plus 4 bytes to store the length descriptor is accumulated	BLOB(1000000)	500000	500004 - value of length descriptor plus 4 bytes for size of length descriptor

DB2 Data Types

For MIXED data, all Shift-Out/Shift-In characters are included in byte accumulation. Highlighted in this table are unique data types for DB2 or common data types that are accumulated differently for DB2 due to a different representation of data as provided to Optim.

Data Type	Metering Description	Sample	Length Descriptor	Accumulated Byte Count and Explanation
Fixed Length Graphic Strings	total number of bytes for the defined column is accumulated	GRAPHIC(4)	N/A	8 - GRAPHIC(4) indicates 4 double-byte characters, therefore a total of 8 bytes is accumulated
Variable Length Graphic Strings	length of data as provided in the length descriptor multiplied by 2, plus 1 byte for the length descriptor is accumulated	VARGRAPHIC(10)	6	13 - value of length descriptor (number of double-byte characters) multiplied by 2 plus 1 byte for size of length descriptor
Double-byte Character Large Objects	length of data as provided in the length descriptor multiplied by 2, plus 4 bytes to store the length descriptor is accumulated	DBCLOB(1000000)	500000	1000004 - value of length descriptor (number of double-byte characters) multiplied by 2 plus 4 bytes for size of length descriptor
ROWID	17 bytes - length of the internal representation	ROWID	N/A	17 - internal representation of ROWID DB2 data type

IMS, VSAM, Sequential Table Data

For VSAM, Sequential, and IMS tables, bytes per column are accumulated according to the LENGTH value defined in the legacy table. If the field is defined with an Occurs clause, the LENGTH value multiplied by the number of Occurs is accumulated. Highlighted in this table are unique data types for legacy tables or common data types that are accumulated differently for VSAM, Sequential, and IMS data due to a different representation of data as provided to Optim.

Data Type	Metering Description	Sample	Length Descriptor	Accumulated Byte Count and Explanation
Fixed Length Graphic Strings	total number of bytes for the defined column is accumulated	TYPE GRAPH LENGTH 4	N/A	8 - TYPE GRAPH LENGTH 4 indicates 4 double-byte characters, therefore a total of 8 bytes is accumulated

Data Type	Metering Description	Sample	Length Descriptor	Accumulated Byte Count and Explanation
Variable Length Graphic Strings	length of data as provided in the length descriptor multiplied by 2, plus 2 bytes for the length descriptor is accumulated	TYPE VARGRAPH LENGTH 10	6	14 - value of length descriptor (number of double-byte characters) multiplied by 2 plus 2 bytes for size of length descriptor
Variable Length Character	length of data as provided in the length descriptor, plus 2 bytes for the length descriptor is accumulated	TYPE VCH LENGTH 8	2	4 - value of length descriptor plus 2 bytes for the size of length descriptor

IMS, VSAM, Sequential - Numeric Data

Data Type	Metering Description	Sample	Length Descriptor	Accumulated Byte Count and Explanation
Binary	total number of bytes specified in the LENGTH field of the legacy table definition	TYPE SHORT LENGTH 2	N/A	2 - a legacy table definition with TYPE BIN accumulates a length of 2
Other Numeric	total number of bytes specified in the LENGTH field of the legacy table definition	TYPE LTTYPE_NUM LENGTH 10	N/A	10 - value in the length field is accumulated

IMS, VSAM, Sequential - Date, Time, Timestamp Data

Data Type	Metering Description	Sample	Length Descriptor	Accumulated Byte Count and Explanation
Date	10 bytes - length of the character representation of DATE as provided to Optim	TYPE DATE LENGTH 10	N/A	10 - character representation of DATE data type as provided in legacy table is 10 bytes
Time	8 bytes - length of the character representation of TIME as provided to Optim	TYPE TIME LENGTH 8	N/A	8 - character representation of TIME data type as provided in legacy table is 8 bytes
Timestamp	19 - 32 bytes - length of the character representation of TIMESTAMP as provided to Optim	TYPE TIMESTAMP PRECISION 26 LENGTH 26	N/A	26 - character representation of this TIMESTAMP data type as provided in legacy table is 26 bytes
Timestamp with Time Zone	25 - 38 bytes - length of the character representation of TIMESTAMP_Z as provided	TYPE TIMESTAMP_Z PRECISION 30 LENGTH 30	N/A	30 - character representation of this TIMESTAMP with TIME ZONE data type as provided in legacy table is 30 bytes

IMS, VSAM, Sequential - Miscellaneous Data Types

Data Type	Metering Description	Sample	Length Descriptor	Accumulated Byte Count and Explanation
Other PL/I field types	total number of bytes specified in the LENGTH field of the legacy table definition	TYPE LTTYPE_OTHER LENGTH 10 OCCURSFROM 3 OCCURSTO 3	N/A	30 - value of the length field multiplied by the OCCURS value is accumulated
Group fields	no additional bytes are accumulated for group fields	LTTYPE_GROUP LENGTH 20	N/A	0 - no additional bytes are accumulated for group fields, as Optim will accumulate each of the elements individually

Sample Report

The following archive process report includes metering information. A portion of the report is shown. The amount of data processed by this request is displayed as 1,000,000,383,836 bytes, followed by gigabyte and terabyte values enclosed in parentheses.

```
Process Options:
   Process Mode         : Online
   Archive Data using   : DB2
   Limit Archive Rows   : 3000000

Total Number of Archive Tables              : 10
Total Number of Archived Rows               : 6906
Total Number of First Pass Start Table Rows : 704
Archive file data byte count : 1,000,000,383,836 Bytes (931.323 GB, 0.909  TB)
```

Notices

This information was developed for products and services offered in the U.S.A.

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

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

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

IBM World Trade Asia Corporation
Licensing 2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: 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 states 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 Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply 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:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: 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 states 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 Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply 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 Corporation
Department number/Building number
Site mailing address
City, State; Zip Code
U.S.A.

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.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

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 the names and addresses used by an actual business enterprise 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_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Java™ is a trademark of Sun Microsystems, Inc.

Linux is a registered trademark of Linus Torvalds.

UNIX is a registered trademark of The Open Group.

Microsoft and Windows are registered trademarks of Microsoft Corporation.

Other company, product, or service names may be trademarks or service marks of others.

Index

IBM ®

Printed in USA