Tivoli

Information Management for z/OS

Program Administration Guide and Reference

Version 7.1

SC31-8753-00

Information Management for z/OS

Program Administration Guide and Reference

Version 7.1

SC31-8753-00

Tivoli Information Management for z/OS Program Administration Guide and Reference

Copyright Notice

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both: IBM, the IBM logo, Tivoli, the Tivoli logo, AIX, CICS, CICS/ESA, DATABASE 2, DB2, DFSMS/MVS, DFSORT, IBMLink, Language Environment, MVS, MVS/DFP, MVS/ESA, NetView, OS/2, OS/2 WARP, OS/390, QMF, RACF, Redbooks, RMF, System/390, Tivoli Enterprise Console, TME 10, VTAM, z/OS.

Lotus and Notes are trademarks of Lotus Development Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

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

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

Other company, product, and service names mentioned in this document may be trademarks or service marks of others.

Notices

References in this publication to Tivoli Systems or IBM products, programs, or services do not imply that they will be available in all countries in which Tivoli Systems or IBM operates. Any reference to these products, programs, or services is not intended to imply that only Tivoli Systems or IBM products, programs, or services can be used. Subject to valid intellectual property or other legally protectable right of Tivoli Systems or IBM, any functionally equivalent product, program, or service can be used instead of the referenced product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by Tivoli Systems or IBM, are the responsibility of the user. Tivoli Systems or IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, New York 10504-1785, U.S.A.

Programming Interface Information

This publication documents intended Programming Interfaces that allow the customer to write programs to obtain the services of Tivoli Information Management for z/OS.

Contents

Preface

This guide describes the program administration tasks involved in using Tivoli® Information Management for z/OS Version 7.1 (hereafter referred to as Tivoli Information Management for z/OS). This guide complements the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* and describes procedures involved in migrating to this release of Tivoli Information Management for z/OS from previous versions.

This publication describes Tivoli Information Management for z/OS Version 7 Release 1. There may be references in this publication to versions of Tivoli Information Management for z/OS's predecessor products. For example:

- TME 10™ Information/Management Version 1.1
- Information/Management Version 6.3, Version 6.2, Version 6.1
- Tivoli Service Desk for OS/390® Version 1.2

Who Should Read This Guide

This guide is intended for:

- Product planners, who assign personnel to manage Tivoli Information Management for z/OS
- System analysts or programmers, who install Tivoli Information Management for z/OS and provide procedures for using Tivoli Information Management for z/OS
- Database administrators, who maintain databases.

To use the this guide effectively, you must understand how to use applications on your MVS/ESA™ operating system. You need to know how to use Tivoli Information Management for z/OS, including how to create stored response chains (SRCs). Therefore, read the *Tivoli Information Management for z/OS User's Guide* before you attempt to do any of the tasks in this book. To do some of the tasks described in this book, you must also know how to use the Panel Modification Facility (PMF) and Terminal Simulator Panels (TSPs). Refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide* and the *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference* for more information.

Prerequisite and Related Documentation

The library for Tivoli Information Management for z/OS Version 7.1 consists of these publications. For a description of each, see "The Tivoli Information Management for z/OS Library" on page 521.

Tivoli Information Management for z/OS Application Program Interface Guide, SC31-8737-00

Tivoli Information Management for z/OS Client Installation and User's Guide, SC31-8738-00

Tivoli Information Management for z/OS Data Reporting User's Guide, SC31-8739-00

Tivoli Information Management for z/OS Desktop User's Guide, SC31-8740-00

Tivoli Information Management for z/OS Diagnosis Guide, GC31-8741-00

Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications, SC31-8744-00

Tivoli Information Management for z/OS Integration Facility Guide, SC31-8745-00

Tivoli Information Management for z/OS Licensed Program Specification, GC31-8746-00

Tivoli Information Management for z/OS Master Index, Glossary, and Bibliography, SC31-8747-00

Tivoli Information Management for z/OS Messages and Codes, GC31-8748-00

Tivoli Information Management for z/OS Operation and Maintenance Reference, SC31-8749-00

Tivoli Information Management for z/OS Panel Modification Facility Guide, SC31-8750-00

Tivoli Information Management for z/OS Planning and Installation Guide and Reference, GC31-8751-00

Tivoli Information Management for z/OS Problem, Change, and Configuration Management, SC31-8752-00

Tivoli Information Management for z/OS Program Administration Guide and Reference, SC31-8753-00

Tivoli Information Management for z/OS Reference Summary, SC31-8754-00

Tivoli Information Management for z/OS Terminal Simulator Guide and Reference, SC31-8755-00

Tivoli Information Management for z/OS User's Guide , SC31-8756-00

Tivoli Information Management for z/OS World Wide Web Interface Guide, SC31-8757-00

Note: Tivoli is in the process of changing product names. Products referenced in this manual may still be available under their old names (for example, TME 10 Enterprise Console instead of Tivoli Enterprise Console®).

What This Guide Contains

The following chapters provide information about administering Tivoli Information Management for z/OS databases:

- "Defining a User Profile" on page 1 tells you how to define and set up default user profiles.
- "Defining Privilege Classes" on page 15 tells you how to create privilege classes.
- "People Records" on page 63 describes how to create People Records.
- "Customizing the Enhanced Panel Style" on page 79 tells you how to customize the enhanced panel style.
- | "Sending E-mail Messages" on page 95 shows how you can use terminal simulator EXECs (TSXs) to send e-mail messages to people through TCP/IP simple mail transfer protocol (SMTP).
- "Defining User Line Commands" on page 137 tells you how to create user line commands.

- “Defining Command Aliases and Authorization” on page 147 tells you how to customize Tivoli Information Management for z/OS commands and restrict the usage of commands by privilege class.
- “Setting Up Stored Response Chains” on page 153 tells you how to set up Stored Response Chains (SRCs).
- “Using OS/390 Text Search for Improved Problem Resolution or Freeform Text Searches” on page 157 describes how to use OS/390 Text Search with Tivoli Information Management for z/OS to create a solution knowledge base or to enable users to perform freeform text searches.

The following chapters provide information about the Tivoli Information Management for z/OS database and data propagation:

- “Understanding Your Tivoli Information Management for z/OS Database” on page 193 provides an explanation of the Tivoli Information Management for z/OS database.
- “Logical Database Partitioning” on page 203 discusses how you can organize your data into multiple “logical” partitions from a single physical database.
- “Performing Database Administration” on page 221 tells you how to perform database administration tasks using two specific function, DBCLEANUP and CHECK IN.
- “Propagating Tivoli Information Management for z/OS Data” on page 231 discusses data propagation and the Tivoli Information Management for z/OS facilities for propagating data.
- “Propagating to a Tivoli Information Management for z/OS Database” on page 253 tells you how to set up the Automatic Log Save Facility to propagate data to another Tivoli Information Management for z/OS database.
- “Propagating to a DB2 Database” on page 287 tells you how to set up the DB2® Extract Facility to propagate data to a DB2 database.
- “Data Propagation User Exits” on page 323 lists the return and reason codes for the data propagation user exits.

The following chapter provides information about a feature of Tivoli Information Management for z/OS called Remote Data Resources:

- “Remote Data Resources in Tivoli Information Management for z/OS” on page 335 describes using an area (or areas) in the BLX-SP to contain data. Such areas are Remote Data Resources.

The following chapter provides information about archiving Tivoli Information Management for z/OS data:

- “The Tivoli Information Management for z/OS Archiver” on page 337 discusses the Archiver function that enables you to move records from one Tivoli Information Management for z/OS database to another.

The following appendixes provide reference information:

- “Panel Lists” on page 365 lists the entry, display, and inquiry panels that are shipped with Tivoli Information Management for z/OS.
- “Training Records” on page 435 contains sample training records you can use to create a set of training records for beginning users to use in conjunction with the step-by-step

exercises in the *Tivoli Information Management for z/OS User's Guide*. You must create training records before beginning users can work with exercises in the *Tivoli Information Management for z/OS User's Guide*.

- "Your Organization's Procedures" on page 443 provides a template for documenting how your organization performs certain tasks (for example, entering data and numbering records) with the Tivoli Information Management for z/OS program. You can use this template to provide operating procedures for all users in your organization to follow.
- "Using Notification Management" on page 447 tells you how to use the former methods of notification management to inform people of the status of problem, change, or activity records. It also describes how to use REXX TSPs (TSXs) and TCP/IP for notification. In addition, it describes how to create the USERS and the RULES records and tailoring the escalation facility.

How Information Is Presented in This Book

The panels as presented in this book are not meant to be exact replicas of the way a panel might appear on the screen. The information on the panels is correct, but the spacing is not always exact.

With Tivoli Information Management for z/OS, two panel styles are available, the standard panel style and the enhanced panel style.

The style of panel does not affect the data that must be entered from it.

This product is enabled for DBCS support. As a result, this book uses the following terms:

- DBCS (double-byte character set)
- SBCS (single-byte character set).
- Mixed data

The term *mixed data* refers to data strings that can contain only DBCS data, only SBCS data, or any combination of DBCS and SBCS data. SBCS data is the same as EBCDIC data. The term *mixed case data* refers to data strings that can contain uppercase, lowercase, or a combination of uppercase and lowercase SBCS data.

Typeface Conventions

This guide uses several typeface conventions for special terms and actions. These conventions have the following meaning:

Bold Entries that you must use literally, choices, or options that you select appear in **bold**.

Italics Variables and values that you must provide appear in *italics*. New terms also appear in italics.

`Monospace` Code examples appear in `monospace` font.

The panels as presented in this book are not meant to be exact replicas of the way a panel might appear on the screen. The information on the panels is correct, but the spacing is not always exact.

Commands, such as END, CONTROL, RESUME, or DOWN, appear in all capital letters in text. Although not commands, the user responses YES and NO also appear in capital letters.

Contacting Customer Support

For support inside the United States, for this or any other Tivoli product, contact Tivoli Customer Support in one of the following ways:

- Send e-mail to **support@tivoli.com**
- Call 1-800-TIVOLI8
- Navigate our Web site at **http://www.support.tivoli.com**

For support outside the United States, refer to your Customer Support Handbook for phone numbers in your country. The Customer Support Handbook is available online at **http://www.support.tivoli.com**.

When you contact Tivoli Customer Support, be prepared to provide identification information for your company so that support personnel can assist you more readily.

The latest downloads and fixes can be obtained at **http://www.tivoli.com/infoman**.

1

Defining a User Profile

This chapter suggests ways that the program administrator can:
- Create a default profile
- Define a default profile
- Control user profiles
- Set values for the PDF editor

As the program administrator, you can create a *default profile* for all users. Your users can modify the default profile to meet their needs. By creating a default profile, you ensure that certain values are initially entered properly. Use the Panel Modification Facility (PMF) to modify profile panels to control what values users can and cannot alter. Refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide*.

Creating Default Profiles

A user's profile determines some of the ways Tivoli Information Management for z/OS operates by providing default data values during sessions. If the user has not set up a user profile, Tivoli Information Management for z/OS uses one you can set up as the default. You can control both the contents of the default profile and what individual users can define in their own user profiles. You must, of course, establish your controls before any other users sign on to Tivoli Information Management for z/OS.

When Tivoli Information Management for z/OS uses a default profile, it first searches for the member BLG0PROF in the partitioned data set (PDS) allocated to the user's ISPPROF DD statement in Interactive System Productivity Facility (ISPF). BLG0PROF in the ISPPROF DDName concatenation is the user's own ISPF profile for the Tivoli Information Management for z/OS application. If BLG0PROF is not there, ISPF searches for the same member (BLG0PROF) in the PDS allocated to the ISPTLIB DD statement in ISPF. BLG0PROF in the ISPTLIB DDNname concatenation is the Tivoli Information Management for z/OS profile. If member BLG0PROF is found in the data set, the profile is read into the ISPF profile variable pool and Tivoli Information Management for z/OS uses it for the session. If member BLG0PROF is not found, Tivoli Information Management for z/OS builds its own profile using internal default values and places it in the ISPF profile variable pool.

When the user ends the current Tivoli Information Management for z/OS session, ISPF writes the contents of the Tivoli Information Management for z/OS profile variable pool to the user's profile data set as the BLG0PROF member so that it is available for the next session.

Defining a Default Profile

By ensuring that the Tivoli Information Management for z/OS profile member of the ISPTLIB PDS exists, you can put your own default values into effect to override the internally defined defaults. In addition, because the ISPF variable pool contains PF key settings as well, you can set default PF keys for a Tivoli Information Management for z/OS session.

Before creating a default profile for your Tivoli Information Management for z/OS users, consider the following questions:

- What defaults do you want to provide your users?
- Do different groups of users require different defaults, or can one value for a given field serve all users?

To define a default profile for new users to use the first time they sign on to Tivoli Information Management for z/OS, follow these steps:

1. Start Tivoli Information Management for z/OS and select **Profile** from the Primary Options Menu or enter the **PROFILE** command.
2. Update the profile by setting the default values you want to include in the default profile.
3. After you finish, select **Permanent profile end** from the Profile Summary panel, or enter the **END** command to permanently update the profile.
4. Issue the **ISPF KEYS** command and set the PF key values to those you want to include in the default profile.
5. Enter **END** to complete the ISPF KEYS command.
6. Enter **QUIT** to exit the Tivoli Information Management for z/OS session and to write the profile to your profile data set.
7. Copy the BLG0PROF member from your profile data set into ISPF's system profile data set (allocated to the ISPTLIB DD statement).

Controlling User Profiles

You can ensure that users do not change certain values in the default profile you have established, or, when they do change the values, that they enter proper alternatives by one of the following methods:

- Use a startup stored response chain (SRC) to prompt the user for correct responses. See “Prompting the User with a Stored Response Chain” on page 3 for more information.
- Modify the validation process for certain data fields in the default profile. See “Changing Validation of Data-Entry Fields” on page 3 for more information.
- Remove data fields from the panel. See “Deleting Profile Data-Entry Fields” on page 3 for information on deleting data fields.

You can also add your own data fields to the profile. See “Adding User-Profile Variable Fields” on page 4 for information on adding data fields.

You can also use a startup TSP to change user profiles and ensure that they contain the correct values. This startup TSP can also be used to make the same change to every user's profile. BLGTPSET is an example of a TSP that you can use for this purpose. See “Using a Startup TSP to Update Profiles” on page 12 for information on TSP BLGTPSET.

Prompting the User with a Stored Response Chain

For those fields that pertain to the individual user (user's name, department, and telephone number), you can create a startup SRC that selects **Profile** from the Primary Options Menu and automatically prompts the new user for these items. Assuming that you assign all users to privilege classes, and that the users know the names of those privilege classes the first time they sign on to Tivoli Information Management for z/OS, you can also prompt for their invocation privilege class (a session control default).

To ensure that new users are prompted for this information only the first time they sign on to Tivoli Information Management for z/OS, have the startup SRC replace its own name in the Invocation SRC field in the user profile with blanks; or, have the startup SRC update the Invocation SRC field with the name of the SRC that processes each time the user logs on. Also, design the startup SRC to select **Permanent profile end**, which forces the SRC to end.

If you specify this SRC's name in the Invocation SRC field of the default profile, new users signing on to Tivoli Information Management for z/OS for the first time are automatically prompted for this data and their profiles are permanently updated with their responses. See "Setting Up Stored Response Chains" on page 153 for more information.

Changing Validation of Data-Entry Fields

You can also control how the user modifies the default profile by changing the validation process for the data collected in the profile fields. Whenever a user enters data into a data-entry field on the profile panels, Tivoli Information Management for z/OS calls program exit BLG01439 to compare the data to a set of acceptable values. Any value that does not match one of the acceptable values is ignored, and an error message is issued. Refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide* for more information about the BLG01439 program exit.

You can modify the field validations to further limit the user's choice of values within the values that are already defined. For example, you could allow only responses of **command** or **data** for the command detection field and eliminate **prompt** as a response. You cannot, however, change the **prompt** response to **ask**; profile validation does not process this value.

By modifying the validation of certain data fields, you can ensure that the user does not change the data in the default profile to a value that is not appropriate in your organization. However, you do not want to be too restrictive, because you could limit the ease-of-use that the profile function provides.

Refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide* for information about changing validation of data-entry fields.

Deleting Profile Data-Entry Fields

You can delete any field from the profile data-entry panels with PMF. When you eliminate certain fields from the profile panels, users cannot alter the default profile data associated with those deleted fields. This method ensures that all users use profile values appropriate for your organization.

Refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide* for information about modifying fields on panels.

Adding User-Profile Variable Fields

You can define as many user-profile variable fields as you want. A *user-profile variable field* is a field in the user's profile that can have the value in it extracted and used when the user enters the equal sign in a data-entry field. Tivoli Information Management for z/OS has this capability built in for the user's name, department, and phone number. You might have others you want to use.

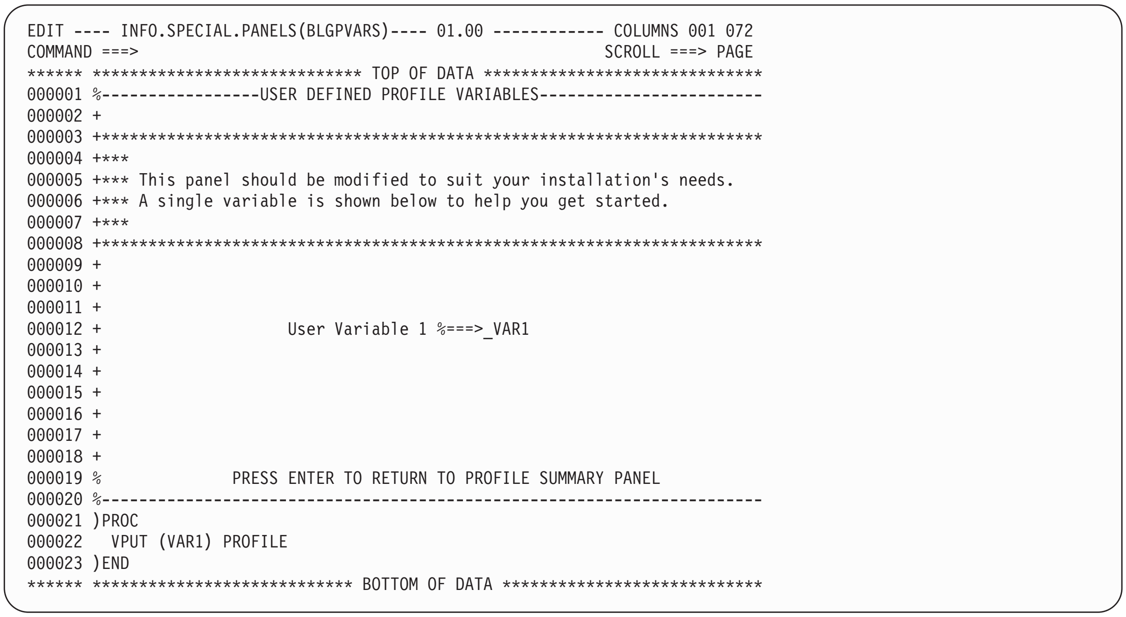
You cannot add user-profile variables to the existing user-profile data-entry panels, but Tivoli Information Management for z/OS provides a special ISPF panel for this purpose. The Profile Summary panel selection **User-defined profile values** enables the user to enter data in the new variables and use them like other profile fields, including using the equal sign function to enter the data on data-entry panels. To define your own user-profile variables, modify the ISPF panel BLGPVARS.

First, find where panel BLGPVARS resides in your system. The Tivoli Information Management for z/OS installation instructions tell you to have this panel reside in a data set that is in your ISPPLIB concatenation. You must update this panel in the way you usually update data sets. Because you cannot update this panel using PMF, this example uses the PDF editor. For more information on updating ISPF panels, refer to *ISPF/PDF Edit and Edit Macros*, *ISPF/PDF Guide and Reference*, and *ISPF Dialog Management Guide and Reference*.

In the following example, the panel is in the INFO.SPECIAL.PANELS data set. Suppose that you want to replace the standard Tivoli Information Management for z/OS field for user's name with one of your own. You also want a field for the user's location in the user profile. The following example shows step-by-step how to replace the user's name field, and how to create a user's location field.

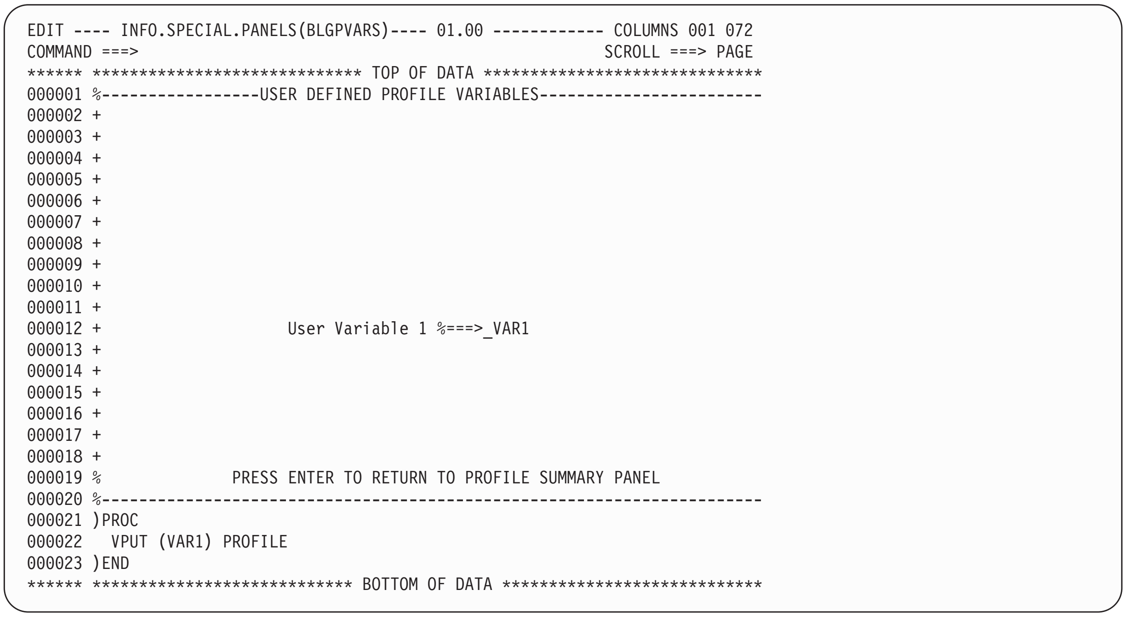
This is how panel BLGPVARS appears on your screen when you are in PDF edit mode.

This is the way the panel is shipped with Tivoli Information Management for z/OS. The underscore character (_) represents the beginning of an entry field. The percent (%) and plus (+) characters represent the beginning of protected fields. Lines 1-20 contain the visible part of the panel. Lines 21-23 contain statements that do panel processing.



```
EDIT ---- INFO.SPECIAL.PANELS(BLGPVARS)---- 01.00 ------------ COLUMNS 001 072
COMMAND ===>                                                  SCROLL ===> PAGE
****** ***************************** TOP OF DATA ******************************
000001 %-----------------USER DEFINED PROFILE VARIABLES------------------------
000002 +
000003 +***********************************************************************
000004 +***
000005 +*** This panel should be modified to suit your installation's needs.
000006 +*** A single variable is shown below to help you get started.
000007 +***
000008 +***********************************************************************
000009 +
000010 +
000011 +
000012 +                    User Variable 1 %===>_VAR1
000013 +
000014 +
000015 +
000016 +
000017 +
000018 +
000019 %              PRESS ENTER TO RETURN TO PROFILE SUMMARY PANEL
000020 %-----------------------------------------------------------------------
000021 )PROC
000022   VPUT (VAR1) PROFILE
000023 )END
****** **************************** BOTTOM OF DATA ****************************
```

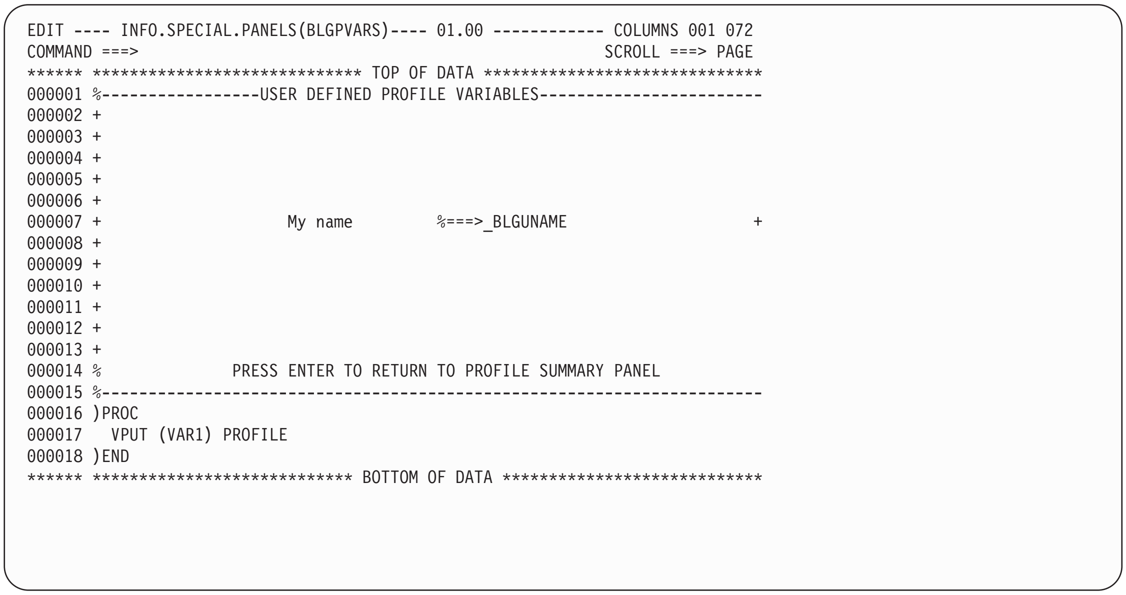
No matter what other changes you want to make to this panel, you probably want to remove the instructions from it. Simply deleting those lines causes the final panel to be shorter. To keep the standard panel length, remove just the instruction text and leave the + attribute character for each line as shown. For the rest of this example, the lines are deleted so that all of the processing lines of the panel can be seen on one screen. Because you probably want to keep the lines, your panel processing lines will spill over to a second editing screen.



```
EDIT ---- INFO.SPECIAL.PANELS(BLGPVARS)---- 01.00 ------------ COLUMNS 001 072
COMMAND ===>                                                  SCROLL ===> PAGE
****** ***************************** TOP OF DATA ******************************
000001 %-----------------USER DEFINED PROFILE VARIABLES------------------------
000002 +
000003 +
000004 +
000005 +
000006 +
000007 +
000008 +
000009 +
000010 +
000011 +
000012 +                    User Variable 1 %===>_VAR1
000013 +
000014 +
000015 +
000016 +
000017 +
000018 +
000019 %              PRESS ENTER TO RETURN TO PROFILE SUMMARY PANEL
000020 %-----------------------------------------------------------------------
000021 )PROC
000022   VPUT (VAR1) PROFILE
000023 )END
****** **************************** BOTTOM OF DATA ****************************
```

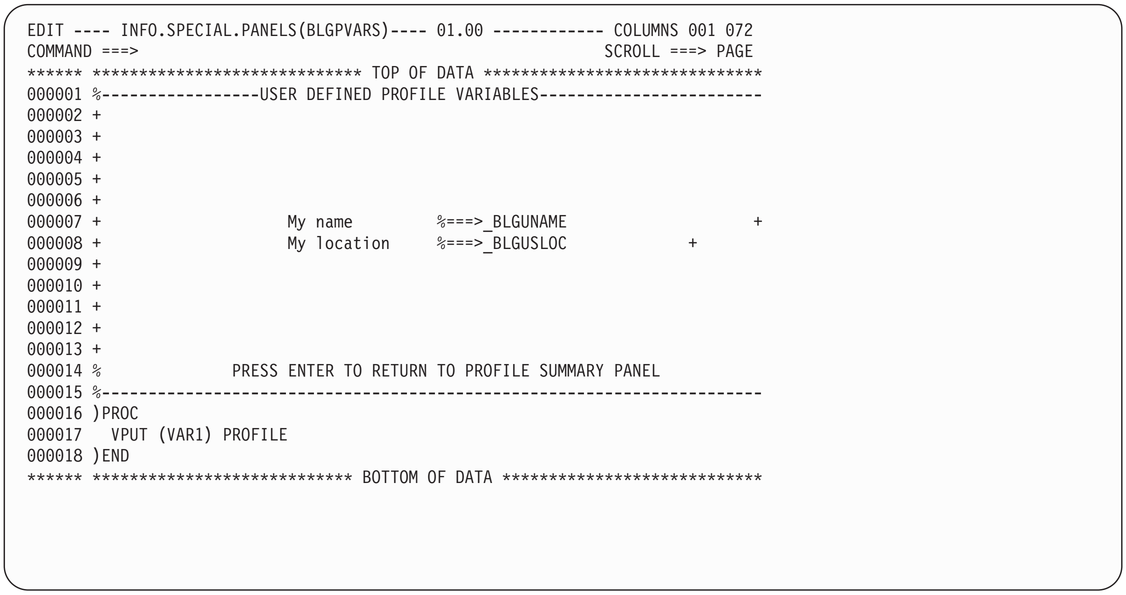
To identify a user-defined field, create an ISPF variable on this panel. Decide on the variable's name and its length in characters. To avoid confusion, give every variable a unique name. Eight-character names offer plenty of chances for name uniqueness. This example uses 8-character names with an arbitrary naming convention that requires that all names start with the characters BLG.

Identify a field for the user's name. Type over the existing information for **User Variable 1**, making sure to retain the attribute characters. Beginning with the character after the underscore, count out spaces equal to the maximum number of characters you want to allow for the length of the name. Put a plus sign to indicate the end of the field as shown. This example allows a length of 25 characters for the name. Change the characters after the underscore to **BLGUNAME**, for the field **My name** with our naming convention prefix of **BLG**. You have just created an ISPF variable name.



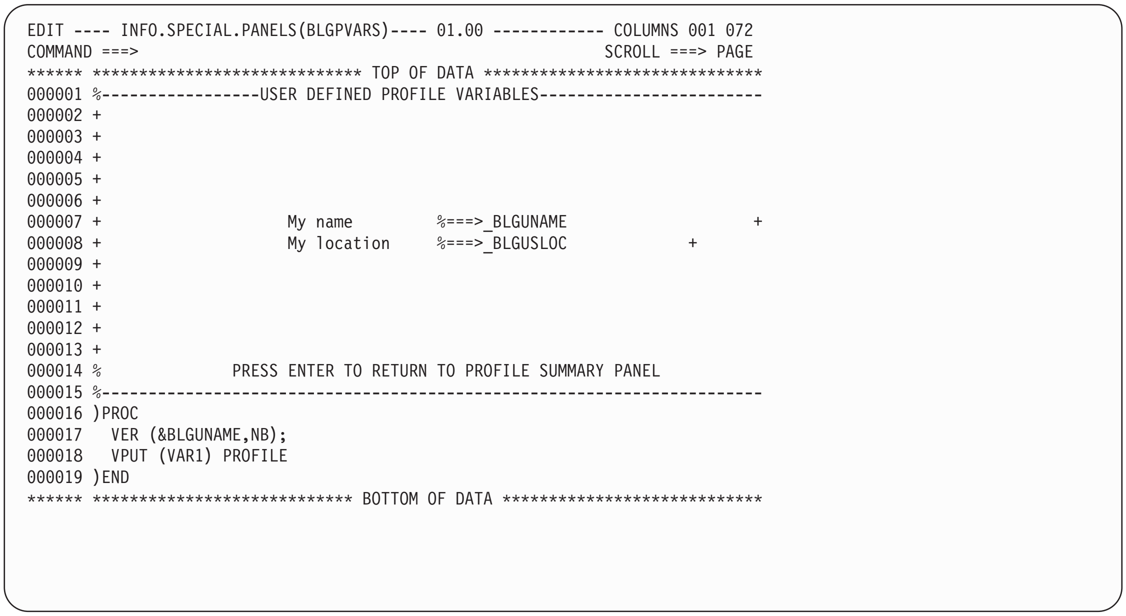
```
EDIT ---- INFO.SPECIAL.PANELS(BLGPVARS)---- 01.00 ------------ COLUMNS 001 072
COMMAND ===>                                                  SCROLL ===> PAGE
****** ****************************** TOP OF DATA *****************************
000001 %-----------------USER DEFINED PROFILE VARIABLES------------------------
000002 +
000003 +
000004 +
000005 +
000006 +
000007 +                    My name         %===>_BLGUNAME                    +
000008 +
000009 +
000010 +
000011 +
000012 +
000013 +
000014 %              PRESS ENTER TO RETURN TO PROFILE SUMMARY PANEL
000015 %-----------------------------------------------------------------------
000016 )PROC
000017   VPUT (VAR1) PROFILE
000018 )END
****** **************************** BOTTOM OF DATA ****************************
```

Next identify a field for the user's location. Put this information into line 8, rather than inserting a new line. (Because you did not delete the entire instruction line the way this example did, your line number for this step should be 13.) This keeps the current panel size. Call the ISPF variable for this field BLGUSLOC, and give the field a length of 15.



```
EDIT ---- INFO.SPECIAL.PANELS(BLGPVARS)---- 01.00 ------------ COLUMNS 001 072
COMMAND ===>                                                  SCROLL ===> PAGE
****** ***************************** TOP OF DATA ******************************
000001 %-----------------USER DEFINED PROFILE VARIABLES------------------------
000002 +
000003 +
000004 +
000005 +
000006 +
000007 +                    My name         %===>_BLGUNAME                    +
000008 +                    My location     %===>_BLGUSLOC             +
000009 +
000010 +
000011 +
000012 +
000013 +
000014 %              PRESS ENTER TO RETURN TO PROFILE SUMMARY PANEL
000015 %-----------------------------------------------------------------------
000016 )PROC
000017   VPUT (VAR1) PROFILE
000018 )END
****** **************************** BOTTOM OF DATA ****************************
```

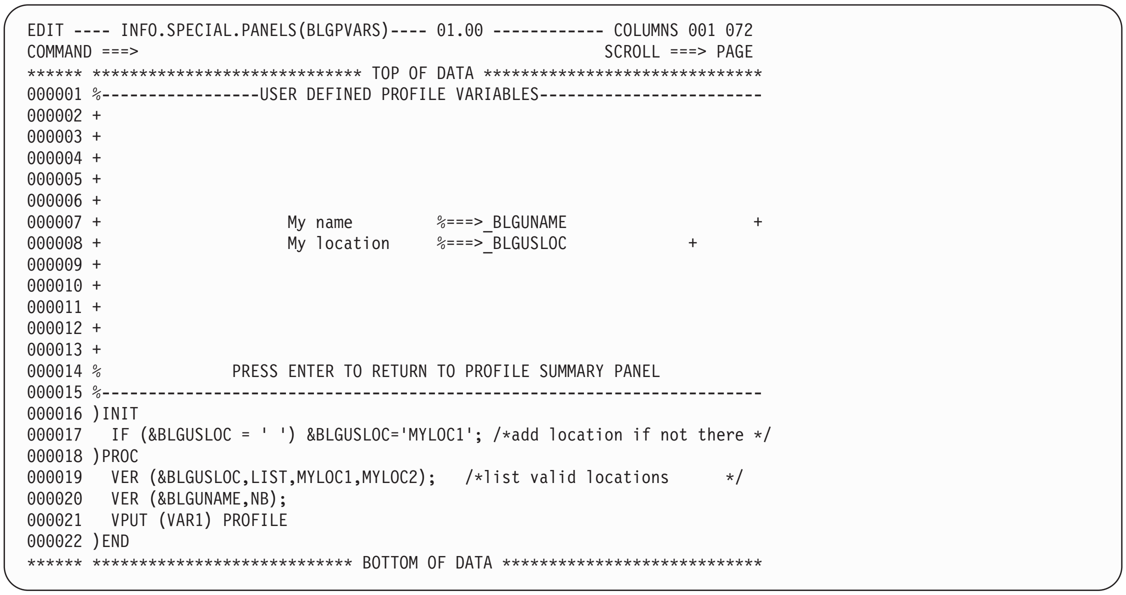
You can make the profile fields you define required fields. This way you can ensure that certain information is always in the user's profile. To make the fields required, change the processing portion of panel BLGPVARS as shown here. Notice that you need to add a "verify" (VER) line (line number 17). Include the **NB** (Not Blank) parameter after the ISPF variable name to make the **My name** field a required field.



```
EDIT ---- INFO.SPECIAL.PANELS(BLGPVARS)---- 01.00 ------------ COLUMNS 001 072
COMMAND ===>                                                  SCROLL ===> PAGE
****** ****************************** TOP OF DATA ******************************
000001 %-----------------USER DEFINED PROFILE VARIABLES------------------------
000002 +
000003 +
000004 +
000005 +
000006 +
000007 +                    My name         %===>_BLGUNAME                    +
000008 +                    My location     %===>_BLGUSLOC             +
000009 +
000010 +
000011 +
000012 +
000013 +
000014 %              PRESS ENTER TO RETURN TO PROFILE SUMMARY PANEL
000015 %-----------------------------------------------------------------------
000016 )PROC
000017   VER (&BLGUNAME,NB);
000018   VPUT (VAR1) PROFILE
000019 )END
****** **************************** BOTTOM OF DATA ****************************
```

You can also prime the profile fields you define by changing the processing portion of the panel. This is useful when you create a field that is almost always going to contain the same information, for example, the location of a problem reporter.

To prime a profile field, add an **INIT** section to the panel processing. This example shows the **My location** field (BLGUSLOC) primed with **myloc1**. (See lines 16 and 17.) Also, you decide that only two entries are valid for the **My location** field, **myloc1** and **myloc2**. You can make the processing portion of the panel verify that this field accepts only those two entries. Use a **VER** statement with a **LIST** parameter as shown in line 19.

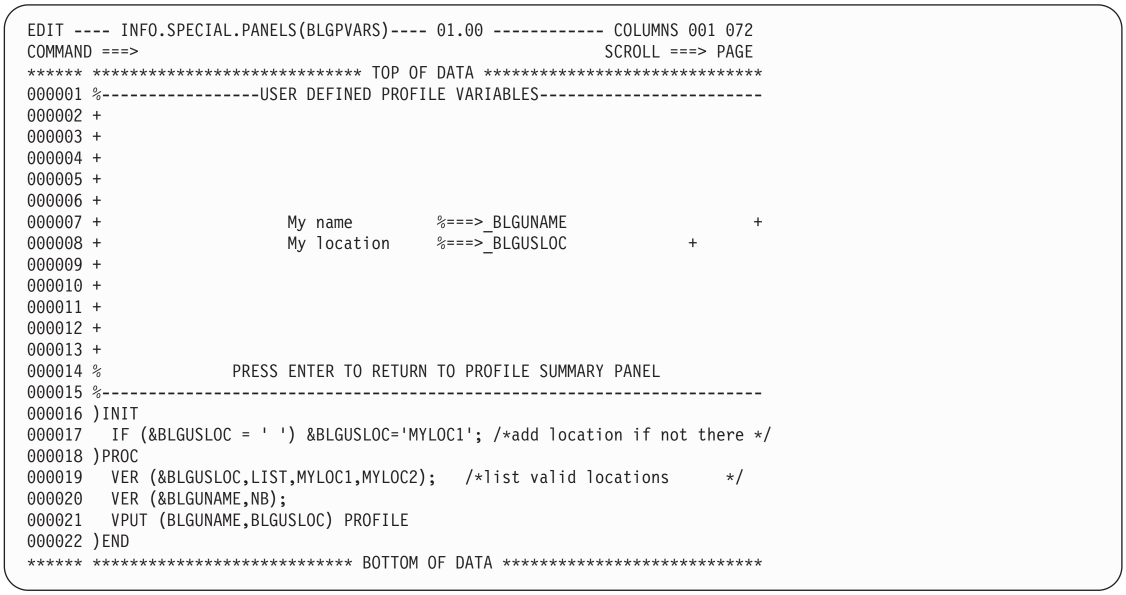


```
EDIT ---- INFO.SPECIAL.PANELS(BLGPVARS)---- 01.00 ------------ COLUMNS 001 072
COMMAND ===>                                                  SCROLL ===> PAGE
****** ***************************** TOP OF DATA ******************************
000001 %-----------------USER DEFINED PROFILE VARIABLES------------------------
000002 +
000003 +
000004 +
000005 +
000006 +
000007 +                    My name         %===>_BLGUNAME                    +
000008 +                    My location     %===>_BLGUSLOC             +
000009 +
000010 +
000011 +
000012 +
000013 +
000014 %              PRESS ENTER TO RETURN TO PROFILE SUMMARY PANEL
000015 %-----------------------------------------------------------------------
000016 )INIT
000017   IF (&BLGUSLOC = ' ') &BLGUSLOC='MYLOC1'; /*add location if not there */
000018 )PROC
000019   VER (&BLGUSLOC,LIST,MYLOC1,MYLOC2);   /*list valid locations      */
000020   VER (&BLGUNAME,NB);
000021   VPUT (VAR1) PROFILE
000022 )END
****** **************************** BOTTOM OF DATA ****************************
```

If you want to make the **My location** field a required field, add the **NB** parameter to line 19 as well:

```
VER (&BLGUSLOC,NB,LIST,MYLOC1,MYLOC2);
```

Finally, save the information. Change the processing portion of the panel to save the new variables. Change line 21 to write the 2 variables you identified, **BLGUNAME** and **BLGUSLOC**, to the user's profile.

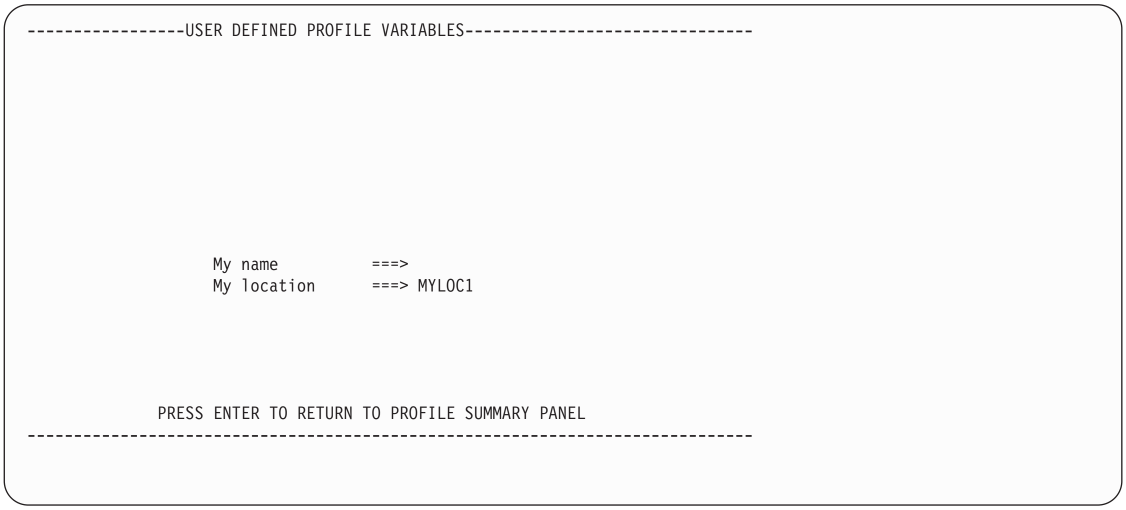


```
EDIT ---- INFO.SPECIAL.PANELS(BLGPVARS)---- 01.00 ------------ COLUMNS 001 072
COMMAND ===>                                                  SCROLL ===> PAGE
****** ***************************** TOP OF DATA *****************************
000001 %-----------------USER DEFINED PROFILE VARIABLES------------------------
000002 +
000003 +
000004 +
000005 +
000006 +
000007 +                   My name        %===>_BLGUNAME                   +
000008 +                   My location    %===>_BLGUSLOC            +
000009 +
000010 +
000011 +
000012 +
000013 +
000014 %              PRESS ENTER TO RETURN TO PROFILE SUMMARY PANEL
000015 %-----------------------------------------------------------------------
000016 )INIT
000017   IF (&BLGUSLOC = ' ') &BLGUSLOC='MYLOC1'; /*add location if not there */
000018 )PROC
000019   VER (&BLGUSLOC,LIST,MYLOC1,MYLOC2);   /*list valid locations      */
000020   VER (&BLGUNAME,NB);
000021   VPUT (BLGUNAME,BLGUSLOC) PROFILE
000022 )END
****** **************************** BOTTOM OF DATA ****************************
```

To use the new profile variables in Tivoli Information Management for z/OS, you must add a p-word to the dictionary for each of them. For this example, add **My location** field, which is identified by variable BLGUSLOC, and **My name** field, which is identified by variable **BLGUNAME**. You must add p-words to the dictionary with validation patterns of **=BLGUSLOC** and **=BLGUNAME**. These p-words must contain the prefix that is associated with the field each new user-defined profile variable is used for. For example, **LOCC/** for location, and one entry for each name prefix, such as **PERS/**, **PERA/**, **PERC/**, P**ERR/**, and so on.

When the prefix index is added as the first item in the list of validation patterns for an assisted-entry panel, the user can enter an equal sign into that field and retrieve the information that is stored for that field in the user's profile. You must include the new variables on the panels that they are to be used on. Add their prefix indexes to the list of valid values for each panel that you want to use the equal sign on. Refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide* for information about adding entries to the dictionary and modifying fields on panels.

Now, when the user chooses option 12 from the Profile Summary panel, the user sees a panel with your user-defined profile variables. The blank lines that were deleted during the example were replaced. The user can enter data here to include it in his user profile. Notice that **My location** field (**BLGUSLOC**) is primed with **myloc1**; however, the user can change it by entering new data in the field. The only other data that is accepted, however, is **myloc2**.



```
-----------------USER DEFINED PROFILE VARIABLES-------------------------------









                    My name          ===>
                    My location      ===> MYLOC1





              PRESS ENTER TO RETURN TO PROFILE SUMMARY PANEL
------------------------------------------------------------------------------
```

Using a Startup TSP to Update Profiles

If you want to make the same change to the profiles of all of your users, or ensure that their profiles contain the correct values, you can use a startup TSP. An example of a startup TSP that makes the same change to the profiles of all of your users is BLGTPSET. As this TSP is supplied, it updates the **Bypass copyright?** profile field to **YES**. The next time the user starts Tivoli Information Management for z/OS, the proprietary information panel is bypassed.

You can run BLGTPSET as a startup TSP in either of the following ways:

- Use the **TSP** parameter on the ISPSTART command that starts Tivoli Information Management for z/OS. For example:

  ```
  ISPSTART PGM(BLGINIT) PARM(SESS(00) TSP(BLGTPSET))
  ```

- Use the **IRC** parameter on the ISPSTART command that starts Tivoli Information Management for z/OS. For example:

  ```
  ISPSTART PGM(BLGINIT) PARM(SESS(00) IRC(RUN BLGTPSET))
  ```

If you want to make other changes to all the profiles, you can use PMF to modify the TSP.

The logic of the TSP is as follows:

1. Call a user exit to determine the level of the user’s profile.
2. Set the **TSCAUFLD** field to the level of the user’s profile.
3. Check whether the user’s level matches the TSP level. As supplied, the level of the TSP is **level1**.
4. If the two levels do not match, process the **ADDDATA**, **PROCESS**, and **USEREXIT** lines to update the profile (in the shipped example, **BYPASS COPYRIGHT?** is set to YES) and also store the TSP level in the user profile. The last **ADDDAT**A quits out of Tivoli Information Management for z/OS to permanently save the updated profile.

5. If the two levels match, **EXIT**.

To modify this TSP:
1. Update the **ADDDATA** line with the changes you want to make to your users' profiles.
2. Change **level1** to **level2**.
3. Copy the updated TSP from a write panel data set to a read panel data set.

For more information on modifying TSPs, refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide* and the *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference*.

Setting Values for the PDF Editor

If your users want to use the PDF editor for text processing, you can identify an invocation macro for them. When your user sets his user profile to use the PDF editor, it can automatically trigger Tivoli Information Management for z/OS to run an invocation macro that you predetermine. You can use a single macro for all types of records and text, or you can specify a different macro for each type of text in each type of record. See "Specifying an Invocation Macro for the PDF Editor" for more information.

You can also identify an ISPF panel for them to use with the PDF editor. This panel can display text type, record number, and other Tivoli Information Management for z/OS data that you predetermine. You can use a single ISPF panel for all types of records and text, or you can specify a different panel for each type of text in each type of record. For information on changing the name of the ISPF panel or changing the ISPF variables that are set when using the PDF editor for freeform text, refer to "Changing Fields Displayed on Freeform Text Panels" in the checklists for complex modifications in the *Tivoli Information Management for z/OS Panel Modification Facility Guide*. For information on how to change the ISPF panel to display different ISPF variables, refer to the *ISPF Dialog Developer's Guide and Reference*

Note: This ISPF panel is used only with the PDF editor. It is ignored if the user uses the Tivoli Information Management for z/OS editor or a workstation editor.

Store your ISPF panels in a data set that is in the ISPPLIB concatenation.

The default logical record length (LRECL) for the PDF editor is 132. You can customize this value by setting a variable before you call the invocation macro. See "Customizing the LRECL for PDF" on page 14 for more information.

No special considerations need to be made if your users want to use a workstation editor.

Specifying an Invocation Macro for the PDF Editor

The way Tivoli Information Management for z/OS determines which, if any, invocation macro to use is by the setting of an ISPF variable called BLGITEXT. You can set this variable using program exit BLG01376. Refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide* for information on how this exit works.

To get you started, you can use PMF to look at panel BTN1S026. This panel calls program exit BLG01376 to set variable BLGITEXT to a value of BLGISMAC. BLGISMAC is the name of the invocation macro shipped with Tivoli Information Management for z/OS.

After setting BLGITEXT, the panel calls the freeform text editor (program exit BLG01208).

Finally, the panel uses program exit BLG01376 to blank out the value of BLGITEXT. This ensures that this macro is not used for other record or text types.

You can call program exit BLG01376 anywhere in your panel flows, as long as it is on a control panel. If you want a different macro for each record type, you can call this program exit early in your record create or update panel flow and, optionally, blank it out at record file time.

If you want a different macro for each type of text in a record, you must insert a control panel (if one is not already there) in the panel flow after the user selects the text type. You must move the program exit from the selection panel to the new control panel. Again, panel BTN1S026 can give you an idea of how to set up the control line to call the editor from a control panel.

Note: The value of BLGITEXT is only used with the PDF editor. It is ignored if the user uses the Tivoli Information Management for z/OS editor.

Store your macros in a data set that is listed in the SYSPROC DD statement concatenation.

Customizing the LRECL for PDF

Tivoli Information Management for z/OS determines the LRECL for the temporary data set for the PDF editor from the setting of ISPF variable BLGIRECL. You can set this variable using program exit BLG01376 to set this variable before it calls the invocation macro for the PDF editor. Refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide* for information on how this exit works.

If BLGIRECL is not defined, not numeric, or not in the range of 1 through 244, an LRECL of 132 is used. If you assign BLGIRECL a value greater than 132, update the text display table panels to specify the same value so the text can be displayed in display mode. Also, if you assign BLGIRECL a value greater than 132, and you use an application running with an application program interface to retrieve records with the data set option for freeform text, only 132 characters of text per line is shown.

Note: The value of BLGIRECL is only used with the PDF editor. It is ignored if the user uses the Tivoli Information Management for z/OS editor.

Preserving the Audit Trail for Freeform Text

The audit trail for freeform text cannot be preserved if an entire record is edited using the ISPF/PDF editor or a workstation editor. If you update the freeform text in an existing record using either the ISPF/PDF or a workstation editor, the text control data fields in the record are lost when you file it. Only the current control information is filed with the record.

As the program administrator, you can control the use of the ISPF/PDF editor and/or workstation editor by using the TEXTAUD keyword in the BLGPARMS session-parameters member macro. If TEXTAUD is specified as **YES**, panel BLG00105 appears when a user attempts to edit freeform text with the ISPF/PDF editor or workstation editor. This panel requires the user to select the type of edit (add or update) to perform on the text.

TEXTAUD also controls whether an API application can replace or delete existing freeform text. If **TEXTAUD=YES** is specified in the BLGPARMS session-parameters member macro, an application cannot replace or delete existing freeform text.

2

Defining Privilege Classes

This chapter describes privilege classes and suggests how to set up privilege classes to authorize users in your organization to perform certain tasks. It includes a series of scenarios that illustrate managing privilege classes.

Assigning Privilege Classes

A privilege class authorizes a user to perform specific tasks and access certain facilities. Your organization does not have to use privilege classes, but it is recommended that you create at least a master privilege class and one other class. The first privilege class that you create is given master authority. If you do not create a privilege class, users have authority to perform every task, including working with SRCs and privilege class records.

You can define a privilege class for either an individual or a group of users. Because it is often desirable for several users to have access to the same data (if they have similar or complementary job responsibilities), you can assign them to the same privilege class. For example, system operators and system programmers can share the same privilege class.

Optionally, you can authorize the use of Tivoli Information Management for z/OS data through privilege classes. A Tivoli Information Management for z/OS privilege class record identifies the privilege class to which a specific individual or group of individuals belongs. A user can perform only the functions permitted by that privilege class, such as accessing certain types of records in a database or making changes to certain types of records.

Each member of a privilege class is identified by an MVS™ logon identifier (user ID). Up to 24 user IDs can be associated with each privilege class; however, by using the panel modification facility (PMF), you can modify the panels to accept a larger number of user IDs. While it is possible for any user to log on to Tivoli Information Management for z/OS, a user who is not a member of a privilege class is limited to displaying the overview, profile, and SRC data.

Categories of Users

Consider creating privilege classes that complement the task structure of your data processing organization. Privilege classes can identify the individuals or groups that represent:

- Functional units in the data processing organization
- Change approvers
- Change reviewers
- Interested parties to a problem
- Tivoli Information Management for z/OS administrators

The number and role of privilege classes depend on the size and structure of your organization. The privilege class for functional units can correspond directly to your data processing organization, or they can include selected groups and individuals. In either case, consider problem reporters, change requesters, coordinators, and assignees. Each group or individual that has change approval or reviewer responsibility should be identified in a privilege class.

Master Privilege Class

As a program administrator, you need universal authority. Therefore, you must be a member of the master privilege class. That is the first privilege class record you create in the Tivoli Information Management for z/OS database. The master privilege class record should contain the user IDs of those individuals assigned as program administrators. (You can add or remove user IDs as required.) As a member of the master privilege class, you can use Tivoli Information Management for z/OS's full capabilities; you are automatically given all authorities for all record types, can use PMF to modify panels, and can perform certain error recovery functions.

You can also process any SRC regardless of whether the SRC lists the master privilege class as a class that can run the SRC. See Figure 2 on page 19 for lists of all the authorities.

You create the master privilege class record in the same way that you subsequently create other privilege classes. However, when you create the master privilege class, you cannot use the **Transfer-to class** field; you do not need to assign authority because you automatically have *all* authorities.

Note: The first privilege class that you enter into the database automatically gets all the authorities for Tivoli Information Management for z/OS. You are not required to name that privilege class MASTER, but it is the “master” privilege class. If you use PMF to create new authorities, the master privilege class gets them automatically.

Once a master privilege class exists, all users must be running under a privilege class to perform a function that is governed by authority. It is recommended that you create at least the master privilege class and one other class. Restrict the master privilege class to a few selected individuals, and assign the remaining users to other privilege classes.

Privilege Class Structure

The first consideration in creating privilege classes is the current or planned functional organization of your data processing installation. Because a privilege class can contain one or more user IDs and a user can be a member of one or more privilege classes, the key to deciding who belongs in what privilege class is to identify the tasks they need to perform.

The structure of your installation's privilege classes should provide each user with the authority needed to perform the assigned system management activities. The two class structures are individual and group.

Individual Privilege Classes

Individual privilege classes identify users with unique responsibilities. For instance, a person responsible for resolving problems would be identified in a privilege class with the authority to create, update, assign, close, display, and delete problem records. The privilege class name could be identified in the **Assignee name** or **Transfer-to class** field in all problem records they would work with.

Group Privilege Classes

Group privilege classes identify users with shared responsibility. If, for instance, you have a group of people who manage problems, their user IDs should be identified in one privilege class that gives all of them the appropriate authorities for problem records.

At some organizations, problems are assigned to a department when they first occur. An owner is not specified at that time. Later, a member of the department selects the problem for solving and becomes the assignee. If you use this method, you can set up Tivoli Information Management for z/OS in either of the following ways:

- At the time a record is created, the user enters the name of a privilege class in the **Assignee name** field. Later, a user who is a member of the assigned privilege class searchesfor all records assigned to the group. On finding a problem to solve, that user either:
  - Puts the user's name in the **Assignee name** field

    or
  - Assumes ownership by putting the user's individual privilege class name in the **Transfer-to class** field and filing the record. That user is then the only person(outside of those users defined in the master privilege class) who can update the record.
- To limit the number of users who can update the record, the user who creates the problem record names the appropriate group privilege class in the **Transfer-to class** field of the record.

  Then, a member of the group uses the group's privilege class to sign on to Tivoli Information Management for z/OS and search for problems that are transferred to the group. When that user finds a problem to solve, he updates and transfers that problem to his individual privilege class by specifying it in the **Transfer-to class** field, and files the record. Until he again updates the record and files it using his individual privilege class, he, any other member of the group privilege class, and any member of the master privilege class can update the record. However, once he updates the record and files it using his individual privilege class, he becomes the owner and is the only person (outside of those users defined in the master privilege class) allowed to update the record.

Whenever you create a group privilege class, identify a group contact. The contact serves as the focal point for questions and issues about the privilege class's responsibilities for particular records and program management tasks. The contact can also communicate information about a record or task to other members of the privilege class. Set up the selection criteria for contacts so that any user with the authority to create privilege classes can assign an appropriate contact.

The privilege class record contains fields to identify the contact's name, phone number, and department. For a privilege class containing only one user ID, use these fields to supply information about that user.

Example of Organization/Privilege Class Structure

Figure 1 shows a privilege class structure for a data processing organization. In this organization, the Installation Controls department has the master privilege class. The Operations department uses group privilege classes, except for SSUPER, which is the system supervisor's individual privilege class. The two programming departments each have a group class as well as two individual classes for members of the department. Table 4 on page 25 shows the authority assignments for this problem management organization.

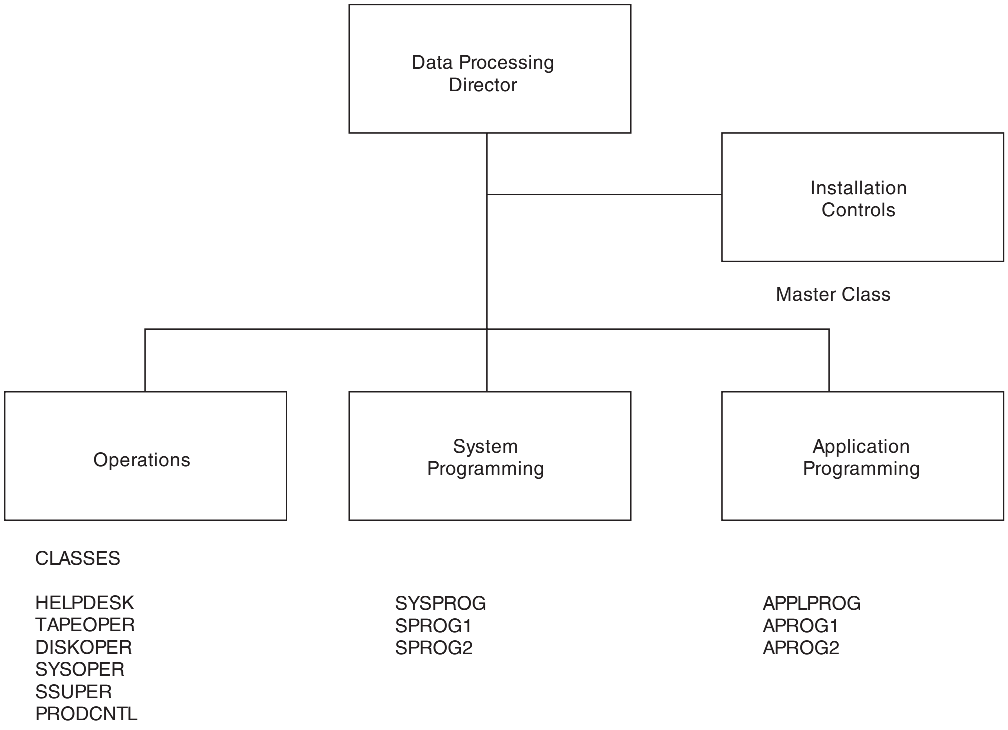


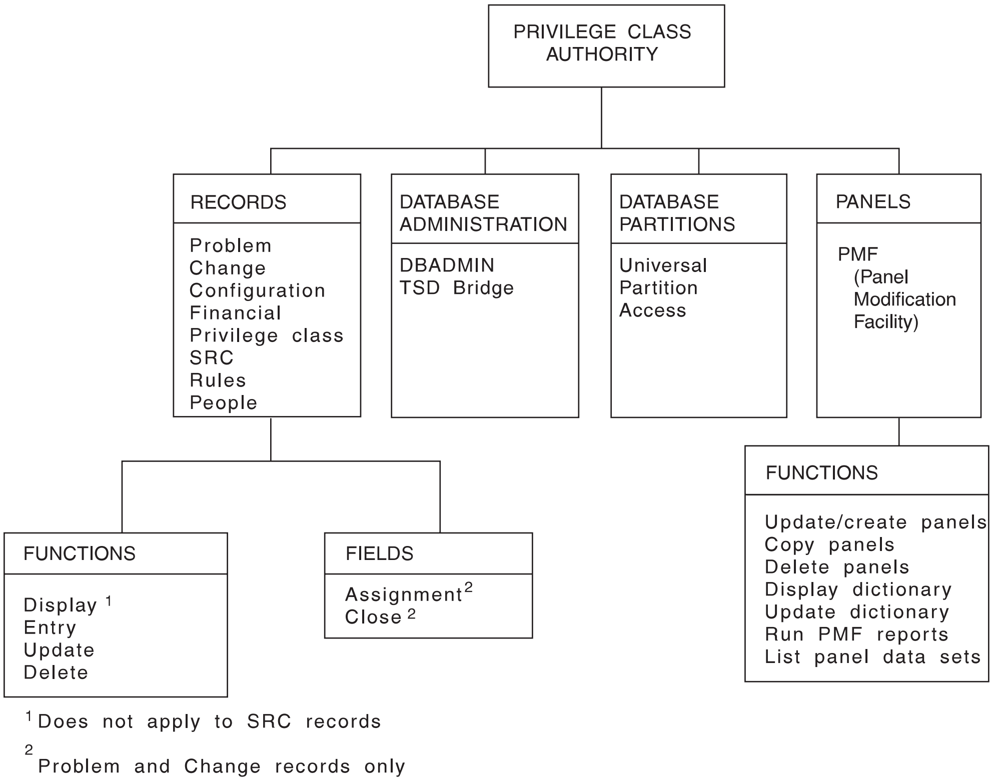
Figure 1. Privilege Class Structure for an Organization

Understanding Authorities

Tivoli Information Management for z/OS uses an authorization system to control the creation, update, display, and deletion of database records. The privilege classes you create are given varying amounts of authority. You can combine the authorities of these assigned privilege classes to:

- Perform a particular function for a given record type, such as create problem records
- Restrict the use of certain selections, input fields, or responses to input fields (such as the **Assignee name** field)
- Create and modify panels, dictionary display and update, and PMF reports

Figure 2 shows all the function- and field-level authorities for records and panels.



[1] Does not apply to SRC records

[2] Problem and Change records only

Figure 2. Function and Field-Level Authorities

Some of the record categories shown in Figure 2 include more than one record type, such as:

- The problem record category includes both problem and call records. Problem record display authority includes solution record display authority.
- The change record category includes both change and activity records.
- The configuration record category includes hardware and software component and feature records and subcomponent records, as well as data center, system, service organization, and Tivoli Inventory records.
- The financial record category includes both hardware and software financial records.
- Each remaining record category (privilege class, SRC, rules, and people) involves only one record type.

When you create a privilege class, specify the types of authority it is to have. You can give the new privilege class the same or fewer authorities than the privilege class under which you are running, but you can never give it more.

Function Level Authorities

The function level authorities (entry, display, update, and delete) allow a user to perform specific functions.

Display Authority

To display, print, or report on any record type other than SRC records, a user must be running under a privilege class that contains display authority for that particular record type. When a user selects **Class** on the Primary Options Menu, any privilege classes that contain the user's own logon identifier can be viewed.

Entry Authority

Entry authority is required to create a new database record. Entry authority also allows the user to copy a record. A privilege class cannot be authorized to create a particular record type unless it can also display that record type.

Update Authority

Update authority allows changes to a database record. A user must be running under a privilege class that has update authority for a given record type before he can change that type of record. A privilege class cannot be authorized to update a particular record type unless it can also display that record type.

This authority is also needed to transfer record ownership. For example, a user in the HELPDESK privilege class creates a record and transfers it to the SYSOPER privilege class. If the SYSOPER privilege class does not have update authority, the record cannot be transferred to that privilege class.

If a record is not owned, the user must be running under a privilege class that has update authority for that record type to update it.

If a record is owned, the user must be running under a privilege class that has update authority for that record type, and that privilege class must own the record or be the transfer-to privilege class before the user can update the record. No other privilege class (except the MASTER privilege class) can update an owned record. See "Deciding Whether to Use Record Ownership" on page 26 for an explanation of record ownership.

Delete Authority

Delete authority is required to delete a record from the database. A privilege class cannot be authorized to delete a particular record type unless it can also display that record type. A user cannot delete a record if any other records reference that record. For example, one of the fields in a problem record is **Original problem number**. If record number 3 in your database has **2** in that field, it might mean that problem 2 was not fixed correctly, and problem 3 resulted from the incorrect fix. You cannot delete problem record 2, because problem record 3 makes a reference to it.

When you delete a parent record of a parent/child relationship, the child records are also deleted.

If a record is not owned, the user must be running under a privilege class that has delete authority for that record type in order to delete the record.

If a record is owned, all of the following conditions must exist before the user can delete it:

- The user must be running under a privilege class that has delete authority for that record type.
- The privilege class in which the user is running must own the record.
- The record cannot be referenced by any other record.

No other privilege class (except the MASTER privilege class) can delete that record.

Field Level Authorities

The assignment and close authorities for problem, change, and activity records designate certain fields that can be completed only by users with the appropriate authority. Using PMF, you can modify these authorities or include additional authorities for other fields or selections. Field level authority is controlled through authorization codes for fields and privilege classes that permit their usage.

Assignment Authority

Assignment authority is required to enter assignment information for problem or change records. A privilege class cannot be given assignment authority unless it has display authority and either entry or update authority for the record type.

A user must have assignment authority for a given record type before any assignment data can be entered in the fields listed in Table 1.

Table 1. Problem and Change Record Fields Requiring Assignment Authority

Problem Records	Change Records
Assignee name Assignee department Assignee phone Date assigned Time assigned Assignment status Assignment number	Assignee name Assignee department Assignee phone Date assigned Time assigned

Even though a privilege class can own and have entry or update authority for a particular record, its members can enter assignment data only if the privilege class has assignment authority.

Close Authority

Close authority is required to enter close information for problem or change records. A privilege class cannot be given close authority unless it has display authority, and either entry or update authority for the record type.

A user must have close authority for a given record type before any close data can be entered in the fields shown in Table 2.

Table 2. Problem and Change Record Fields Requiring Close Authority

Problem Records	Change Records
Resolved by Resolver department Resolver phone Resolver class Date closed Time closed Cause code	Closed by Closer department Closer phone Closer class Completion date Completion time Completion code

Even though a privilege class can own and have entry or update authority for a particular record, its members can enter close data only if the privilege class has close authority.

PMF Authorities

Panel modification requires PMF authority. A user must be running under a privilege class that has the associated PMF authority to create, update, copy, or delete panels; to update or display the dictionary data set; to print PMF reports; or to list panel data sets.

The PMF authorities are selection-level authorities; they correspond to the options on the PMF entry panel.

It is recommended that you restrict PMF authorities to individuals who require them. Keep in mind, however, that in order to use PMF efficiently, a user needs all authorities.

Update Authority

Panel update authority is required to create or update panels. With panel update authority, a user can make external or internal panel modifications and use the **Panel set** selection to create a new set of panels.

Panel Copy Authority

Panel copy authority allows a user to duplicate a panel on the same or a different panel data set. (If you duplicate a panel on the same data set, you must rename it.) To modify a copied panel, the user must be running under a privilege class that has panel update authority.

Copy authority is required to move updated panels from the write panel data set to a read panel data set to make them available for use.

Panel Delete Authority

Panel delete authority is required to delete a panel from a panel data set.

Dictionary Display Authority

Dictionary display authority allows a user to display prefix and s-word information in the dictionary data set. With this authority, more than one user can display the dictionary, even while it is being updated.

Dictionary Update Authority

Dictionary update authority allows a user to display, update, add, and delete prefix and s-word information in the dictionary data set. A maximum of two users can update the dictionary at the same time: one user is allowed to update s-word information and the other user is allowed to update prefix information.

PMF Reports Authority

PMF reports authority allows a user to run PMF reports. The PMF reports provide panel and dictionary information that the user needs to plan panel modifications.

Panel List Authority

Panel list authority allows a user to display the list of panel data set names and data set definition labels referenced by the current session-parameters member. With panel list authority, a user can look at the panels contained in the panel data set and perform any of that user’s authorized PMF functions on the panels listed. In order to have panel list authority, a privilege class record must also have all the other PMF authorities.

Database Administration Authority

Database administration (DBADMIN) is a Tivoli Information Management for z/OS System application authority for the Tivoli Information Management for z/OS program administrator. Having DBADMIN authority enables the program administrator to use the following functions in Tivoli Information Management for z/OS:

ALIAS Record
The ALIAS record is used to define alias names for TSPs and TSXs. You must have DBADMIN authority to create, change, or delete this record, but you do not need this authority to display or print the ALIAS record.

CHECK IN
The CHECK IN function finds records that are marked *checked out* by an application running with the application program interface and checks the records back in. Use this function only if the application ended without checking the records in.

COMMAND Record
The COMMAND record is used to define aliases for commands and to restrict usage of commands. You must have DBADMIN authority to create, change, or delete this record, but you do not need this authority to display or print the COMMAND record.

Data Attribute Records
Data attribute records are used to define and store your Tivoli Information Management for z/OS data model using records rather than panels. You must have DBADMIN authority to create, update, and delete these records.

Data View Records
Data view records are used to define and store your Tivoli Information Management for z/OS data model using records rather than panels. You must have DBADMIN authority to create, update, and delete these records.

DATETIME Records
DATETIME records are used to define relationships between date fields and their related time fields for universal time processing. You must have DBADMIN authority to create, update, or delete these records, but you do not need this authority to display or print the DATETIME records.

DBCLEANUP
DBCLEANUP is a database administration function that cognizes and files any records that are marked busy or deleted as a result of a system outage or ABEND.

INDEX Records
Index records are used to define OS/390 Text Search indexes and to store the information which enables you to index your freeform text data so it can be searched. You must have DBADMIN authority to create, update, and delete these records.

INVENTORY Records
Inventory records contain data extracted from the Tivoli Inventory application (the Tivoli Inventory application is described in *Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications*). You must have DBADMIN authority to delete inventory records.

LOGSAVE Records
LOGSAVE records are used by the Automatic Log Save Facility, which enables you

to propagate Tivoli Information Management for z/OS data to another Tivoli Information Management for z/OS database. You must have DBADMIN authority to create, change, or delete a LOGSAVE record, but you do not need this authority to display and print a LOGSAVE record.

TIMEZONE Records

TIMEZONE records are used to define the time zone symbols, the time difference from the Universal Time zone, and the Daylight Saving Time rule to use for each time zone. They are also used to define the rules for when Daylight Saving Time starts and stops for each time zone. The TIMEZONE record is used when Universal Time processing is enabled in the session-parameters member. You must have DBADMIN authority to create, update, or delete this record, but you do not need this authority to display or print the TIMEZONE record.

Validation Records

Validation records are used in conjunction with assisted-entry panels when you want to present users with a list of valid values to choose from. You must have DBADMIN authority to work with validation records; you must have PMF authority to change the panel flow to access the validation record.

Universal Partition Access Authority

Data in a Tivoli Information Management for z/OS database can be organized into logical partitions within a single physical database. A privilege class authority called *Universal Partition Access* provides access to all logical partitions. Any user with this authority has access to all records in any partition in the Tivoli Information Management for z/OS database. For more information on logical databases, see “Logical Database Partitioning” on page 203.

Assigning Authorities

When you assign authorities, consider the tasks that each of your users perform. If you organized your privilege classes according to this consideration, it is easy to assign the necessary authority to each class. Keep in mind that the fewer users with a particular authority, the more control you have over your data. This is particularly important with delete authority. If too many privilege classes can delete records, there is a greater chance that you could lose valuable data.

To allow a user to perform a specific task, you can give him a combination of authorities. For example, when a user updates a problem record, a summary of the problem is first displayed so the user can verify that it is the correct record. Thus, for the user to update this problem, he must first be allowed to display it. In this case, you would give the user authority to display and to update problem records.

Table 3 summarizes the authority levels required for specific tasks.

Table 3. Authority Levels for Specific Tasks

If you want to:	Display	Entry	Update	Assign	Close	Delete
Display records	X					
Create new records	X	X				
Update existing records	X		X			
Assign records when created	X	X		X		
Assign records when updated	X		X	X		
Close records when created	X	X			X	

Table 3. Authority Levels for Specific Tasks (continued)

If you want to:	Display	Entry	Update	Assign	Close	Delete
Close records when updated	X		X		X	
Delete records not owned	X					X
Delete records owned	X		X			X

With only display authority, the user can only display records. However, for other tasks, the user must have authority levels in appropriate combinations, always including display authority.

Example of Authority Assignments for Problem Management Users

The matrix in Table 4 shows the authority assignments for the problem management organization shown in Figure 1 on page 18.

Table 4. Privilege Class Authority Matrix for Organization

	PROBLEM						PEOPLE				
CLASS	DIS	EN	UP	AS	CL	DE	DIS	EN	UP	DE	UNIVERSAL
MASTER	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
HELPDESK	Y	Y	Y	Y	Y		Y	Y	Y		Y
TAPEOPER	Y	Y					Y				
DISKOPER	Y	Y					Y				
SYSOPER	Y	Y	Y	Y			Y	Y	Y		
SSUPER	Y	Y	Y	Y	Y		Y	Y	Y		Y
PRODCNTL	Y						Y				
SYSPROG	Y	Y	Y	Y			Y	Y	Y		
SPROG1	Y	Y	Y	Y	Y		Y	Y	Y		Y
SPROG2	Y	Y	Y	Y			Y	Y	Y		
APPLPROG	Y	Y	Y	Y			Y	Y	Y		
APROG1	Y	Y	Y	Y	Y		Y	Y	Y		Y
APROG2	Y	Y	Y	Y			Y	Y	Y		
DIS=DISPLAY EN=ENTRY UP=UPDATE AS=ASSIGNMENT CL=CLOSE DE=DELETE											
	SRC			PRIVILEGE CLASS				PMF	DBADMIN		
CLASS	EN	UP	DE	DIS	EN	UP	DE	ALL	DB	TSD	RESERVED
MASTER	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	
HELPDESK				Y							
TAPEOPER				Y							
DISKOPER				Y							
SYSOPER				Y							
SSUPER	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	
PRODCNTL				Y							
SYSPROG	Y	Y	Y	Y							
SPROG1	Y	Y	Y	Y	Y	Y	Y		Y	Y	
SPROG2	Y	Y	Y	Y							
APPLPROG	Y	Y	Y	Y							

Table 4. Privilege Class Authority Matrix for Organization (continued)

APROG1	Y	Y	Y	Y	Y	Y	Y		Y	Y	
APROG2	Y	Y	Y	Y							

Several factors are worth noting about the assignments in Table 4 on page 25:

- The master privilege class has all authorities. It is the only class having authority to delete problem records, thus ensuring that valuable data is not deleted and is available for reports. It is also the only class having authority to delete people records.
- SSUPER has all authorities except problem delete and people delete. SSUPER represents a group and is responsible for creating and maintaining SRCs, privilege classes, and panels for the group.
- SPROG1 and APROG1 have all authorities except problem delete, people delete, and PMF. They are also representatives for their groups. They have the same responsibilities as does SSUPER, but they cannot modify panels.
- HELPDESK closes problems that they resolve on the spot.
- Other groups have varying levels of authority.
- All classes are authorized to display privilege classes to stay informed of authority changes.

Deciding Whether to Use Record Ownership

Your installation can decide to control changes to data records through record ownership. The decision to use ownership for records is based on the amount of data security you want to have. If it is important to you to limit the number of users who can update or delete a particular record, that record should be transferred to an owner. However, because Tivoli Information Management for z/OS neither restricts nor enforces the use of record ownership, it is your responsibility to establish your own standards and enforce those standards within your installation.

If you decide to use ownership, each privilege class to which a record can be transferred must be given update authority for that type of record so that the record can be transferred to it. Once ownership is established, only users defined in the owning, the transfer-to, or the master privilege classes are authorized to update the record. Likewise, if a record is owned, only the master privilege class or the owning privilege class (if it has delete authority for that type of record) can delete it.

To establish initial ownership of a record, a member of a privilege class with create or update authority for the record type must enter the name of a privilege class in the record’s **Transfer-to class** field and file the record. (Note that the record can be transferred only to a privilege class that has update authority for that type of record.) When the record is filed, the privilege class that created the record (if it is a new record) or updated the record is assigned ownership immediately. If a user transfers the record to his current class, the **Transfer-to class** field is cleared when the record is filed. If a user transfers the record to a class other than his current class, the **Transfer-to class** field is cleared and ownership of the record is reassigned when a member of the transfer-to privilege class accesses the record in update mode, makes a change to the record, and subsequently files it.

If the user who creates a record does not specify an owner, any user with update authority for that record type can assign ownership of the record to himself at any time. Once ownership is assigned, only a user identified in the owning privilege class or the master privilege class can transfer ownership to another privilege class.

An **Owning Partition Name** can be added to each record so that records can be grouped into logical partitions. Information about this aspect of record ownership can be found in “Logical Database Partitioning” on page 203.

Managing Privilege Classes

The basic tasks performed in managing privilege classes are:

- Defining individual and group privilege classes
- Determining all privilege class names
- Displaying authorities for a privilege class
- Deleting members from and adding members to a privilege class
- Deleting a privilege class

To learn how to perform these tasks, go through the step-by-step scenarios on the following pages.

Note: Because list processor panels are used in Tivoli Information Management for z/OS Version 7.1 to collect and display the data for new privilege class records, you may want to run TSX BLGTPRIV to migrate all your existing priviliege class records to use the new list processor panels. Running BLGTPRIV will ensure that the same set of panels is used to enter or display data related to users of a privilege class. For more information about BLGTPRIV, see the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference*.

Defining an Individual Privilege Class

In this scenario, you define a privilege class for Tim Bennett, a user, who is the Problem/Change Coordinator. To fulfill his job responsibilities, the Problem/Change Coordinator requires all authorities for problems, changes, people, solutions, privilege classes, and SRCs. Tim also performs some program administration duties when the program administrator is absent. Therefore, he needs database administration authority. (If database partitioning is used, Tim could also be given universal authority, but only if he needs access to all of the records in the Tivoli Information Management for z/OS database. See “Logical Database Partitioning” on page 203 for details.)

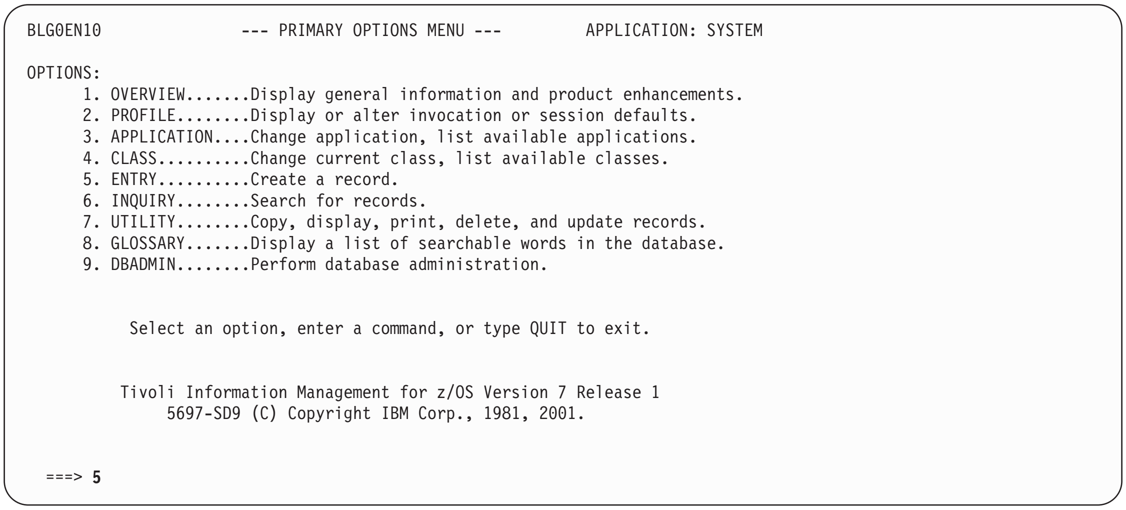
Note: You can only create a privilege class having a subset of the authorities defined in the privilege class in which you are currently running.

If the application specified in the upper-right corner of the screen is Management, change the application setting to System. Type:

```
3,1
```

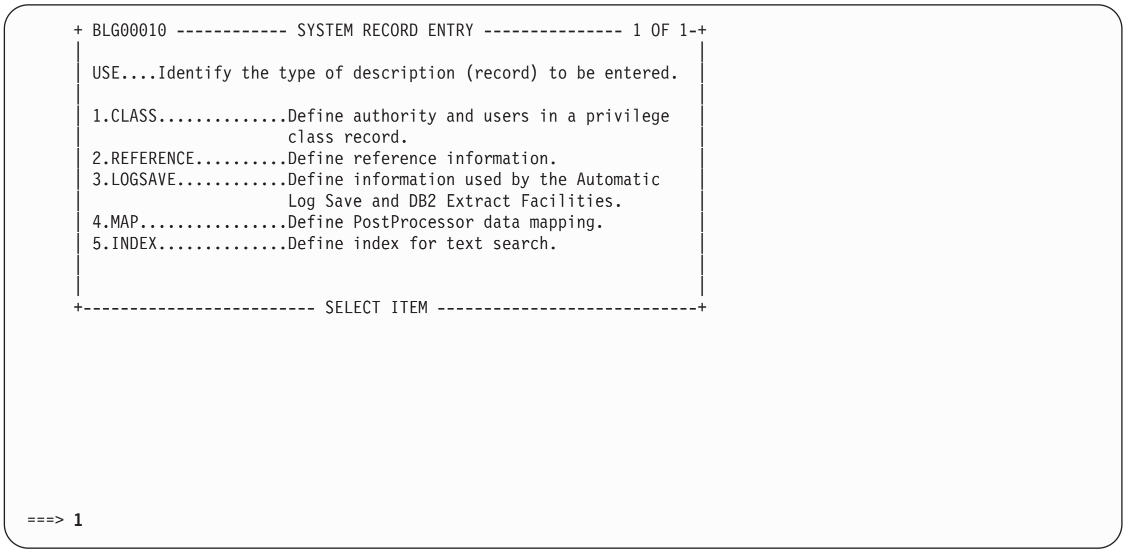
and press **Enter**.

From BLG0EN10, Primary Options Menu, select **Entry**. (Notice that you do not use the **Class** selection to create or to search privilege class records.) Type **5** and press **Enter**.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---         APPLICATION: SYSTEM

OPTIONS:
      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
               5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


  ===> 5
```

Select **Class**. Type **1** and press **Enter**.

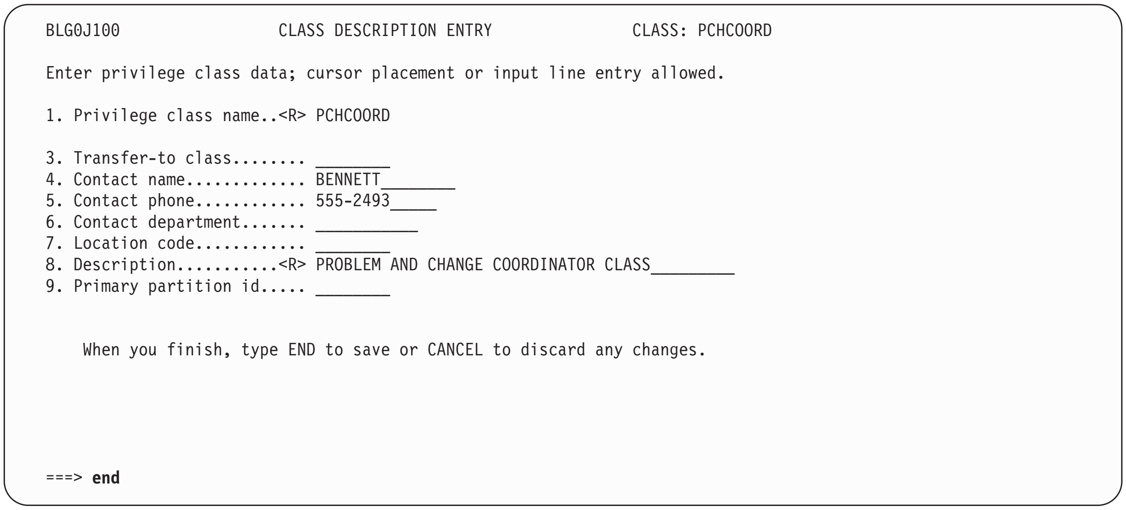


```
      + BLG00010 ------------ SYSTEM RECORD ENTRY --------------- 1 OF 1-+
      |                                                                  |
      | USE....Identify the type of description (record) to be entered.  |
      |                                                                  |
      | 1.CLASS.............Define authority and users in a privilege    |
      |                     class record.                                |
      | 2.REFERENCE.........Define reference information.                |
      | 3.LOGSAVE...........Define information used by the Automatic     |
      |                     Log Save and DB2 Extract Facilities.         |
      | 4.MAP...............Define PostProcessor data mapping.           |
      | 5.INDEX.............Define index for text search.                |
      |                                                                  |
      |                                                                  |
      +------------------------ SELECT ITEM ----------------------------+









===> 1
```

On BLG0J100, the Class Description Entry panel, complete the required fields and any others. After typing the data in the appropriate fields, press **Enter**.

(The **Primary partition id** field is used for partitioning your database. See “Logical Database Partitioning” on page 203 for details.)

To leave the panel, type **end** and press **Enter**.

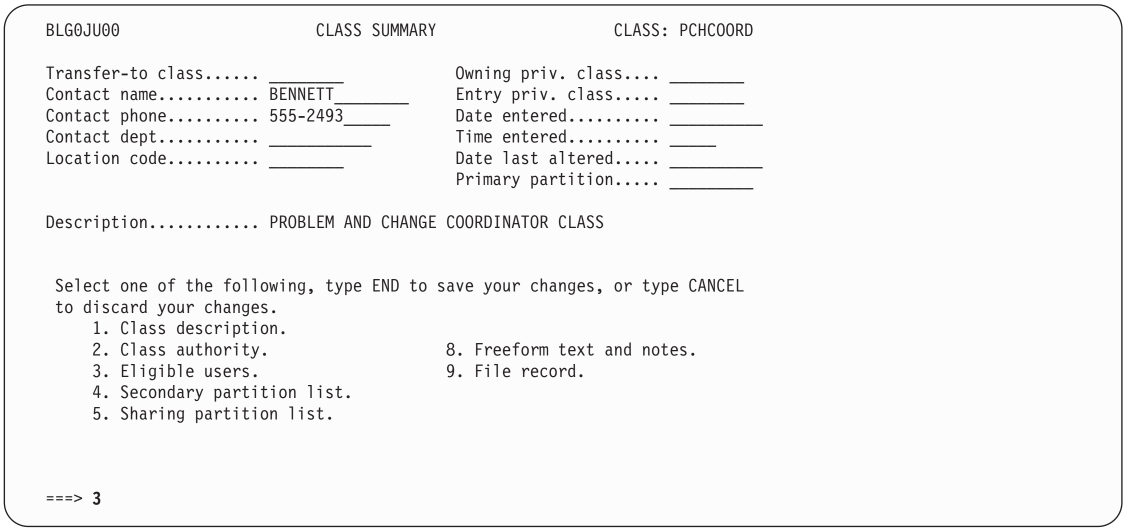


```
BLG0J100                 CLASS DESCRIPTION ENTRY               CLASS: PCHCOORD

Enter privilege class data; cursor placement or input line entry allowed.

1. Privilege class name..<R> PCHCOORD

3. Transfer-to class........ ________
4. Contact name............. BENNETT_______
5. Contact phone............ 555-2493_____
6. Contact department....... ___________
7. Location code............ ________
8. Description...........<R> PROBLEM AND CHANGE COORDINATOR CLASS_________
9. Primary partition id..... ________


    When you finish, type END to save or CANCEL to discard any changes.





===> end
```

Note: If you are managing older privilege classes, the old panel formats will be displayed unless you have run TSX BLGTPRIV to migrate older privilege class records to use the list processor panels for Tivoli Information Management for z/OS Version 7.1. For information on running TSX BLGTPRIV, see the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference*.

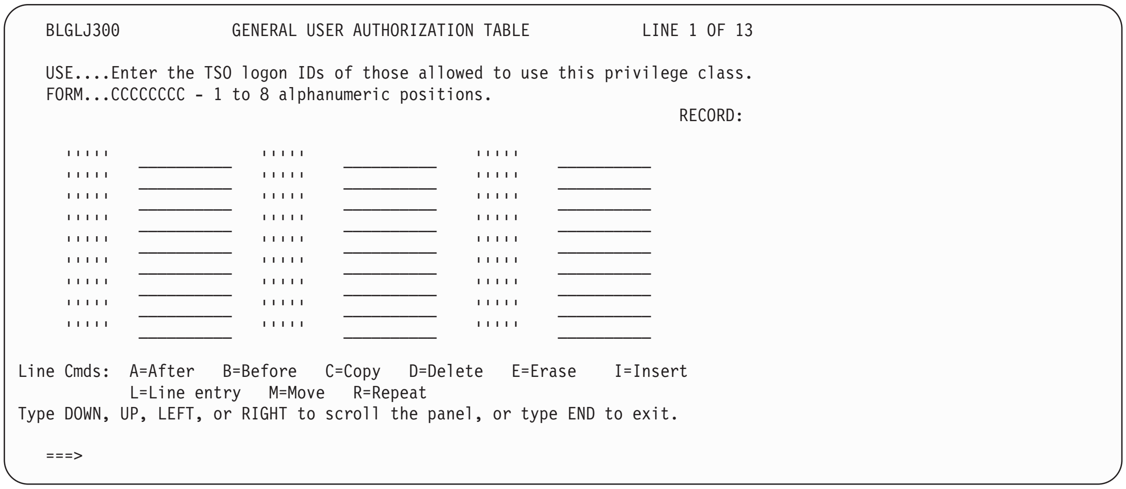
The Class Summary panel gives a summary in the upper portion and lists all categories of data that can be entered in the lower portion. This panel is redisplayed to allow you to add as much data as you want until you are ready to file the record.

To enter one or more eligible users, select **Eligible Users.** Type **3** and press **Enter**.



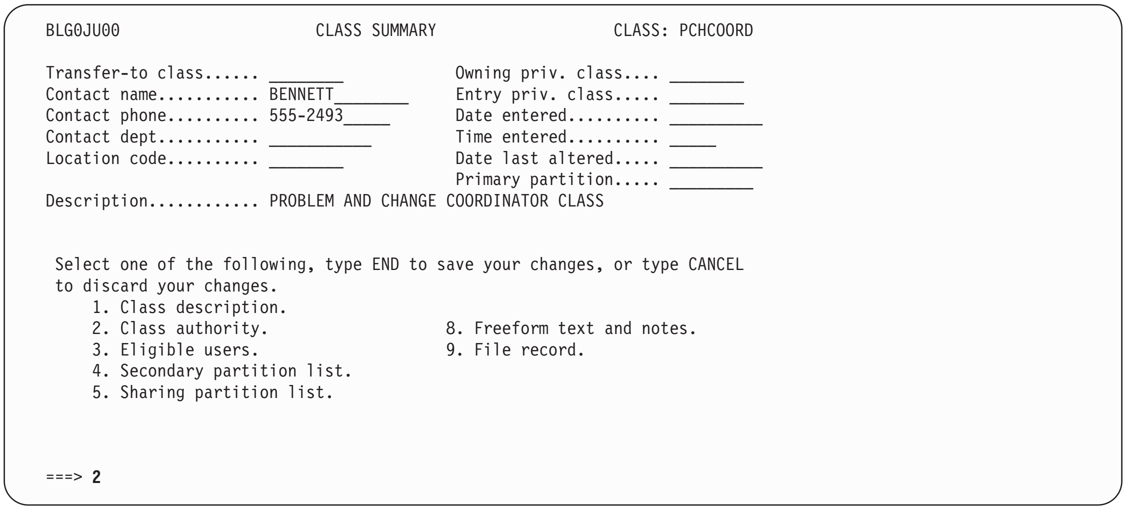
```
BLG0JU00                     CLASS SUMMARY                  CLASS: PCHCOORD

Transfer-to class...... ________            Owning priv. class.... ________
Contact name........... BENNETT________     Entry priv. class..... ________
Contact phone.......... 555-2493_____       Date entered.......... __________
Contact dept........... ___________         Time entered.......... _____
Location code.......... ________            Date last altered..... __________
                                            Primary partition..... _________

Description............ PROBLEM AND CHANGE COORDINATOR CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.
     2. Class authority.                   8. Freeform text and notes.
     3. Eligible users.                    9. File record.
     4. Secondary partition list.
     5. Sharing partition list.



===> 3
```

On panel BLGLJ300, enter one or more eligible users. To leave the panel and return to the Class Summary panel, type **end** and press **Enter**.



```
BLGLJ300            GENERAL USER AUTHORIZATION TABLE            LINE 1 OF 13

USE....Enter the TSO logon IDs of those allowed to use this privilege class.
FORM...CCCCCCCC - 1 to 8 alphanumeric positions.
                                                                    RECORD:

  '''''   __________   '''''   __________   '''''   __________
  '''''   __________   '''''   __________   '''''   __________
  '''''   __________   '''''   __________   '''''   __________
  '''''   __________   '''''   __________   '''''   __________
  '''''   __________   '''''   __________   '''''   __________
  '''''   __________   '''''   __________   '''''   __________
  '''''   __________   '''''   __________   '''''   __________
  '''''   __________   '''''   __________   '''''   __________
  '''''   __________   '''''   __________   '''''   __________

Line Cmds:  A=After   B=Before   C=Copy   D=Delete   E=Erase    I=Insert
            L=Line entry   M=Move   R=Repeat
Type DOWN, UP, LEFT, or RIGHT to scroll the panel, or type END to exit.

===>
```

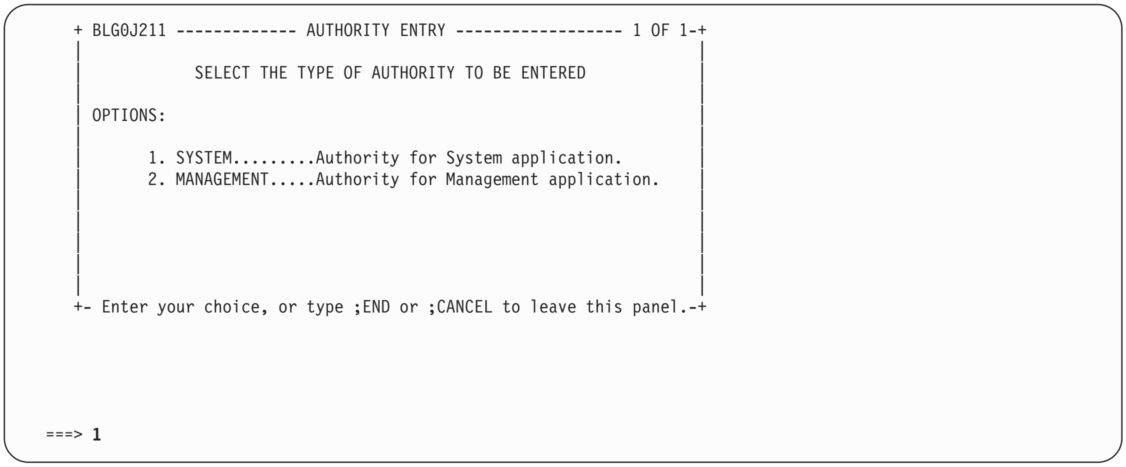
Now to specify the authority, select **Class authority**. Type **2** and press **Enter**.



```
BLG0JU00                    CLASS SUMMARY                  CLASS: PCHCOORD

Transfer-to class...... ________           Owning priv. class.... ________
Contact name........... BENNETT_______     Entry priv. class..... ________
Contact phone.......... 555-2493_____      Date entered.......... __________
Contact dept........... ___________        Time entered.......... _____
Location code.......... ________           Date last altered..... __________
                                           Primary partition..... _________
Description............ PROBLEM AND CHANGE COORDINATOR CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.
     2. Class authority.                  8. Freeform text and notes.
     3. Eligible users.                   9. File record.
     4. Secondary partition list.
     5. Sharing partition list.



===> 2
```

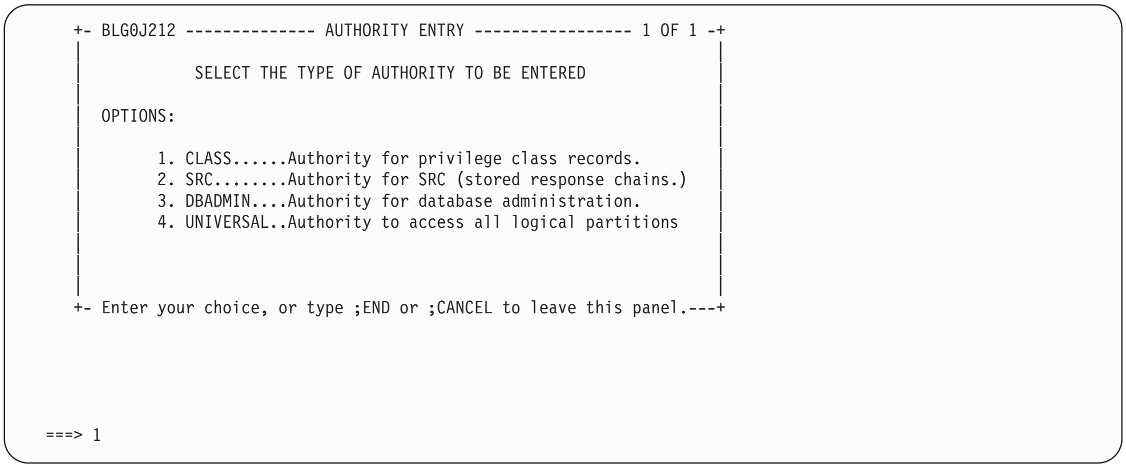
Assign authority for system application records. Select **System** by typing **1** and pressing **Enter**.

You select **System** because SRCs and privilege classes are system application records, and because database administration is a System application authority.



```
   + BLG0J211 ------------- AUTHORITY ENTRY ------------------ 1 OF 1-+
   |                                                                  |
   |            SELECT THE TYPE OF AUTHORITY TO BE ENTERED            |
   |                                                                  |
   | OPTIONS:                                                         |
   |                                                                  |
   |       1. SYSTEM.........Authority for System application.        |
   |       2. MANAGEMENT.....Authority for Management application.    |
   |                                                                  |
   |                                                                  |
   |                                                                  |
   |                                                                  |
   |                                                                  |
   +- Enter your choice, or type ;END or ;CANCEL to leave this panel.-+





===> 1
```

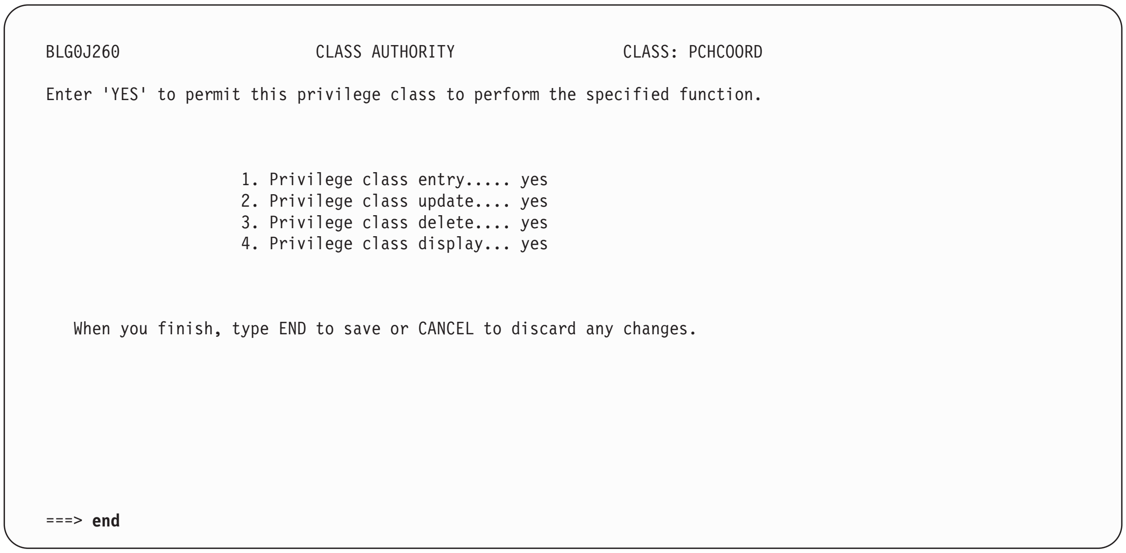
Select **Class**. Type **1** and press **Enter**.



```
   +- BLG0J212 -------------- AUTHORITY ENTRY ----------------- 1 OF 1 -+
   |                                                                    |
   |            SELECT THE TYPE OF AUTHORITY TO BE ENTERED              |
   |                                                                    |
   |  OPTIONS:                                                          |
   |                                                                    |
   |        1. CLASS......Authority for privilege class records.        |
   |        2. SRC........Authority for SRC (stored response chains.)   |
   |        3. DBADMIN....Authority for database administration.        |
   |        4. UNIVERSAL..Authority to access all logical partitions    |
   |                                                                    |
   |                                                                    |
   |                                                                    |
   +- Enter your choice, or type ;END or ;CANCEL to leave this panel.---+





===> 1
```

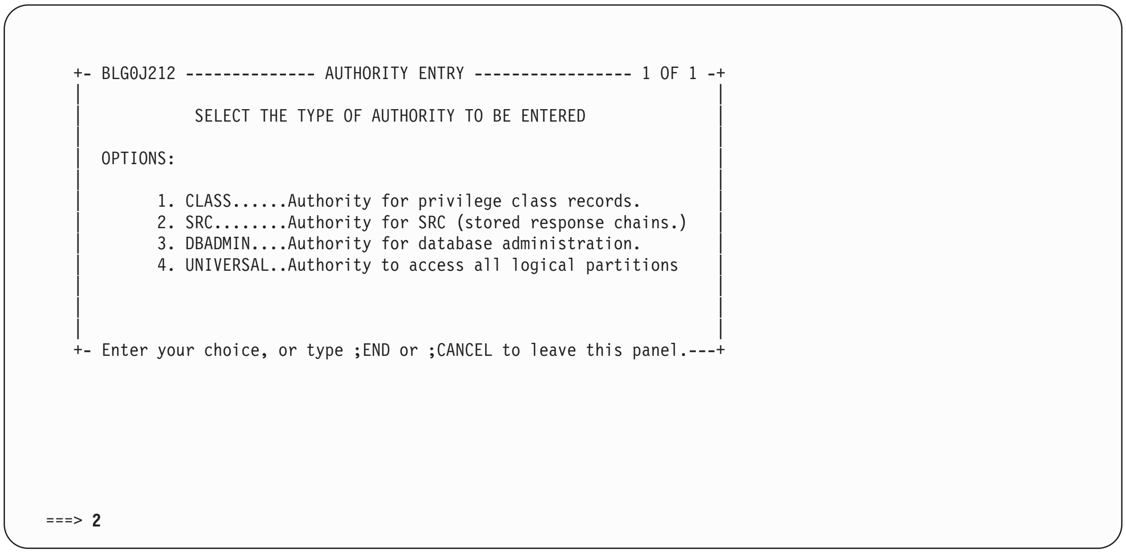
From this panel, complete the necessary fields by typing **YES** for all the functions, because Tim Bennett requires all authorities in order to perform his job.

To exit, type **end** and press **Enter**.



```
BLG0J260                    CLASS AUTHORITY                 CLASS: PCHCOORD

Enter 'YES' to permit this privilege class to perform the specified function.



                    1. Privilege class entry..... yes
                    2. Privilege class update.... yes
                    3. Privilege class delete.... yes
                    4. Privilege class display... yes



   When you finish, type END to save or CANCEL to discard any changes.








===> end
```

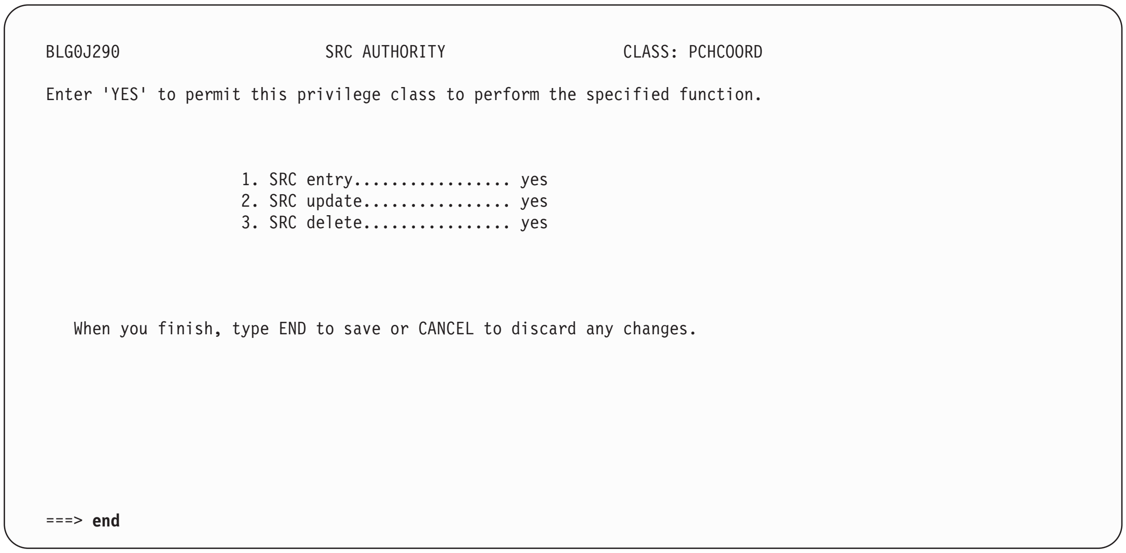
You return to BLG0J212, the Authority Entry panel. Select **SRC** (for stored response chains) by typing **2** and pressing **Enter**.



```
   +- BLG0J212 -------------- AUTHORITY ENTRY ----------------- 1 OF 1 -+
   |                                                                    |
   |          SELECT THE TYPE OF AUTHORITY TO BE ENTERED                |
   |                                                                    |
   |  OPTIONS:                                                          |
   |                                                                    |
   |        1. CLASS......Authority for privilege class records.        |
   |        2. SRC........Authority for SRC (stored response chains.)   |
   |        3. DBADMIN....Authority for database administration.        |
   |        4. UNIVERSAL..Authority to access all logical partitions    |
   |                                                                    |
   |                                                                    |
   |                                                                    |
   +- Enter your choice, or type ;END or ;CANCEL to leave this panel.---+







===> 2
```

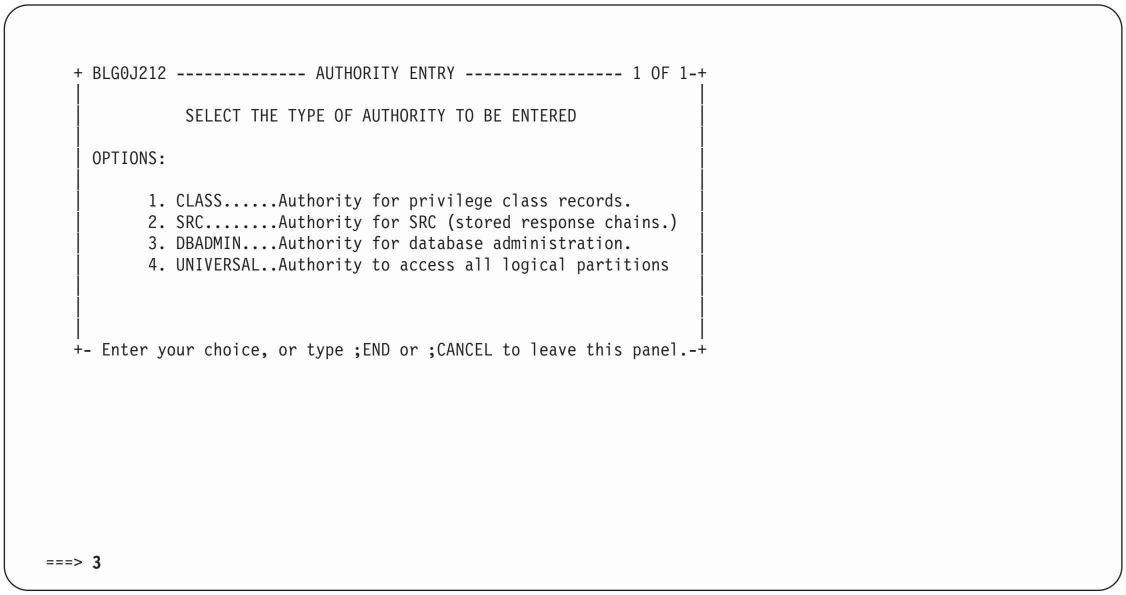
From this panel, type **yes** in the field for each function you want Tim Bennett to be able to do.

To leave this panel type **end** and press **Enter**.



```
BLG0J290                      SRC AUTHORITY                  CLASS: PCHCOORD

Enter 'YES' to permit this privilege class to perform the specified function.



                     1. SRC entry................. yes
                     2. SRC update................ yes
                     3. SRC delete................ yes





   When you finish, type END to save or CANCEL to discard any changes.









===> end
```

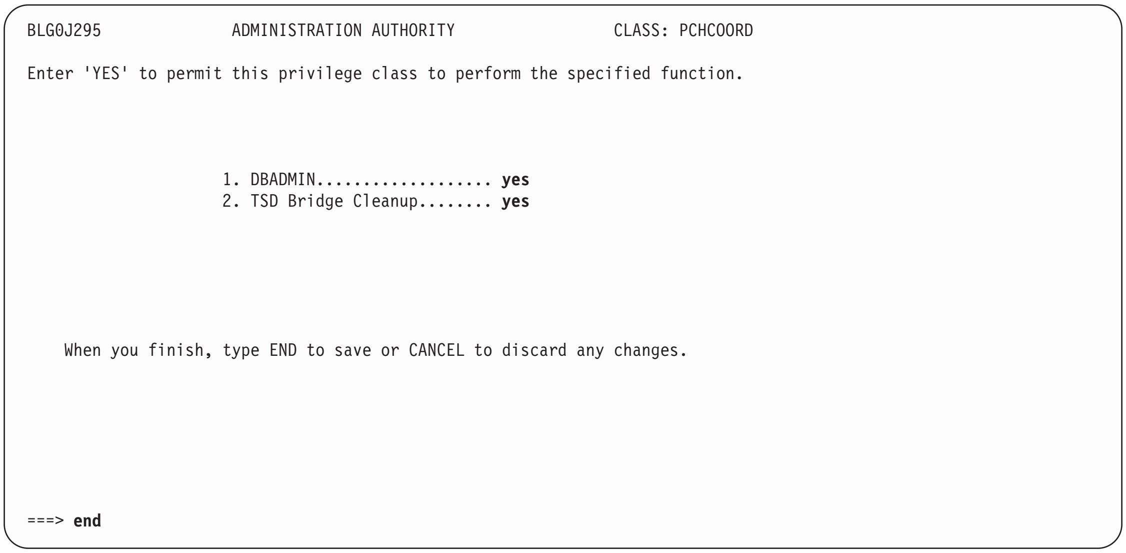
You return to BLG0J212, the Authority Entry panel. Select **DBADMIN** by typing **3** and pressing **Enter**.



```
   + BLG0J212 -------------- AUTHORITY ENTRY ----------------- 1 OF 1-+
   |                                                                  |
   |          SELECT THE TYPE OF AUTHORITY TO BE ENTERED              |
   |                                                                  |
   | OPTIONS:                                                         |
   |                                                                  |
   |       1. CLASS......Authority for privilege class records.       |
   |       2. SRC........Authority for SRC (stored response chains.)  |
   |       3. DBADMIN....Authority for database administration.       |
   |       4. UNIVERSAL..Authority to access all logical partitions   |
   |                                                                  |
   |                                                                  |
   |                                                                  |
   +- Enter your choice, or type ;END or ;CANCEL to leave this panel.-+









===> 3
```

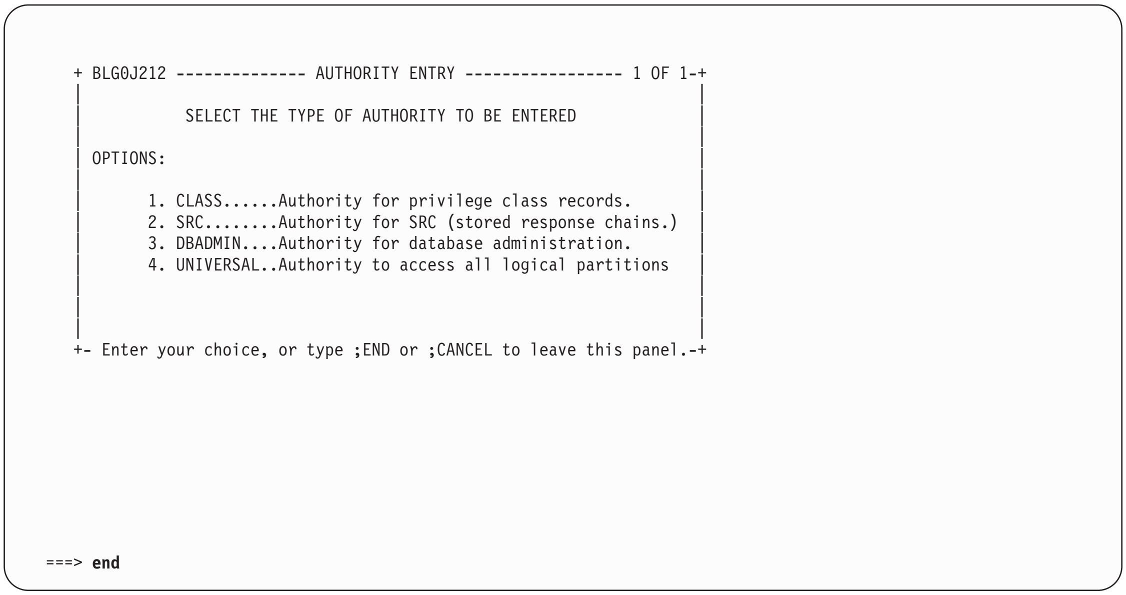
From this panel, type **yes** in the field for each function you want Tim Bennett to be able to do.

To leave this panel type **end** and press **Enter**.



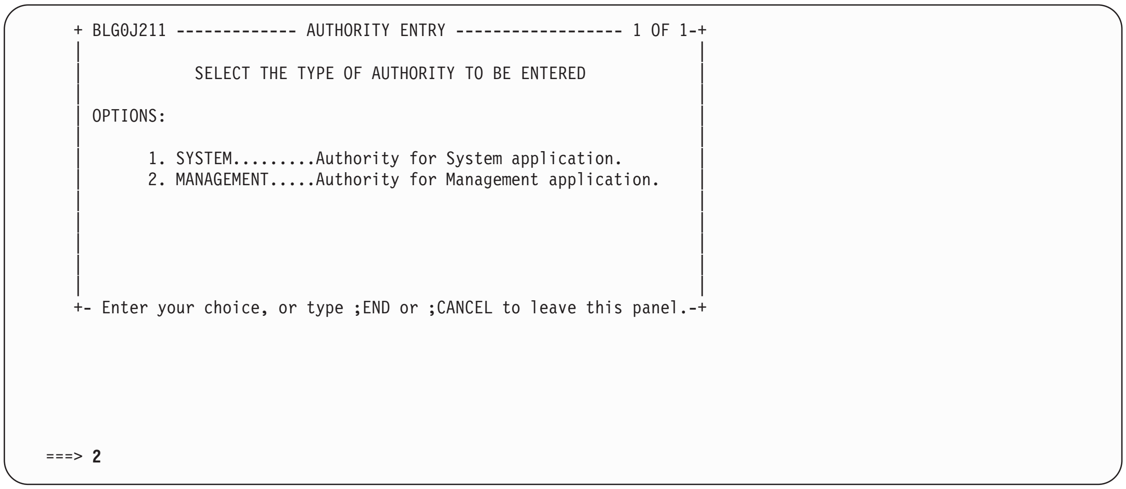
```
BLG0J295             ADMINISTRATION AUTHORITY                CLASS: PCHCOORD

Enter 'YES' to permit this privilege class to perform the specified function.




                     1. DBADMIN................... yes
                     2. TSD Bridge Cleanup........ yes







    When you finish, type END to save or CANCEL to discard any changes.








===> end
```

You return to BLG0J212, the Authority Entry panel. Type **end** and press **Enter** because you are finished giving System application authority.



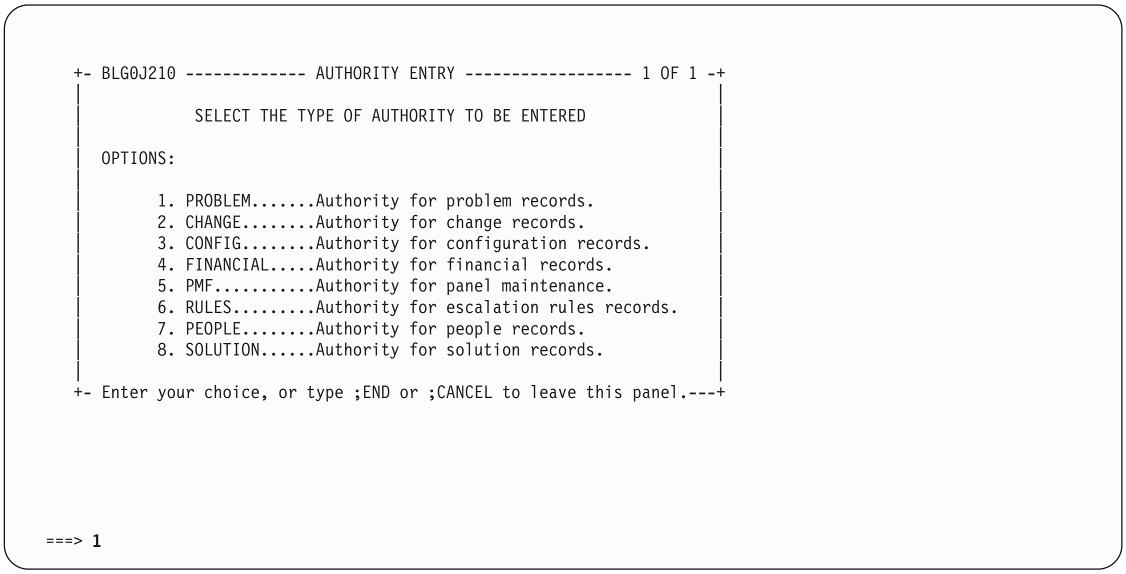
```
     + BLG0J212 -------------- AUTHORITY ENTRY ----------------- 1 OF 1-+
     |                                                                  |
     |           SELECT THE TYPE OF AUTHORITY TO BE ENTERED             |
     |                                                                  |
     | OPTIONS:                                                         |
     |                                                                  |
     |       1. CLASS......Authority for privilege class records.       |
     |       2. SRC........Authority for SRC (stored response chains.)  |
     |       3. DBADMIN....Authority for database administration.       |
     |       4. UNIVERSAL..Authority to access all logical partitions   |
     |                                                                  |
     |                                                                  |
     |                                                                  |
     +- Enter your choice, or type ;END or ;CANCEL to leave this panel.-+









  ===> end
```

Now, you are ready to give Management application authorities. Select **Management**. Type **2** and press **Enter**.



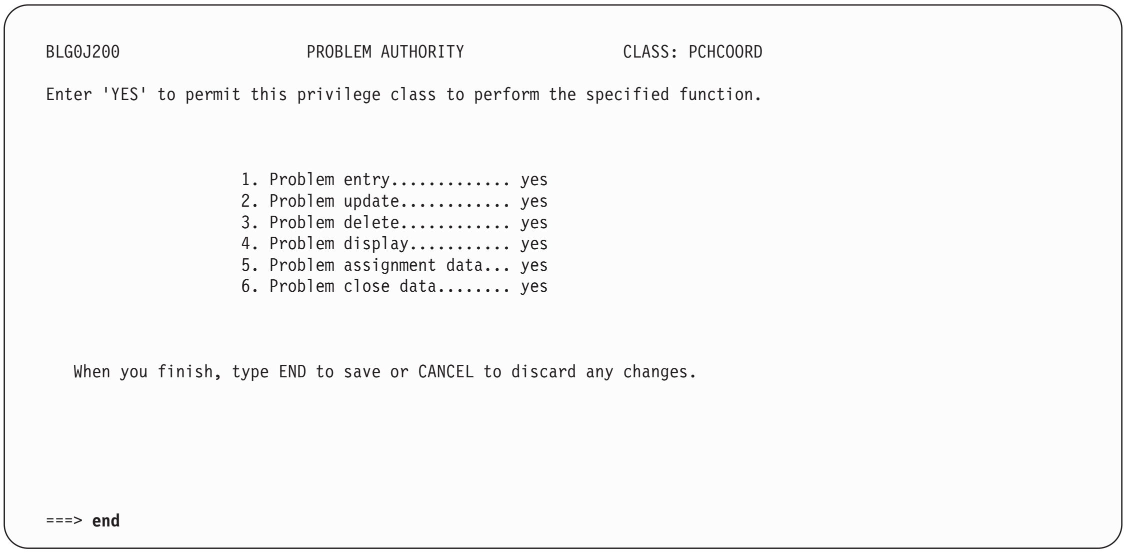
```
+ BLG0J211 ------------- AUTHORITY ENTRY ------------------ 1 OF 1-+
|                                                                  |
|            SELECT THE TYPE OF AUTHORITY TO BE ENTERED            |
|                                                                  |
| OPTIONS:                                                         |
|                                                                  |
|      1. SYSTEM.........Authority for System application.         |
|      2. MANAGEMENT.....Authority for Management application.     |
|                                                                  |
|                                                                  |
|                                                                  |
|                                                                  |
|                                                                  |
+- Enter your choice, or type ;END or ;CANCEL to leave this panel.-+






===> 2
```

Select **Problem** to give authority for problem records. Type **1** and press **Enter**.



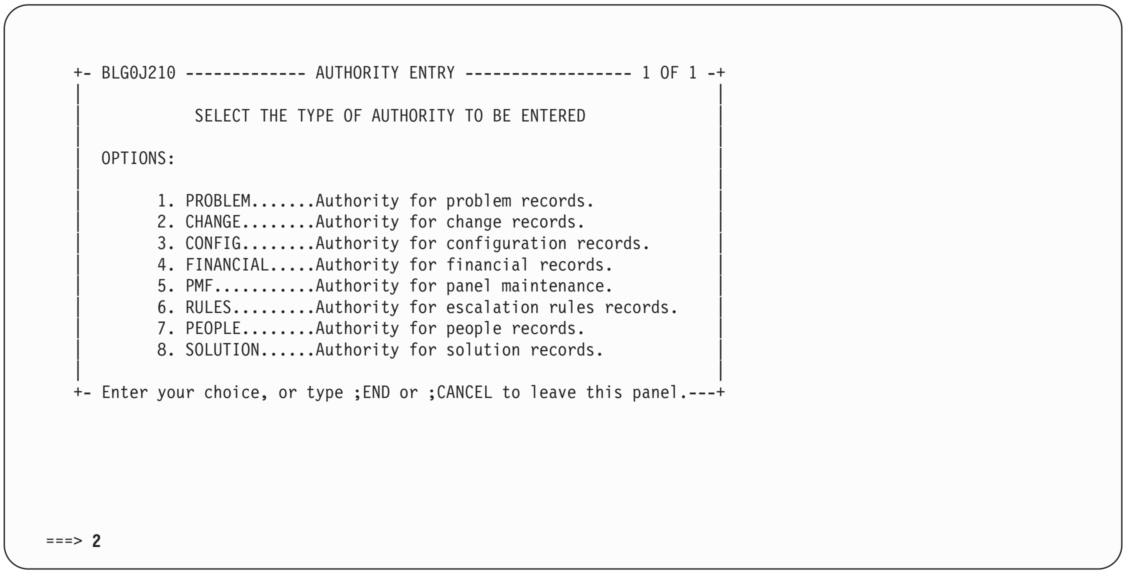
```
+- BLG0J210 ------------- AUTHORITY ENTRY ------------------ 1 OF 1 -+
|                                                                    |
|            SELECT THE TYPE OF AUTHORITY TO BE ENTERED              |
|                                                                    |
|  OPTIONS:                                                          |
|                                                                    |
|       1. PROBLEM.......Authority for problem records.              |
|       2. CHANGE........Authority for change records.               |
|       3. CONFIG........Authority for configuration records.        |
|       4. FINANCIAL.....Authority for financial records.            |
|       5. PMF...........Authority for panel maintenance.            |
|       6. RULES.........Authority for escalation rules records.     |
|       7. PEOPLE........Authority for people records.               |
|       8. SOLUTION......Authority for solution records.             |
|                                                                    |
+- Enter your choice, or type ;END or ;CANCEL to leave this panel.---+






===> 1
```

Type **yes** for as many functions as you want Tim Bennett to be able to do. Because he is the problem coordinator, he needs all authorities. To exit, type **end** and press **Enter**.



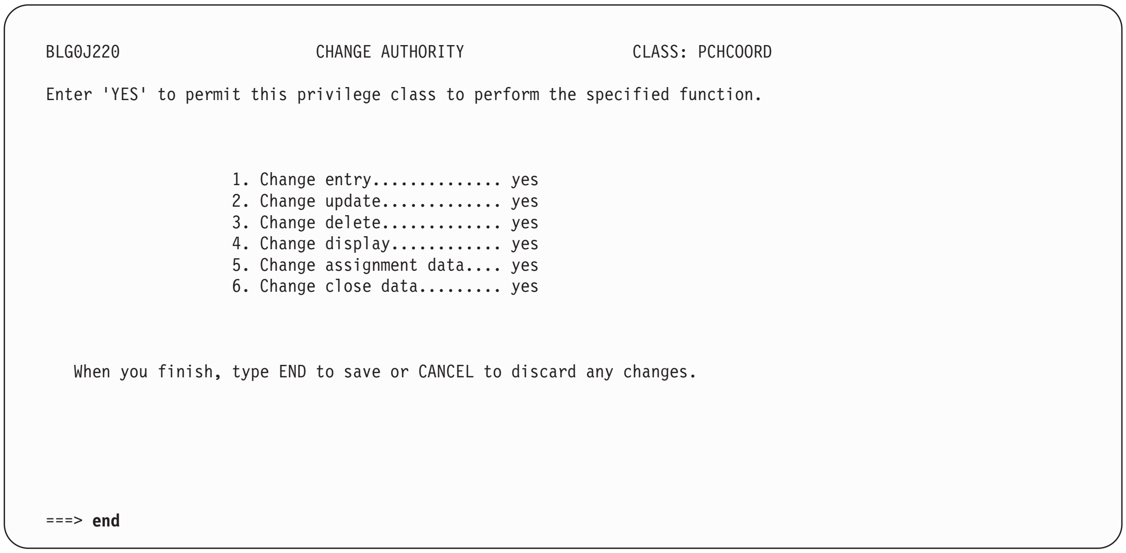
```
BLG0J200                    PROBLEM AUTHORITY                 CLASS: PCHCOORD

Enter 'YES' to permit this privilege class to perform the specified function.



                     1. Problem entry............. yes
                     2. Problem update............ yes
                     3. Problem delete............ yes
                     4. Problem display........... yes
                     5. Problem assignment data... yes
                     6. Problem close data........ yes



   When you finish, type END to save or CANCEL to discard any changes.







===> end
```

Select **Change** to give authority for change records. Type **2** and press **Enter**.



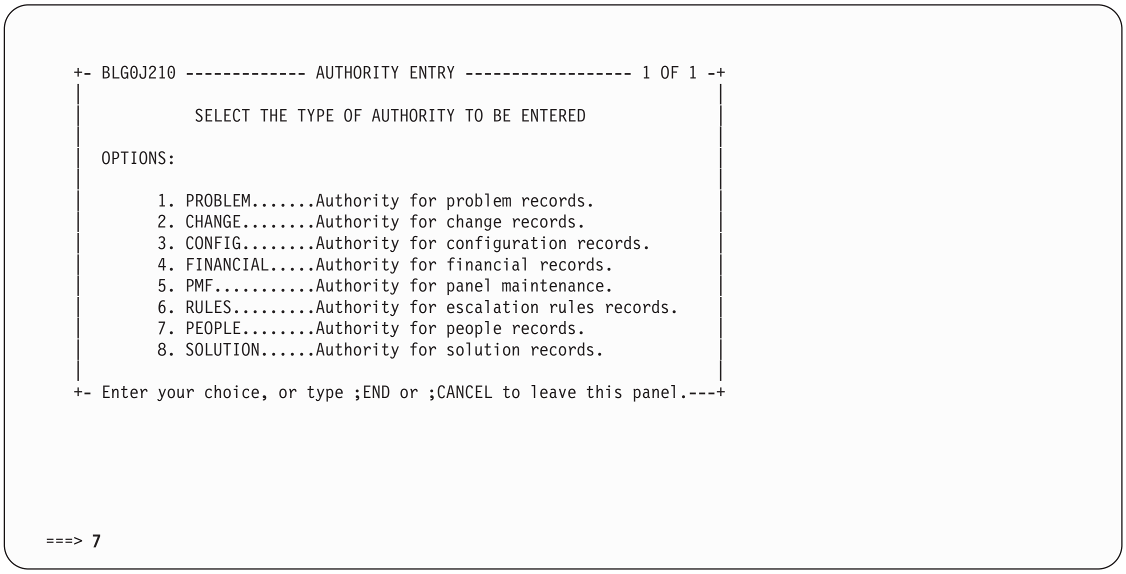
```
   +- BLG0J210 ------------- AUTHORITY ENTRY ------------------ 1 OF 1 -+
   |                                                                    |
   |           SELECT THE TYPE OF AUTHORITY TO BE ENTERED               |
   |                                                                    |
   |  OPTIONS:                                                          |
   |                                                                    |
   |        1. PROBLEM.......Authority for problem records.             |
   |        2. CHANGE........Authority for change records.              |
   |        3. CONFIG........Authority for configuration records.       |
   |        4. FINANCIAL.....Authority for financial records.           |
   |        5. PMF...........Authority for panel maintenance.           |
   |        6. RULES.........Authority for escalation rules records.    |
   |        7. PEOPLE........Authority for people records.              |
   |        8. SOLUTION......Authority for solution records.            |
   |                                                                    |
   +- Enter your choice, or type ;END or ;CANCEL to leave this panel.---+






===> 2
```

From this panel, type **yes** for all the functions, because Tim Bennett requires all authorities in order to perform his job as change coordinator. To exit, type **end** and press **Enter**.



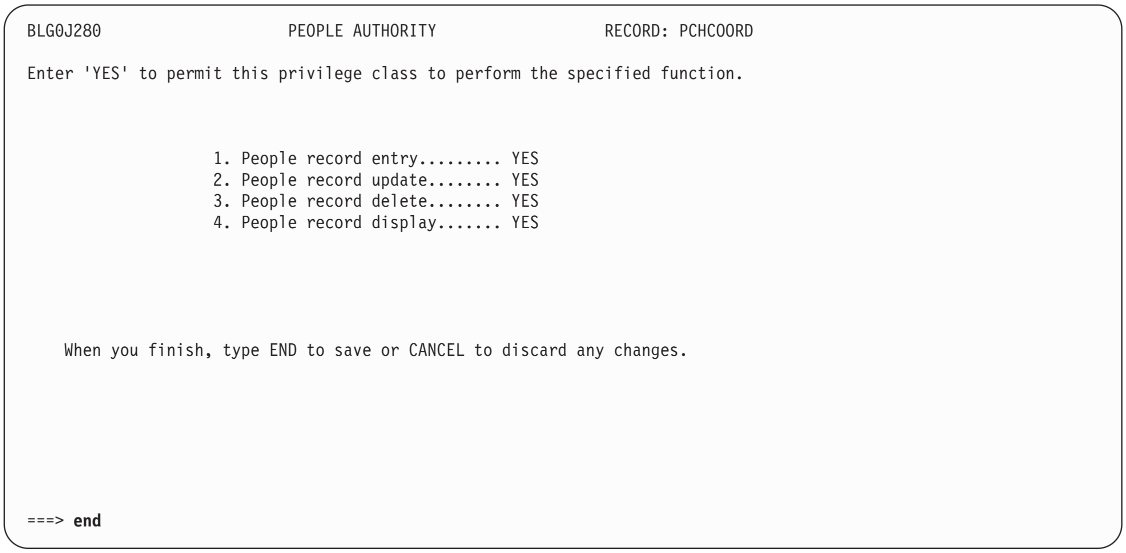
```
BLG0J220                    CHANGE AUTHORITY                 CLASS: PCHCOORD

Enter 'YES' to permit this privilege class to perform the specified function.



                    1. Change entry............. yes
                    2. Change update............ yes
                    3. Change delete............ yes
                    4. Change display........... yes
                    5. Change assignment data.... yes
                    6. Change close data......... yes



   When you finish, type END to save or CANCEL to discard any changes.






===> end
```

Select **People** to give authority for people records. Type **7** and press **Enter**.



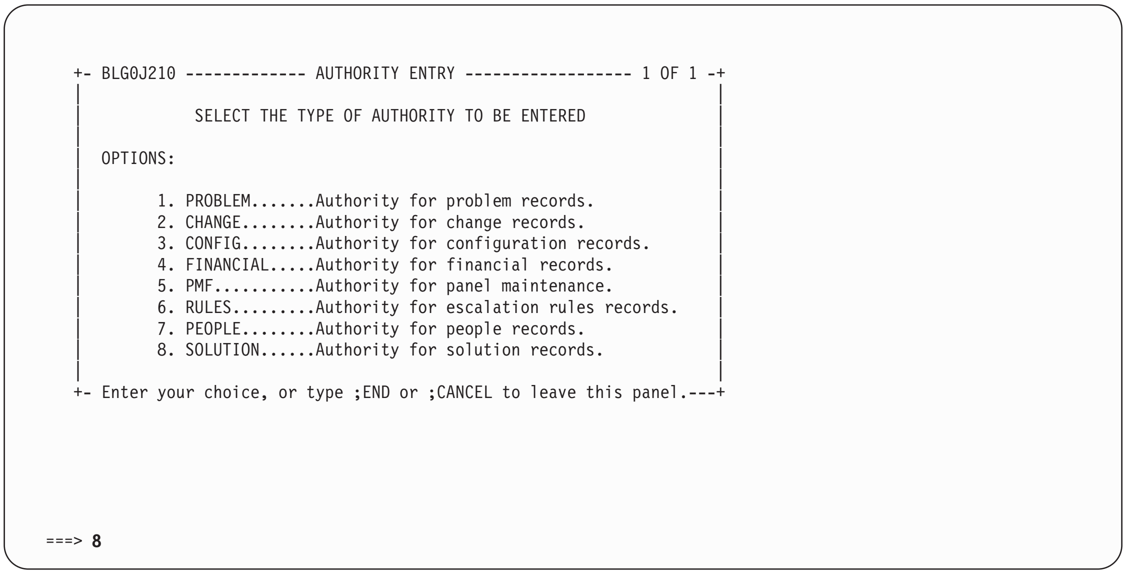
```
   +- BLG0J210 ------------- AUTHORITY ENTRY ------------------ 1 OF 1 -+
   |                                                                    |
   |          SELECT THE TYPE OF AUTHORITY TO BE ENTERED                |
   |                                                                    |
   |  OPTIONS:                                                          |
   |                                                                    |
   |      1. PROBLEM.......Authority for problem records.               |
   |      2. CHANGE........Authority for change records.                |
   |      3. CONFIG........Authority for configuration records.         |
   |      4. FINANCIAL.....Authority for financial records.             |
   |      5. PMF...........Authority for panel maintenance.             |
   |      6. RULES.........Authority for escalation rules records.      |
   |      7. PEOPLE........Authority for people records.                |
   |      8. SOLUTION......Authority for solution records.              |
   |                                                                    |
   +- Enter your choice, or type ;END or ;CANCEL to leave this panel.---+





===> 7
```

On this panel, type **yes** for all the functions, because Tim Bennett requires all authorities having to do with people records. To leave this panel and return to the Class Summary Panel, type **end,end,end** and press **Enter**.



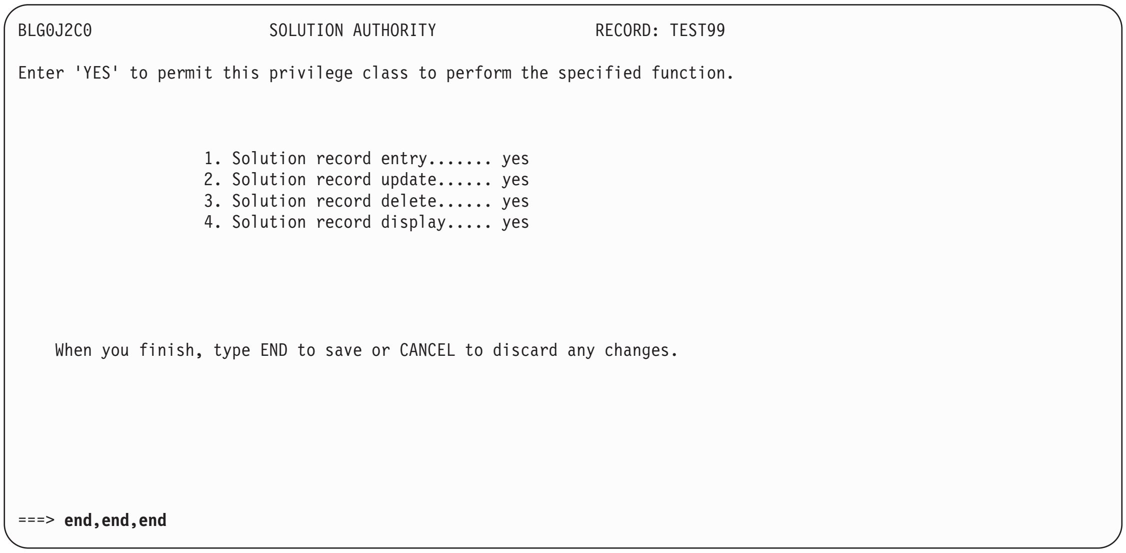
```
BLG0J280                    PEOPLE AUTHORITY                 RECORD: PCHCOORD

Enter 'YES' to permit this privilege class to perform the specified function.



                    1. People record entry......... YES
                    2. People record update........ YES
                    3. People record delete........ YES
                    4. People record display....... YES






    When you finish, type END to save or CANCEL to discard any changes.







===> end
```

Select **Solution** to give authority for solution records. Type **8** and press **Enter**.



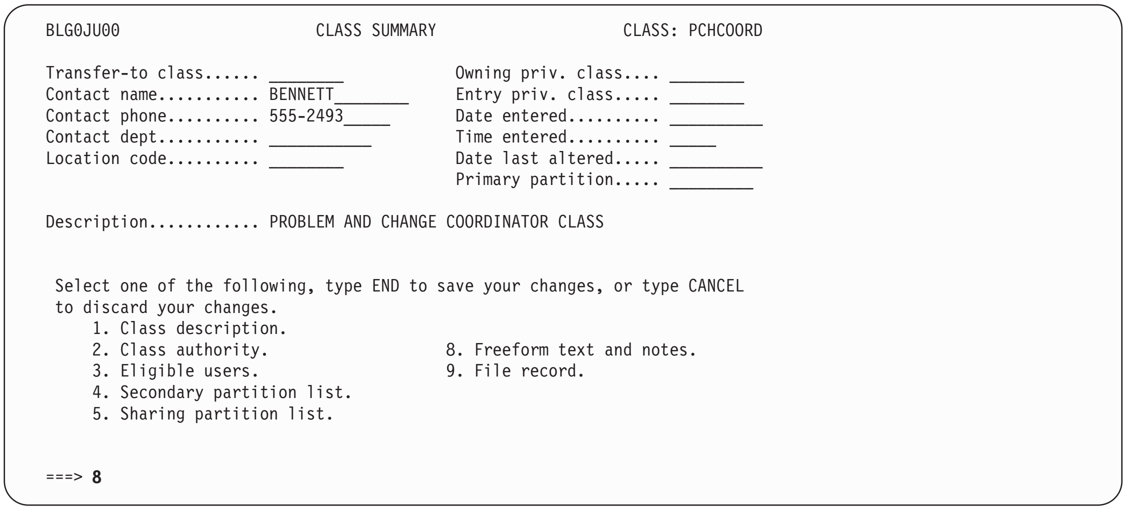
```
     +- BLG0J210 ------------- AUTHORITY ENTRY ------------------ 1 OF 1 -+
     |                                                                    |
     |          SELECT THE TYPE OF AUTHORITY TO BE ENTERED                |
     |                                                                    |
     |  OPTIONS:                                                          |
     |                                                                    |
     |        1. PROBLEM.......Authority for problem records.             |
     |        2. CHANGE........Authority for change records.              |
     |        3. CONFIG........Authority for configuration records.       |
     |        4. FINANCIAL.....Authority for financial records.           |
     |        5. PMF...........Authority for panel maintenance.           |
     |        6. RULES.........Authority for escalation rules records.    |
     |        7. PEOPLE........Authority for people records.              |
     |        8. SOLUTION......Authority for solution records.            |
     |                                                                    |
     +- Enter your choice, or type ;END or ;CANCEL to leave this panel.---+






  ===> 8
```

On this panel, type **yes** for all the functions, because Tim Bennett requires all authorities having to do with solution records. Note that solution record display authority is already set to yes, because *problem* display authority was previously set to yes, and as was discussed in the “Understanding Authorities” on page 19, problem display and solution display use the same authority. To leave this panel and return to the Class Summary Panel, type **end,end,end** and press **Enter**.



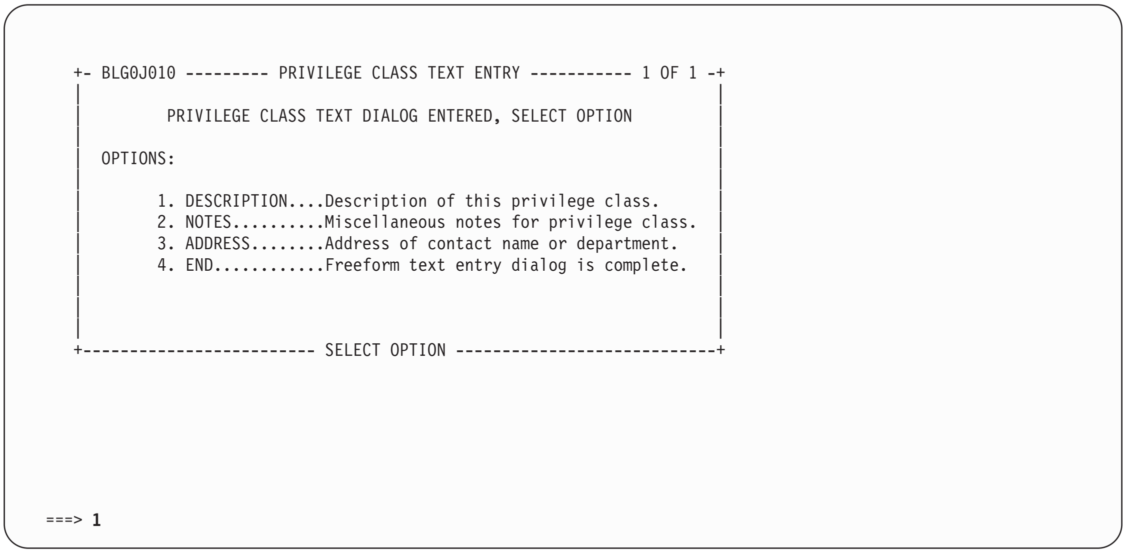
```
BLG0J2C0                  SOLUTION AUTHORITY                RECORD: TEST99

Enter 'YES' to permit this privilege class to perform the specified function.



                    1. Solution record entry....... yes
                    2. Solution record update...... yes
                    3. Solution record delete...... yes
                    4. Solution record display..... yes






    When you finish, type END to save or CANCEL to discard any changes.







===> end,end,end
```

Up to this point, you have described a privilege class and given it several kinds of authority. Now, provide some descriptive text about this class. Select **Freeform text and notes**. Type **8** and press **Enter**.



```
BLG0JU00                  CLASS SUMMARY                   CLASS: PCHCOORD

Transfer-to class...... ________            Owning priv. class.... ________
Contact name........... BENNETT________     Entry priv. class..... ________
Contact phone.......... 555-2493____        Date entered.......... __________
Contact dept........... ___________         Time entered.......... _____
Location code.......... ________            Date last altered..... __________
                                            Primary partition..... _________

Description............ PROBLEM AND CHANGE COORDINATOR CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.
     2. Class authority.                  8. Freeform text and notes.
     3. Eligible users.                   9. File record.
     4. Secondary partition list.
     5. Sharing partition list.


===> 8
```

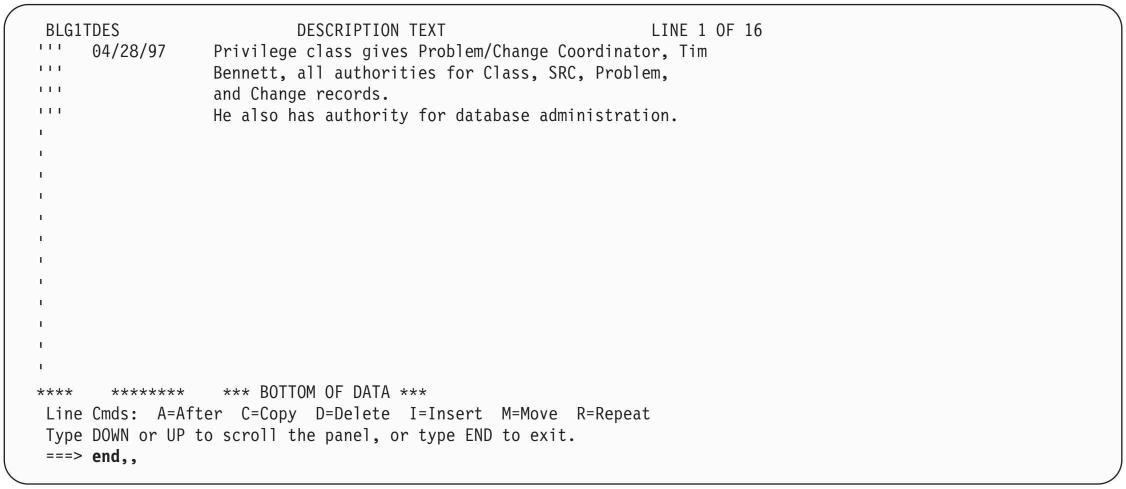
Select **Description**. Type **1** and press **Enter**.



```
+- BLG0J010 --------- PRIVILEGE CLASS TEXT ENTRY ----------- 1 OF 1 -+
|                                                                     |
|         PRIVILEGE CLASS TEXT DIALOG ENTERED, SELECT OPTION          |
|                                                                     |
|  OPTIONS:                                                           |
|                                                                     |
|        1. DESCRIPTION....Description of this privilege class.       |
|        2. NOTES..........Miscellaneous notes for privilege class.   |
|        3. ADDRESS........Address of contact name or department.     |
|        4. END............Freeform text entry dialog is complete.    |
|                                                                     |
|                                                                     |
|                                                                     |
+------------------------- SELECT OPTION ----------------------------+




 ===> 1
```

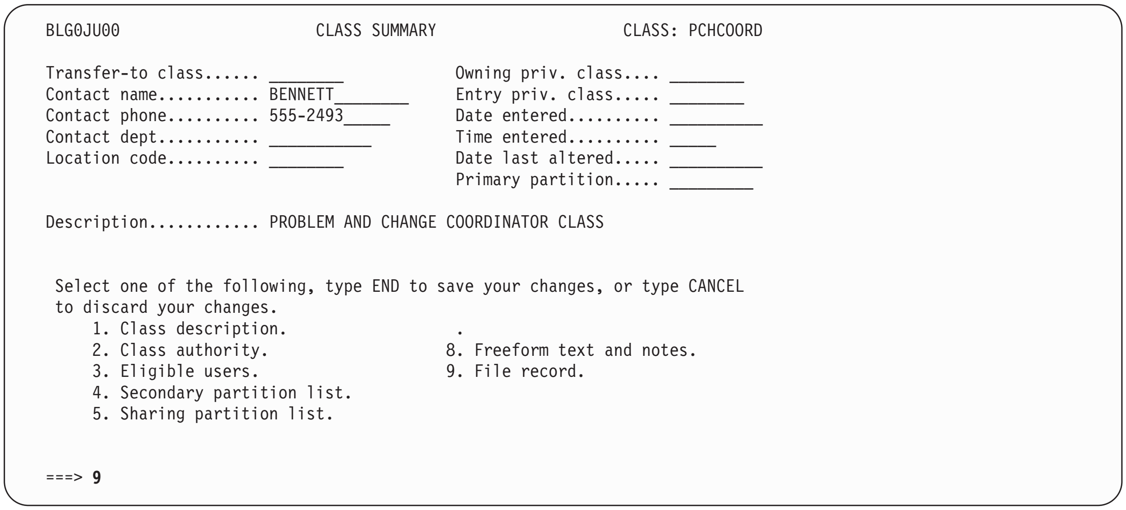
If your profile setting for **Editor selection** is **INFO**, the following panel appears. For more information on selecting an editor, refer to the *Tivoli Information Management for z/OS User's Guide*.

Type the text on the full-screen text editor panel. When the text is complete, save it and bypass the text selection panel. Type **end,,** and press **Enter**.



```
 BLG1TDES                DESCRIPTION TEXT                     LINE 1 OF 16
'''   04/28/97     Privilege class gives Problem/Change Coordinator, Tim
'''                Bennett, all authorities for Class, SRC, Problem,
'''                and Change records.
'''                He also has authority for database administration.
'
'
'
'
'
'
'
'
'
'
'
'
****    ********    *** BOTTOM OF DATA ***
 Line Cmds:  A=After  C=Copy  D=Delete  I=Insert  M=Move  R=Repeat
 Type DOWN or UP to scroll the panel, or type END to exit.
 ===> end,,
```

Once again, the Class Summary panel appears. If you want to include other users in this class, you can select **Eligible users** from this panel and add them. (If your database is logically partitioned, you can use selections 4 and 5 to identify partition information for this privilege class.) For this example, save the data you have created. Select **File record**. Type **9** and press **Enter**.



```
BLG0JU00                     CLASS SUMMARY                    CLASS: PCHCOORD

Transfer-to class...... ________            Owning priv. class.... ________
Contact name........... BENNETT________     Entry priv. class..... ________
Contact phone.......... 555-2493_____       Date entered.......... __________
Contact dept........... ___________         Time entered.......... _____
Location code.......... ________            Date last altered..... __________
                                            Primary partition..... _________

Description............ PROBLEM AND CHANGE COORDINATOR CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.                   .
     2. Class authority.                    8. Freeform text and notes.
     3. Eligible users.                     9. File record.
     4. Secondary partition list.
     5. Sharing partition list.


===> 9
```

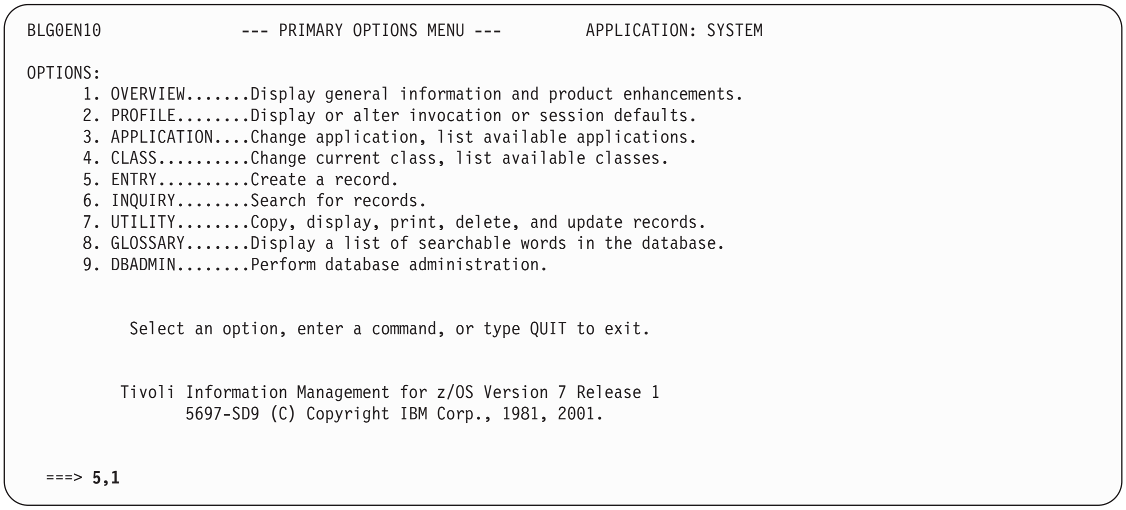
Write down the record ID (name) assigned to the record. The record ID is the privilege class name that you entered on panel BLG0J100, Class Description Entry. The record ID for this privilege class is PCHCOORD.

Defining a Group Privilege Class

As the Problem/Change Coordinator, Tim Bennett, needs to give three people at the user help desk authority to open, update, assign, and close problems.

If the application specified in the upper-right corner of the screen is Management, change the setting to System. Using an immediate response chain (IRC), type **3,1** and press **Enter**.

Select **Entry** from the Primary Options Menu and **Class** from the System Record Entry panel. Type **5,1** and press **Enter**.

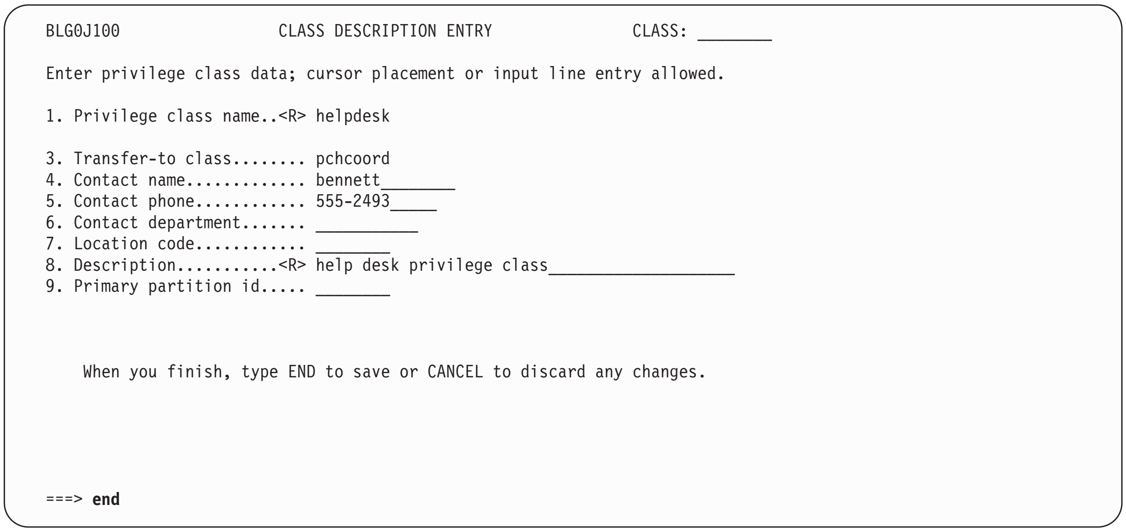


```
BLG0EN10              --- PRIMARY OPTIONS MENU ---         APPLICATION: SYSTEM

OPTIONS:
      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


  ===> 5,1
```

Complete the data on the Class Description Entry panel.

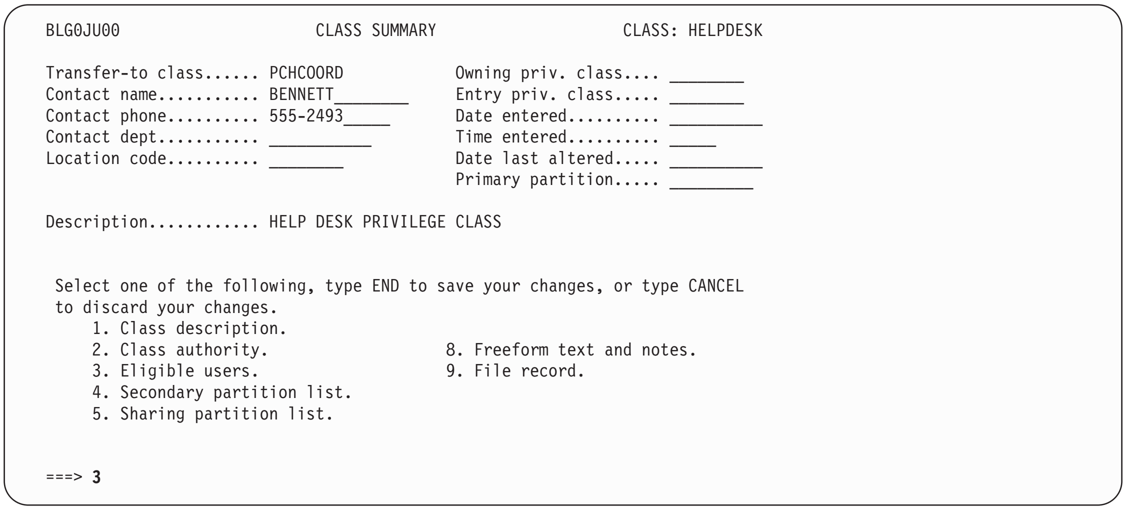
Transfer the record to your privilege class to be the owner (you are the only person besides the master privilege class who can update or delete this record).

To leave the panel, type **end** and press **Enter**.



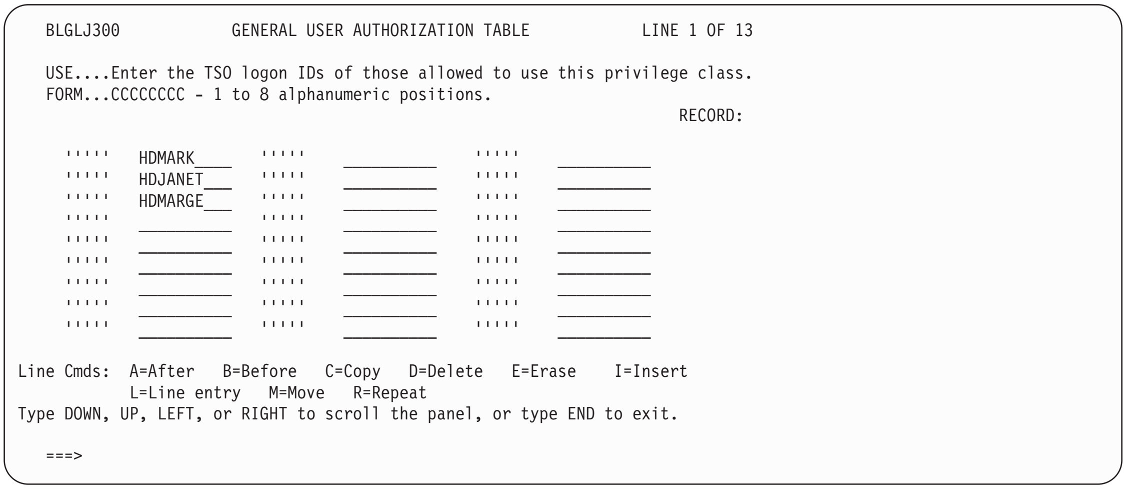
```
BLG0J100                 CLASS DESCRIPTION ENTRY              CLASS: ________

Enter privilege class data; cursor placement or input line entry allowed.

1. Privilege class name..<R> helpdesk

3. Transfer-to class........ pchcoord
4. Contact name............. bennett________
5. Contact phone............ 555-2493_____
6. Contact department....... ___________
7. Location code............ ________
8. Description...........<R> help desk privilege class____________________
9. Primary partition id..... ________



    When you finish, type END to save or CANCEL to discard any changes.





===> end
```

To add two more users, select **Eligible users**. Type **3** and press **Enter**.



```
BLG0JU00                    CLASS SUMMARY                    CLASS: HELPDESK

Transfer-to class...... PCHCOORD           Owning priv. class.... ________
Contact name........... BENNETT________    Entry priv. class..... ________
Contact phone.......... 555-2493_____      Date entered.......... __________
Contact dept........... ___________        Time entered.......... _____
Location code.......... ________           Date last altered..... __________
                                           Primary partition..... _________

Description............ HELP DESK PRIVILEGE CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.
     2. Class authority.                  8. Freeform text and notes.
     3. Eligible users.                   9. File record.
     4. Secondary partition list.
     5. Sharing partition list.


===> 3
```

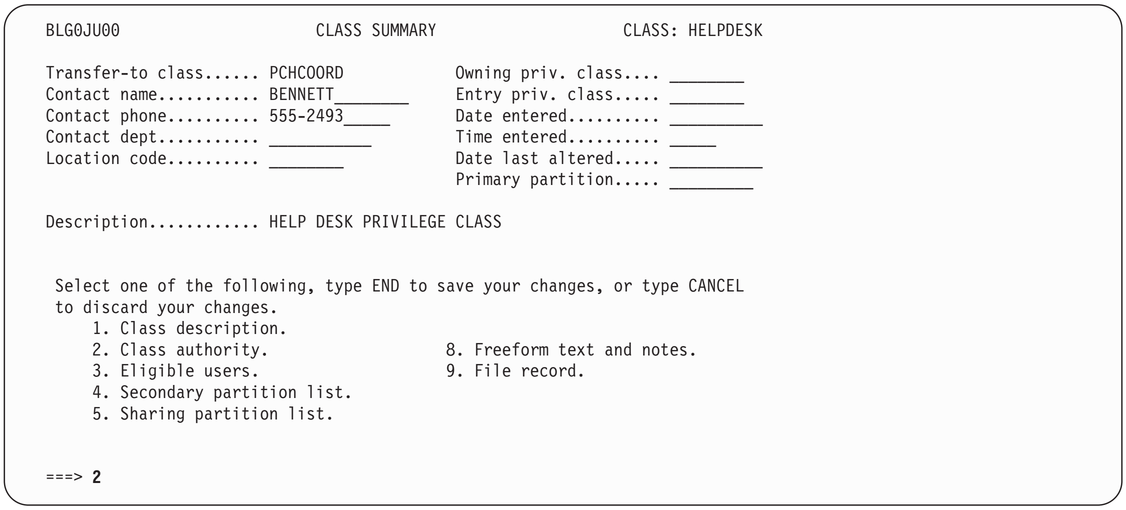
Enter the user IDs on panel BLGLJ300.



```
   BLGLJ300           GENERAL USER AUTHORIZATION TABLE           LINE 1 OF 13

   USE....Enter the TSO logon IDs of those allowed to use this privilege class.
   FORM...CCCCCCCC - 1 to 8 alphanumeric positions.
                                                                       RECORD:

     '''''   HDMARK____   '''''    __________    '''''    __________
     '''''   HDJANET___   '''''    __________    '''''    __________
     '''''   HDMARGE___   '''''    __________    '''''    __________
     '''''   __________   '''''    __________    '''''    __________
     '''''   __________   '''''    __________    '''''    __________
     '''''   __________   '''''    __________    '''''    __________
     '''''   __________   '''''    __________    '''''    __________
     '''''   __________   '''''    __________    '''''    __________
     '''''   __________   '''''    __________    '''''    __________

Line Cmds:  A=After   B=Before   C=Copy   D=Delete   E=Erase    I=Insert
            L=Line entry   M=Move   R=Repeat
Type DOWN, UP, LEFT, or RIGHT to scroll the panel, or type END to exit.

   ===>
```

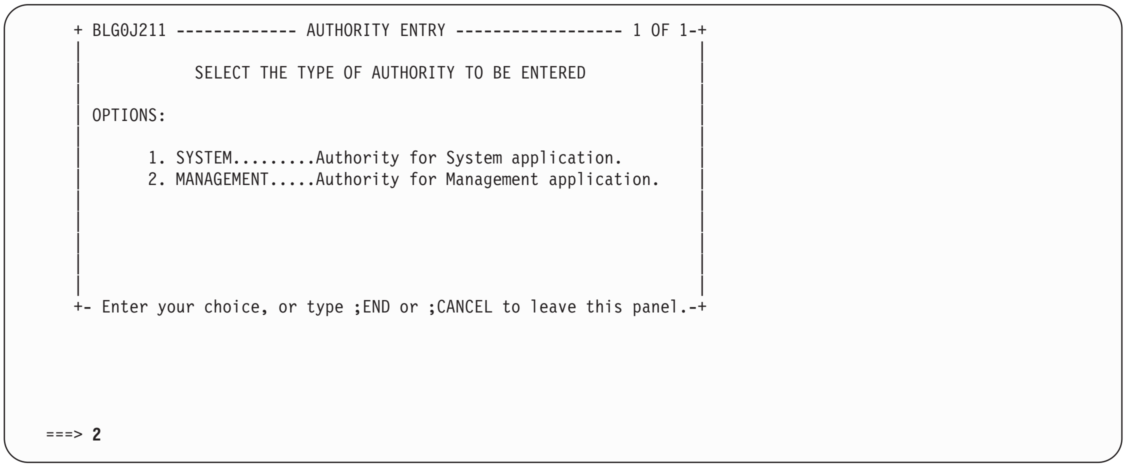
Then type **End** and press **Enter** to return to the Class Summary panel.

To give this privilege class its authority, select **Class authority**. Type **2** and press **Enter**.



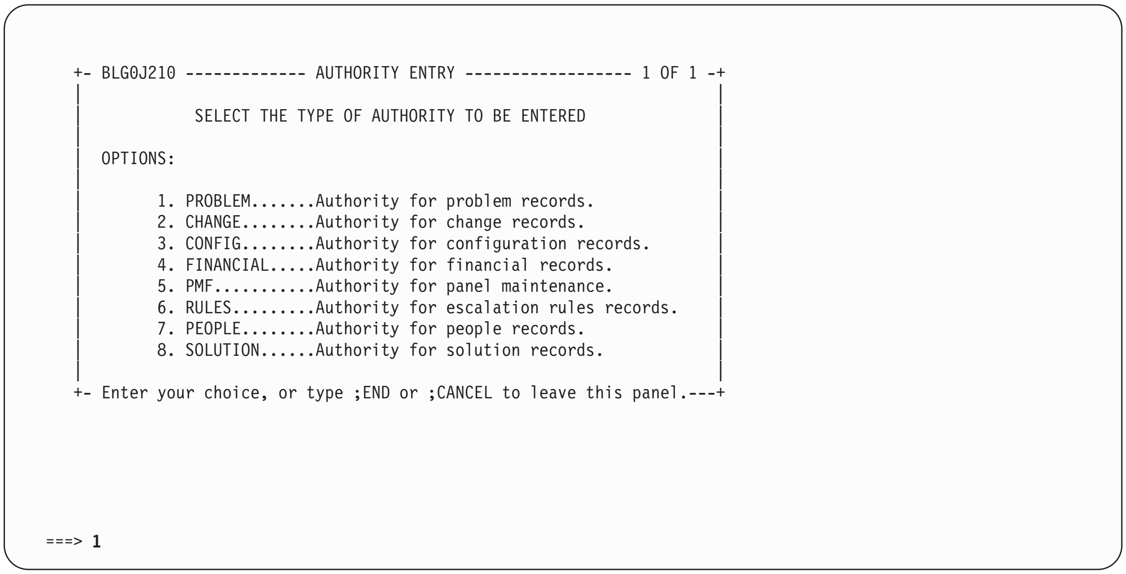
```
BLG0JU00                    CLASS SUMMARY                   CLASS: HELPDESK

Transfer-to class...... PCHCOORD           Owning priv. class.... ________
Contact name........... BENNETT________    Entry priv. class..... ________
Contact phone.......... 555-2493_____      Date entered.......... __________
Contact dept........... ___________        Time entered.......... _____
Location code.......... ________           Date last altered..... __________
                                           Primary partition..... _________

Description............ HELP DESK PRIVILEGE CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.
     2. Class authority.                 8. Freeform text and notes.
     3. Eligible users.                  9. File record.
     4. Secondary partition list.
     5. Sharing partition list.


===> 2
```

Select **Management**. Type **2** and press **Enter**.



```
   + BLG0J211 ------------- AUTHORITY ENTRY ------------------ 1 OF 1-+
   |                                                                  |
   |            SELECT THE TYPE OF AUTHORITY TO BE ENTERED            |
   |                                                                  |
   | OPTIONS:                                                         |
   |                                                                  |
   |       1. SYSTEM.........Authority for System application.       |
   |       2. MANAGEMENT.....Authority for Management application.   |
   |                                                                  |
   |                                                                  |
   |                                                                  |
   |                                                                  |
   |                                                                  |
   +- Enter your choice, or type ;END or ;CANCEL to leave this panel.-+




===> 2
```

Select **Problem** to give authority for problem records. Type **1** and press **Enter**.

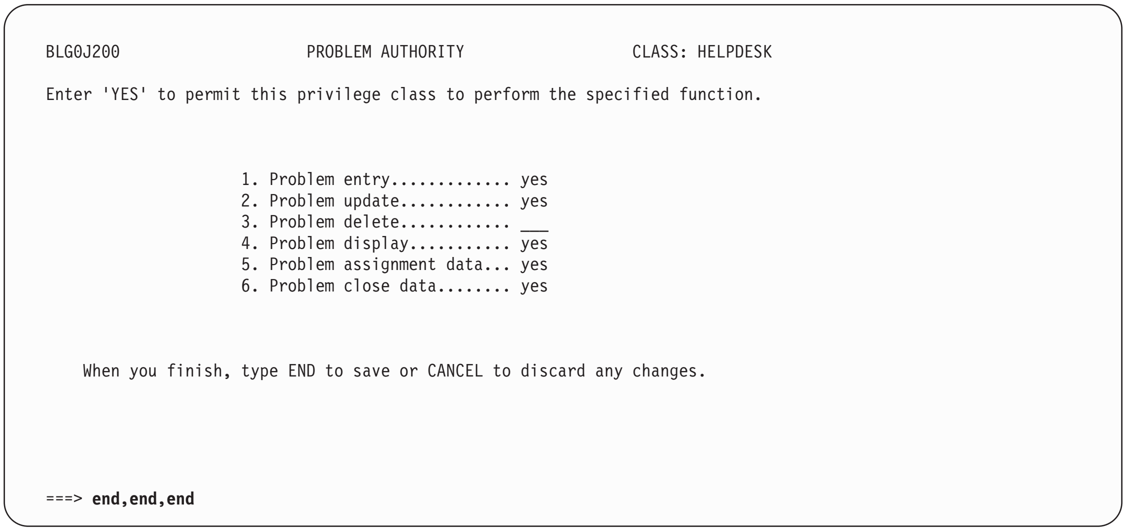


```
+- BLG0J210 ------------- AUTHORITY ENTRY ------------------ 1 OF 1 -+
|                                                                   |
|           SELECT THE TYPE OF AUTHORITY TO BE ENTERED              |
|                                                                   |
|  OPTIONS:                                                         |
|                                                                   |
|        1. PROBLEM.......Authority for problem records.            |
|        2. CHANGE........Authority for change records.             |
|        3. CONFIG........Authority for configuration records.      |
|        4. FINANCIAL.....Authority for financial records.          |
|        5. PMF...........Authority for panel maintenance.          |
|        6. RULES.........Authority for escalation rules records.   |
|        7. PEOPLE........Authority for people records.             |
|        8. SOLUTION......Authority for solution records.           |
|                                                                   |
+- Enter your choice, or type ;END or ;CANCEL to leave this panel.---+




===> 1
```

Type **yes** for problem entry, update, display, close, and assignment authorities.

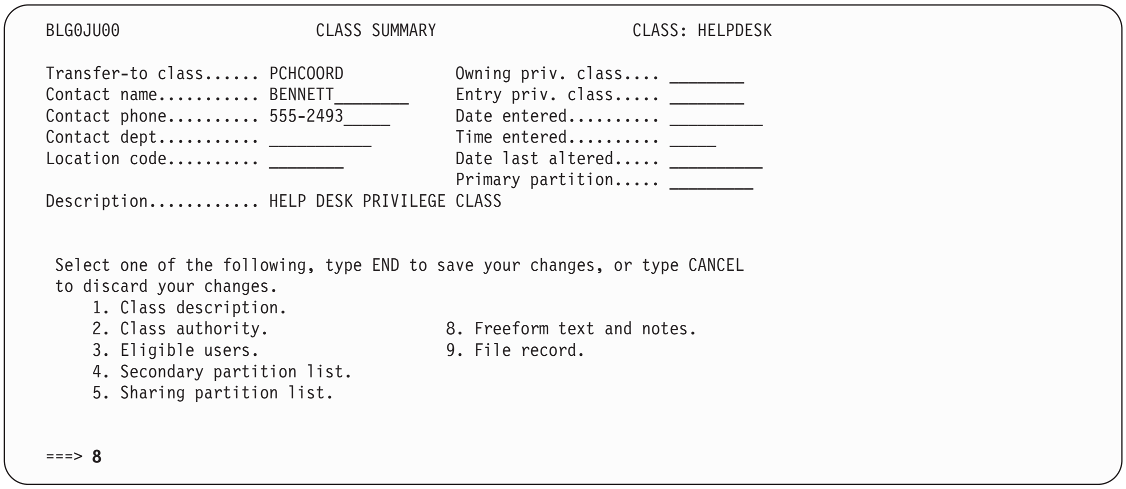
Note: Display authority is required for the other authorities.

To leave the panel and return to the Class Summary panel, type **end,end,end** and press **Enter**.



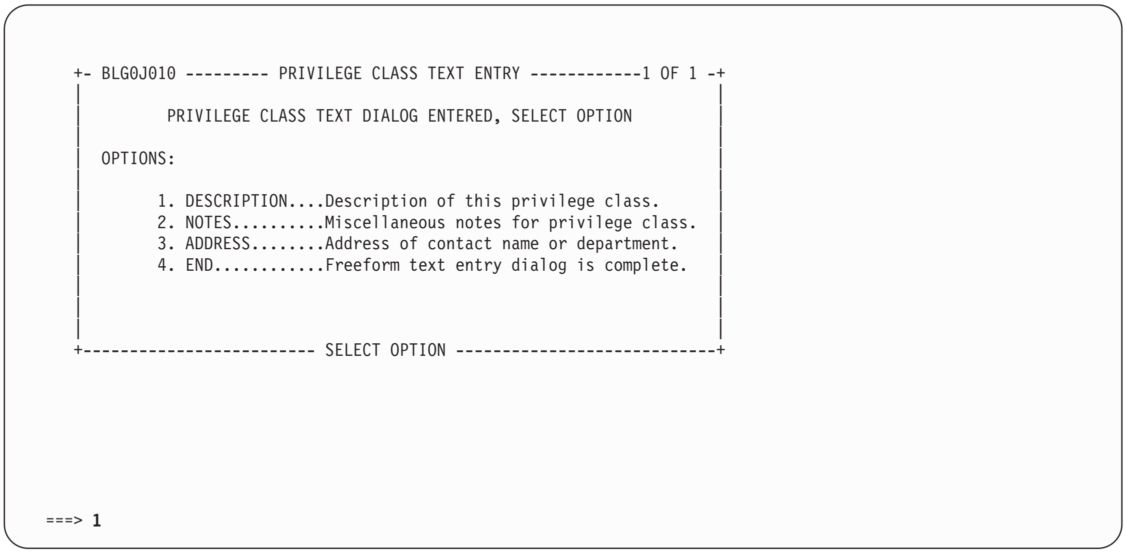
```
BLG0J200                    PROBLEM AUTHORITY                 CLASS: HELPDESK

Enter 'YES' to permit this privilege class to perform the specified function.



                     1. Problem entry............. yes
                     2. Problem update............ yes
                     3. Problem delete............ ___
                     4. Problem display........... yes
                     5. Problem assignment data... yes
                     6. Problem close data........ yes



    When you finish, type END to save or CANCEL to discard any changes.




===> end,end,end
```

To provide descriptive text for the user help desk personnel's privilege class, select **Freeform text and notes**. Type **8** and press **Enter**.



```
BLG0JU00                    CLASS SUMMARY                    CLASS: HELPDESK

Transfer-to class...... PCHCOORD           Owning priv. class.... ________
Contact name........... BENNETT________    Entry priv. class..... ________
Contact phone.......... 555-2493_____      Date entered.......... __________
Contact dept........... ___________        Time entered.......... _____
Location code.......... ________           Date last altered..... __________
                                           Primary partition..... _________
Description............ HELP DESK PRIVILEGE CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.
     2. Class authority.                  8. Freeform text and notes.
     3. Eligible users.                   9. File record.
     4. Secondary partition list.
     5. Sharing partition list.


===> 8
```

Select **Description**. Type **1** and press **Enter**.

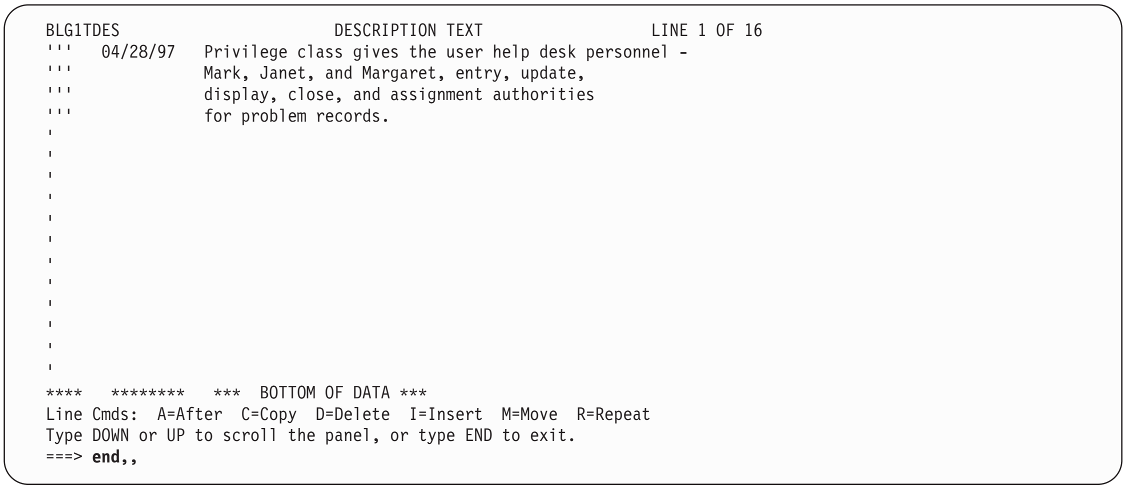


```


   +- BLG0J010 --------- PRIVILEGE CLASS TEXT ENTRY ------------1 OF 1 -+
   |                                                                    |
   |         PRIVILEGE CLASS TEXT DIALOG ENTERED, SELECT OPTION         |
   |                                                                    |
   |  OPTIONS:                                                          |
   |                                                                    |
   |        1. DESCRIPTION....Description of this privilege class.      |
   |        2. NOTES..........Miscellaneous notes for privilege class.  |
   |        3. ADDRESS........Address of contact name or department.    |
   |        4. END............Freeform text entry dialog is complete.   |
   |                                                                    |
   |                                                                    |
   |                                                                    |
   +------------------------ SELECT OPTION ----------------------------+







===> 1
```

If your profile setting for **Editor selection** is INFO, the following panel appears.

Type the description text on the full-screen text panel.

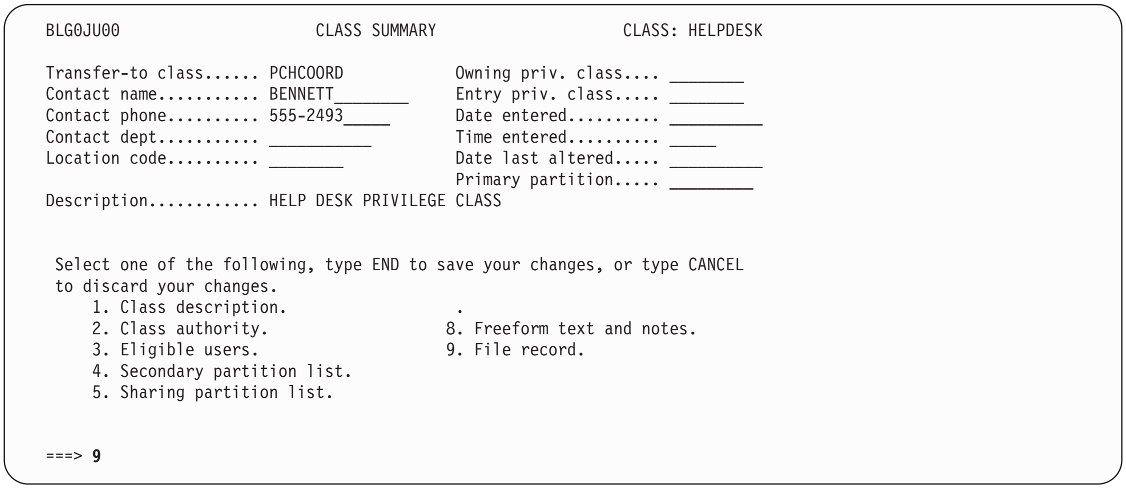
Save the text and return to the Class Summary panel. Type **end,,** and press **Enter**.



```
BLG1TDES                       DESCRIPTION TEXT                 LINE 1 OF 16
'''   04/28/97   Privilege class gives the user help desk personnel -
'''              Mark, Janet, and Margaret, entry, update,
'''              display, close, and assignment authorities
'''              for problem records.
'
'
'
'
'
'
'
'
'
'
'
'
****   ********   ***  BOTTOM OF DATA ***
Line Cmds:  A=After  C=Copy  D=Delete  I=Insert  M=Move  R=Repeat
Type DOWN or UP to scroll the panel, or type END to exit.
===> end,,
```

To save the privilege class record, select **File record**. Type **9** and press **Enter**.

Write down the privilege class name you assigned for future use. The name of the privilege class in this example is HELPDESK.

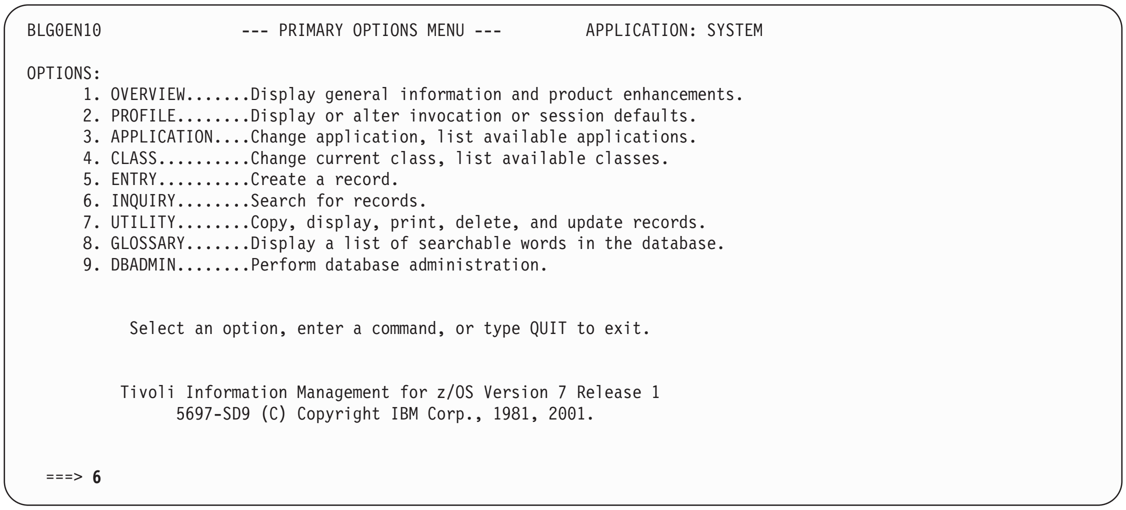


```
BLG0JU00                     CLASS SUMMARY                    CLASS: HELPDESK

Transfer-to class...... PCHCOORD           Owning priv. class.... ________
Contact name........... BENNETT________    Entry priv. class..... ________
Contact phone.......... 555-2493_____      Date entered.......... __________
Contact dept........... ___________        Time entered.......... _____
Location code.......... ________           Date last altered..... __________
                                           Primary partition..... _________
Description............ HELP DESK PRIVILEGE CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.                   .
     2. Class authority.                  8. Freeform text and notes.
     3. Eligible users.                   9. File record.
     4. Secondary partition list.
     5. Sharing partition list.


===> 9
```

Determining All Privilege Class Names

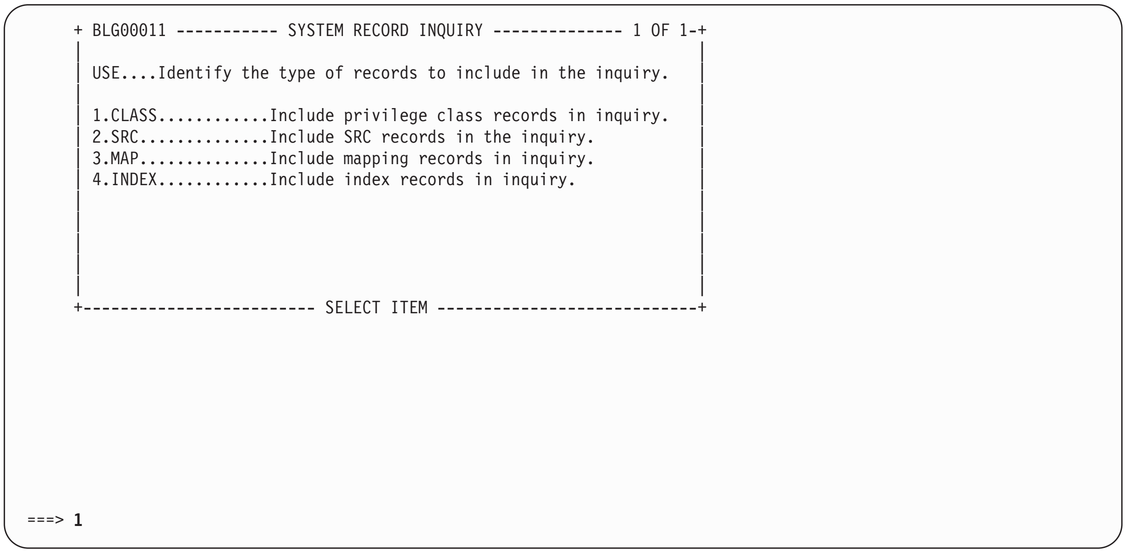
Mark, Janet, and Margaret at the user help desk require an updated report of all privilege classes and the users assigned to them. You perform a search to print the required report.

From the Primary Options Menu for the System application, select **Inquiry**. Type **6** and press **Enter**.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:
      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


  ===> 6
```

Select **Class**. Type **1** and press **Enter**.

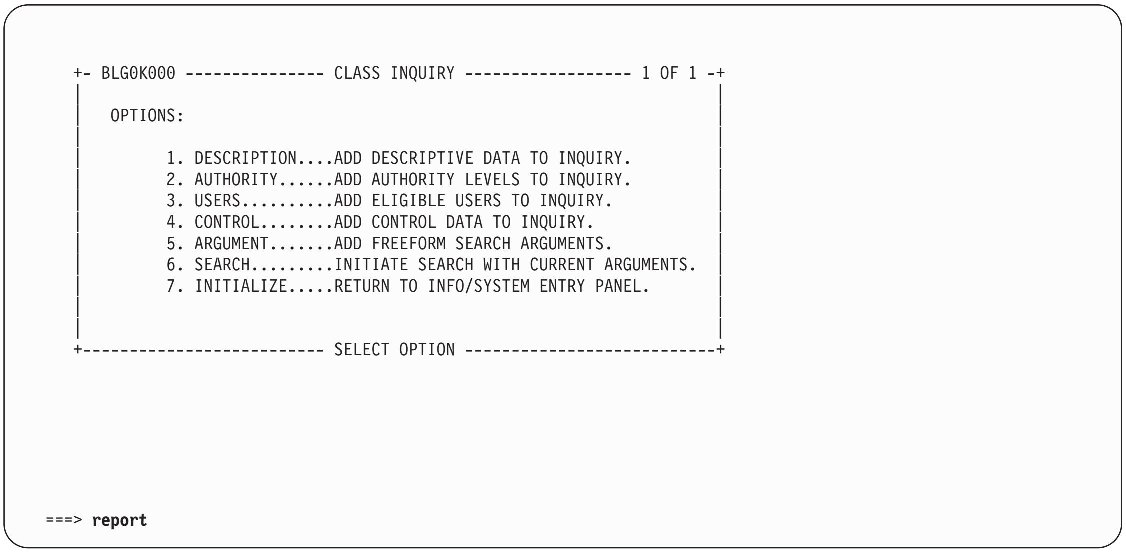


```
       + BLG00011 ----------- SYSTEM RECORD INQUIRY -------------- 1 OF 1-+
       |                                                                 |
       | USE....Identify the type of records to include in the inquiry.  |
       |                                                                 |
       | 1.CLASS...........Include privilege class records in inquiry.   |
       | 2.SRC.............Include SRC records in the inquiry.            |
       | 3.MAP.............Include mapping records in inquiry.            |
       | 4.INDEX...........Include index records in inquiry.             |
       |                                                                 |
       |                                                                 |
       |                                                                 |
       |                                                                 |
       |                                                                 |
       +------------------------ SELECT ITEM ----------------------------+







===> 1
```

This panel appears if you are using the structured search method of searching. If you are using the quick search method of searching, panel BLG0K190 appears. Refer to the *Tivoli Information Management for z/OS User's Guide* for information about methods of searching.

You want all privilege classes included; therefore, you need not further qualify the search argument.

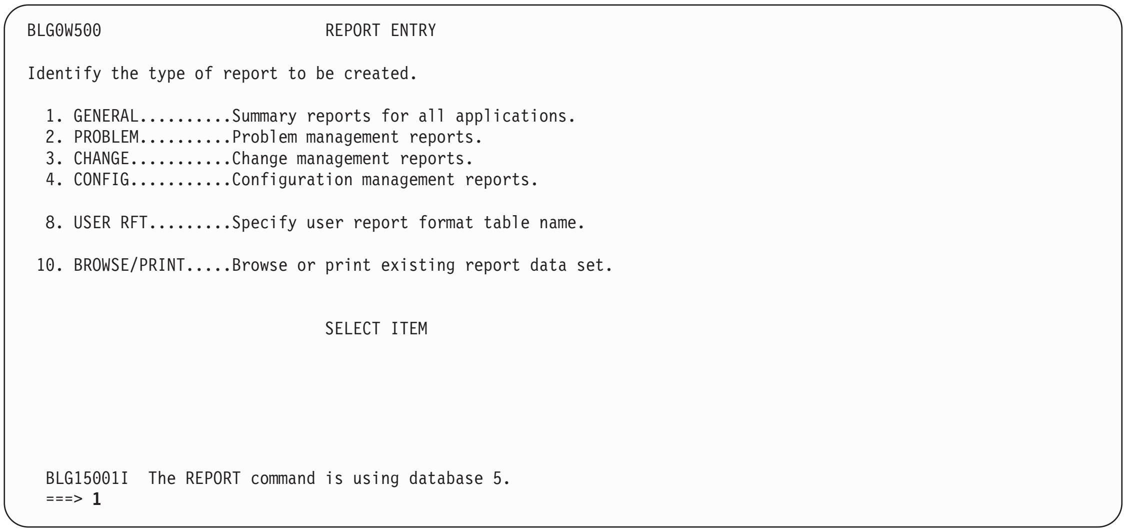
To get the report, issue the REPORT command; type **report** and press **Enter**.



```
+- BLG0K000 --------------- CLASS INQUIRY ------------------ 1 OF 1 -+
|                                                                    |
|   OPTIONS:                                                         |
|                                                                    |
|         1. DESCRIPTION....ADD DESCRIPTIVE DATA TO INQUIRY.          |
|         2. AUTHORITY......ADD AUTHORITY LEVELS TO INQUIRY.          |
|         3. USERS..........ADD ELIGIBLE USERS TO INQUIRY.            |
|         4. CONTROL........ADD CONTROL DATA TO INQUIRY.              |
|         5. ARGUMENT.......ADD FREEFORM SEARCH ARGUMENTS.            |
|         6. SEARCH.........INITIATE SEARCH WITH CURRENT ARGUMENTS.   |
|         7. INITIALIZE.....RETURN TO INFO/SYSTEM ENTRY PANEL.        |
|                                                                    |
|                                                                    |
+-------------------------- SELECT OPTION ---------------------------+




===> report
```

The Page Summary report provides the data needed by the user help desk. Because the report you want is common to all record types, you want a report in the general category.

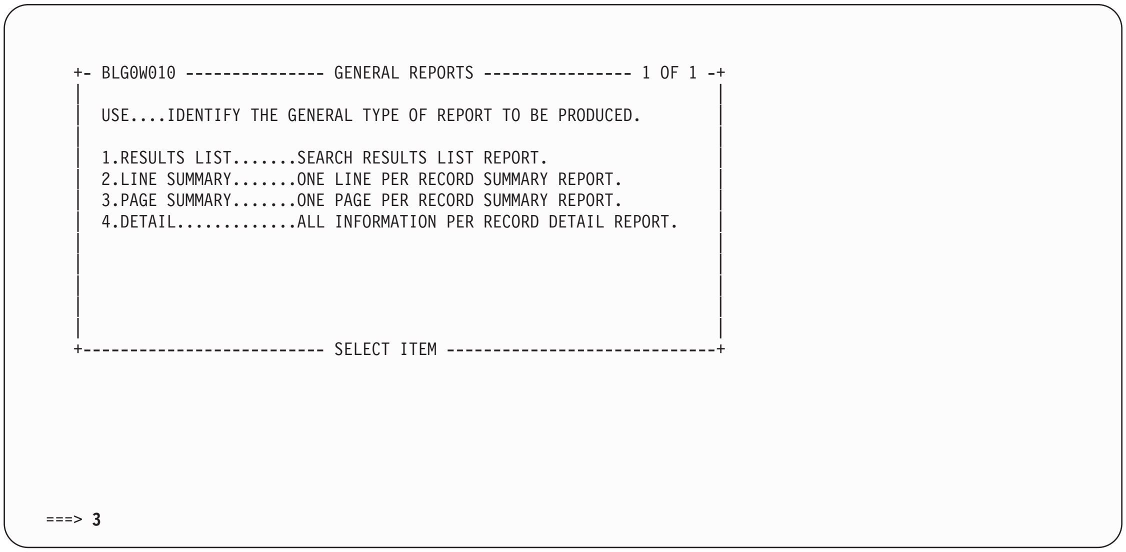
Select **General**. Type **1** and press **Enter**.



```
BLG0W500                        REPORT ENTRY

Identify the type of report to be created.

  1. GENERAL..........Summary reports for all applications.
  2. PROBLEM..........Problem management reports.
  3. CHANGE...........Change management reports.
  4. CONFIG...........Configuration management reports.

  8. USER RFT.........Specify user report format table name.

 10. BROWSE/PRINT.....Browse or print existing report data set.


                                SELECT ITEM






  BLG15001I  The REPORT command is using database 5.
  ===> 1
```

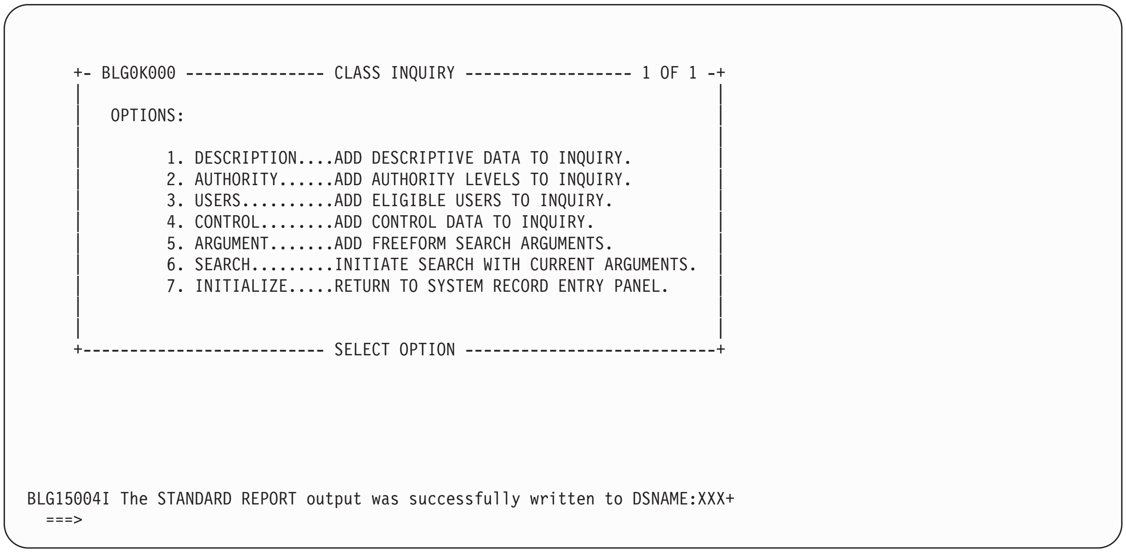
Now, select **Page Summary**. Type **3** and press **Enter**.

If your user profile does not have output destination settings already set up, the next panel you see asks you to enter output destination information. For this example, assume you already have output destination information set up in your profile. Refer to the *Tivoli Information Management for z/OS User's Guide* for more information on output destinations.



```
    +- BLG0W010 --------------- GENERAL REPORTS ---------------- 1 OF 1 -+
    |                                                                    |
    |  USE....IDENTIFY THE GENERAL TYPE OF REPORT TO BE PRODUCED.        |
    |                                                                    |
    |  1.RESULTS LIST.......SEARCH RESULTS LIST REPORT.                  |
    |  2.LINE SUMMARY.......ONE LINE PER RECORD SUMMARY REPORT.          |
    |  3.PAGE SUMMARY.......ONE PAGE PER RECORD SUMMARY REPORT.          |
    |  4.DETAIL.............ALL INFORMATION PER RECORD DETAIL REPORT.    |
    |                                                                    |
    |                                                                    |
    |                                                                    |
    |                                                                    |
    |                                                                    |
    +-------------------------- SELECT ITEM -----------------------------+




  ===> 3
```

You return to panel BLG0K000 where a message tells you that your report was successfully written.

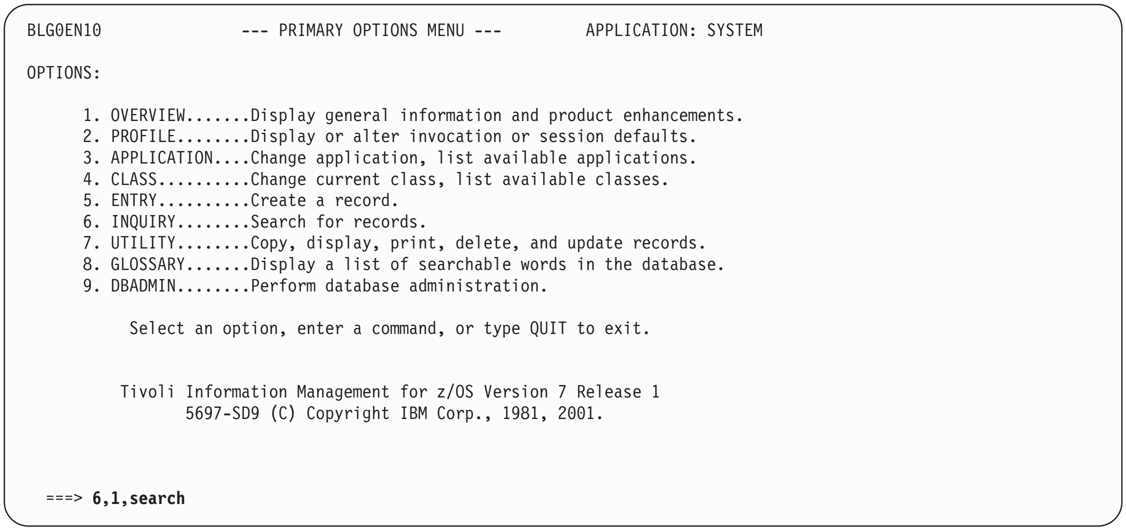


```
    +- BLG0K000 --------------- CLASS INQUIRY ------------------ 1 OF 1 -+
    |                                                                    |
    |   OPTIONS:                                                         |
    |                                                                    |
    |         1. DESCRIPTION....ADD DESCRIPTIVE DATA TO INQUIRY.         |
    |         2. AUTHORITY......ADD AUTHORITY LEVELS TO INQUIRY.         |
    |         3. USERS..........ADD ELIGIBLE USERS TO INQUIRY.           |
    |         4. CONTROL........ADD CONTROL DATA TO INQUIRY.             |
    |         5. ARGUMENT.......ADD FREEFORM SEARCH ARGUMENTS.           |
    |         6. SEARCH.........INITIATE SEARCH WITH CURRENT ARGUMENTS.  |
    |         7. INITIALIZE.....RETURN TO SYSTEM RECORD ENTRY PANEL.     |
    |                                                                    |
    |                                                                    |
    +-------------------------- SELECT OPTION ---------------------------+




BLG15004I The STANDARD REPORT output was successfully written to DSNAME:XXX+
  ===>
```

Displaying Authorities for a Privilege Class

In this scenario, you need to know what authorities the help desk group has. However, you do not remember the name of the privilege class. First, perform a search to identify the class and then display the class.

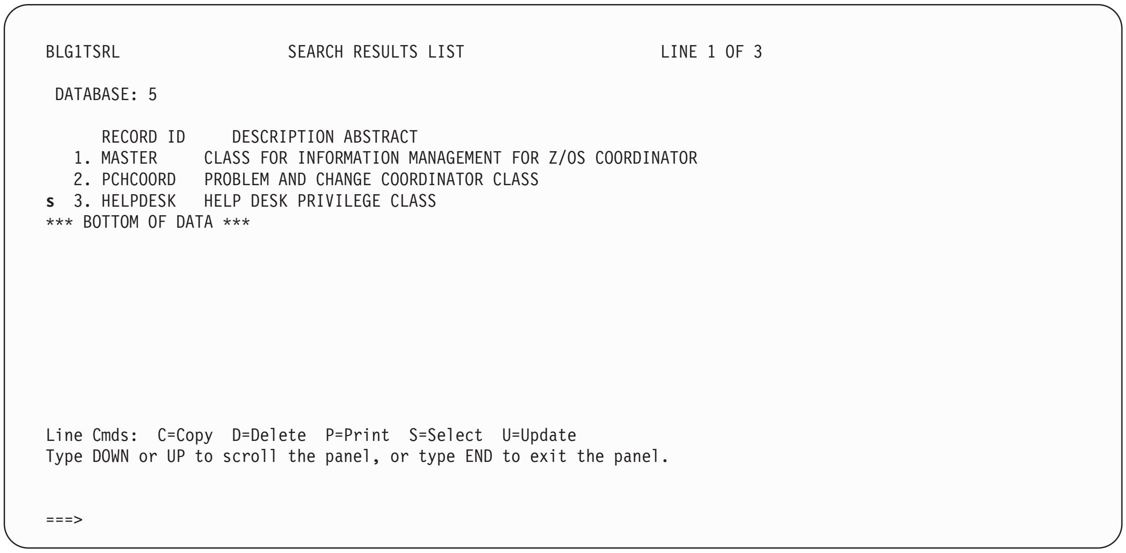
From the Primary Options Menu for the System application, perform a search for privilege class records. Type **6,1,search** and press **Enter**.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.

           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.



  ===> 6,1,search
```

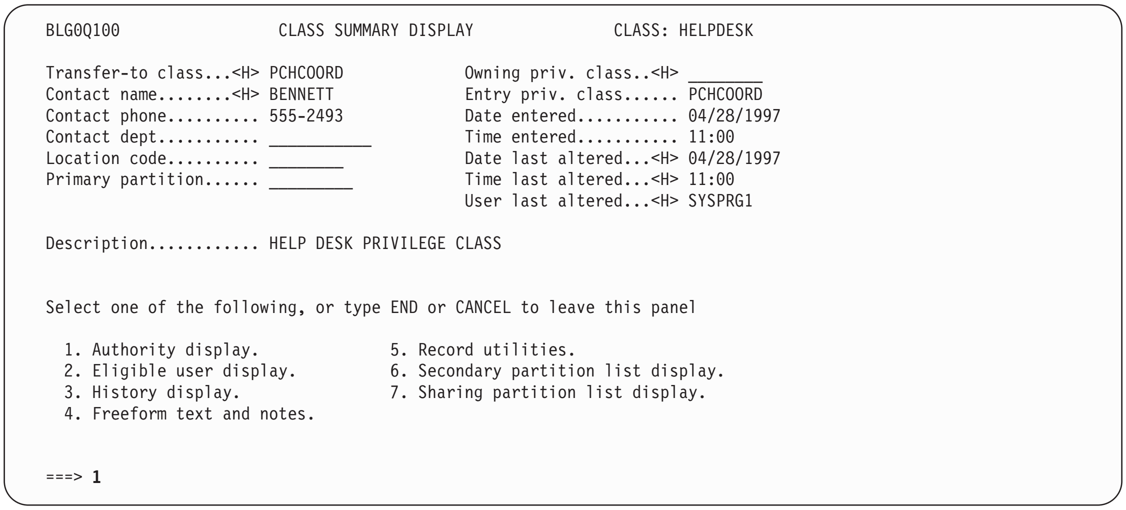
The first page of the search results list contains the class you want, record 3.

Select the class for display. In the line command area for record 3, type **s** and press **Enter**.



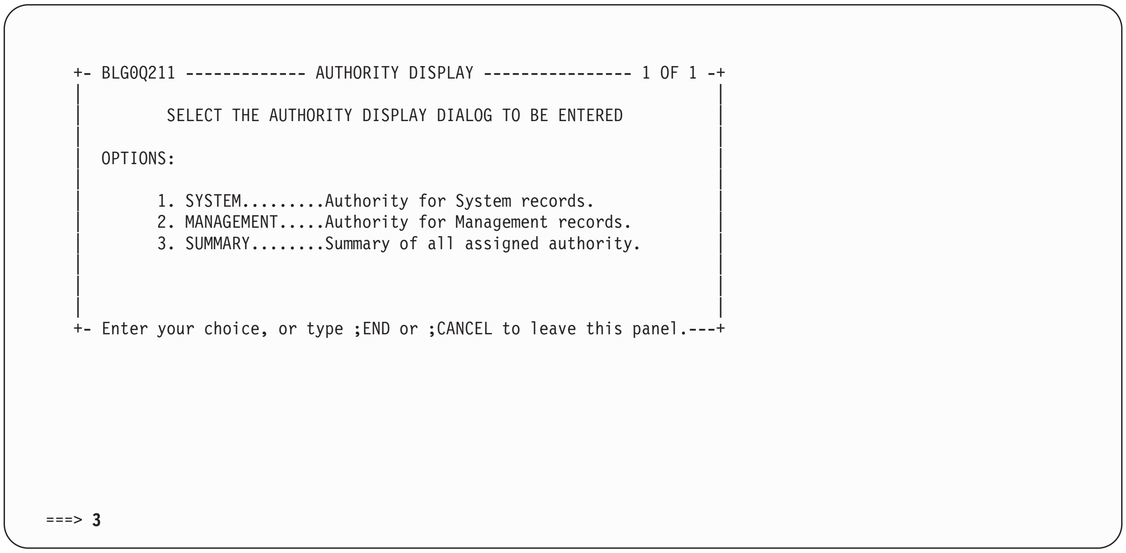
```
  BLG1TSRL                SEARCH RESULTS LIST                  LINE 1 OF 3

   DATABASE: 5

        RECORD ID     DESCRIPTION ABSTRACT
     1. MASTER     CLASS FOR INFORMATION MANAGEMENT FOR Z/OS COORDINATOR
     2. PCHCOORD   PROBLEM AND CHANGE COORDINATOR CLASS
  s  3. HELPDESK   HELP DESK PRIVILEGE CLASS
  *** BOTTOM OF DATA ***








  Line Cmds:  C=Copy  D=Delete  P=Print  S=Select  U=Update
  Type DOWN or UP to scroll the panel, or type END to exit the panel.


  ===>
```

Select **Authority display**. Type **1** and press **Enter**.



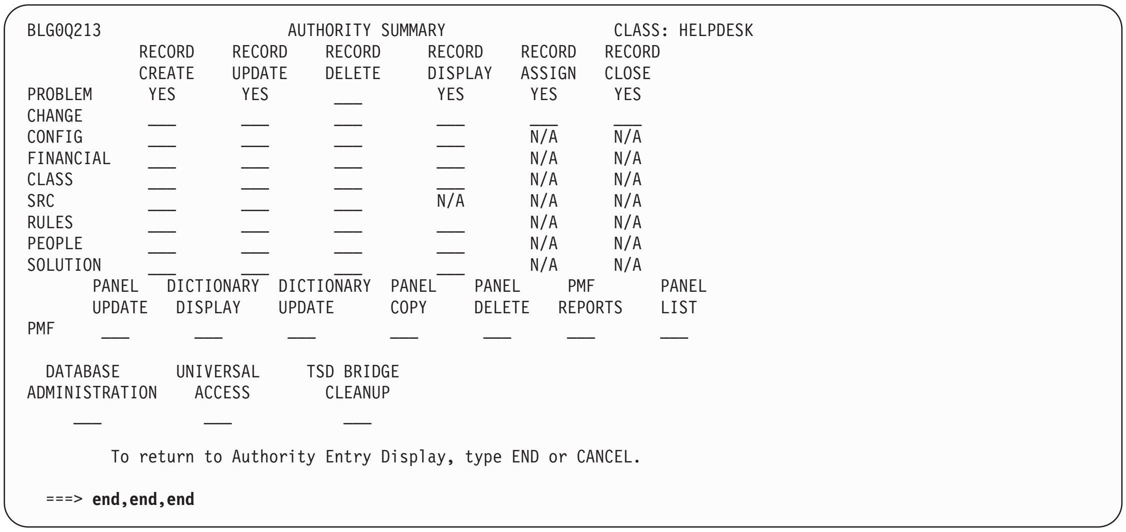
```
BLG0Q100                 CLASS SUMMARY DISPLAY               CLASS: HELPDESK

Transfer-to class...<H> PCHCOORD            Owning priv. class..<H> ________
Contact name........<H> BENNETT             Entry priv. class...... PCHCOORD
Contact phone.......... 555-2493            Date entered........... 04/28/1997
Contact dept........... ___________         Time entered........... 11:00
Location code.......... ________            Date last altered...<H> 04/28/1997
Primary partition...... _________           Time last altered...<H> 11:00
                                            User last altered...<H> SYSPRG1

Description............ HELP DESK PRIVILEGE CLASS


Select one of the following, or type END or CANCEL to leave this panel

  1. Authority display.             5. Record utilities.
  2. Eligible user display.         6. Secondary partition list display.
  3. History display.               7. Sharing partition list display.
  4. Freeform text and notes.


===> 1
```

Because you want to see all of the authorities this record has, select **Summary**. Type **3** and press **Enter**.



```
   +- BLG0Q211 ------------- AUTHORITY DISPLAY ---------------- 1 OF 1 -+
   |                                                                    |
   |         SELECT THE AUTHORITY DISPLAY DIALOG TO BE ENTERED          |
   |                                                                    |
   |  OPTIONS:                                                          |
   |                                                                    |
   |        1. SYSTEM.........Authority for System records.             |
   |        2. MANAGEMENT.....Authority for Management records.         |
   |        3. SUMMARY........Summary of all assigned authority.        |
   |                                                                    |
   |                                                                    |
   |                                                                    |
   +- Enter your choice, or type ;END or ;CANCEL to leave this panel.---+







===> 3
```

Now a summary of all authorities appears. To return to the search results list, type **end,end,end** and press **Enter**.

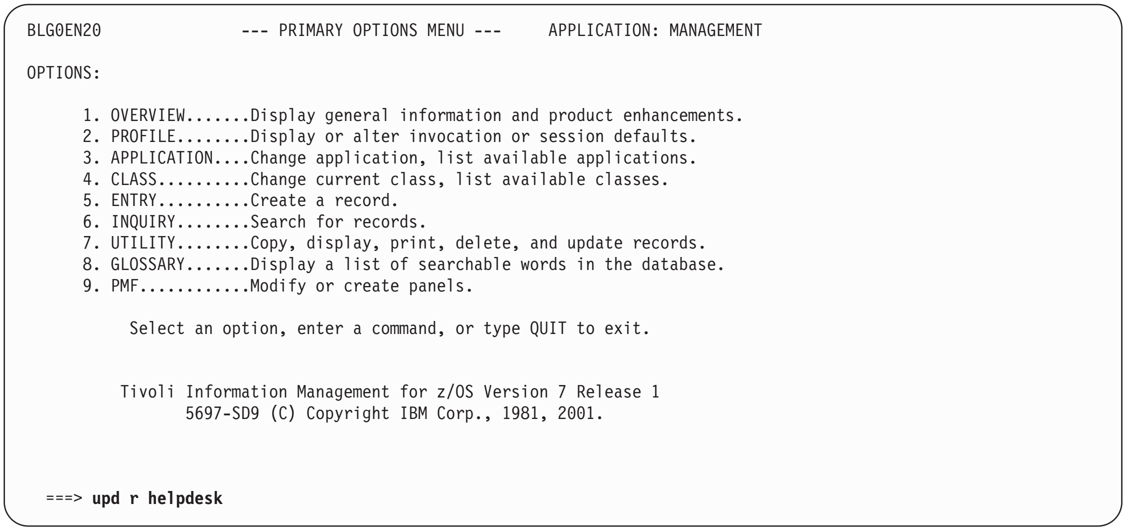


```
BLG0Q213                    AUTHORITY SUMMARY                  CLASS: HELPDESK
            RECORD    RECORD    RECORD     RECORD    RECORD   RECORD
            CREATE    UPDATE    DELETE     DISPLAY   ASSIGN   CLOSE
PROBLEM      YES       YES       ___        YES       YES      YES
CHANGE       ___       ___       ___        ___       ___      ___
CONFIG       ___       ___       ___        ___       N/A      N/A
FINANCIAL    ___       ___       ___        ___       N/A      N/A
CLASS        ___       ___       ___        ___       N/A      N/A
SRC          ___       ___       ___        N/A       N/A      N/A
RULES        ___       ___       ___        ___       N/A      N/A
PEOPLE       ___       ___       ___        ___       N/A      N/A
SOLUTION     ___       ___       ___        ___       N/A      N/A
       PANEL   DICTIONARY  DICTIONARY  PANEL    PANEL     PMF       PANEL
       UPDATE   DISPLAY    UPDATE      COPY     DELETE   REPORTS    LIST
PMF     ___       ___       ___        ___       ___      ___       ___

  DATABASE      UNIVERSAL     TSD BRIDGE
ADMINISTRATION    ACCESS        CLEANUP

     ___           ___            ___

         To return to Authority Entry Display, type END or CANCEL.

  ===> end,end,end
```

Deleting One Member of a Privilege Class

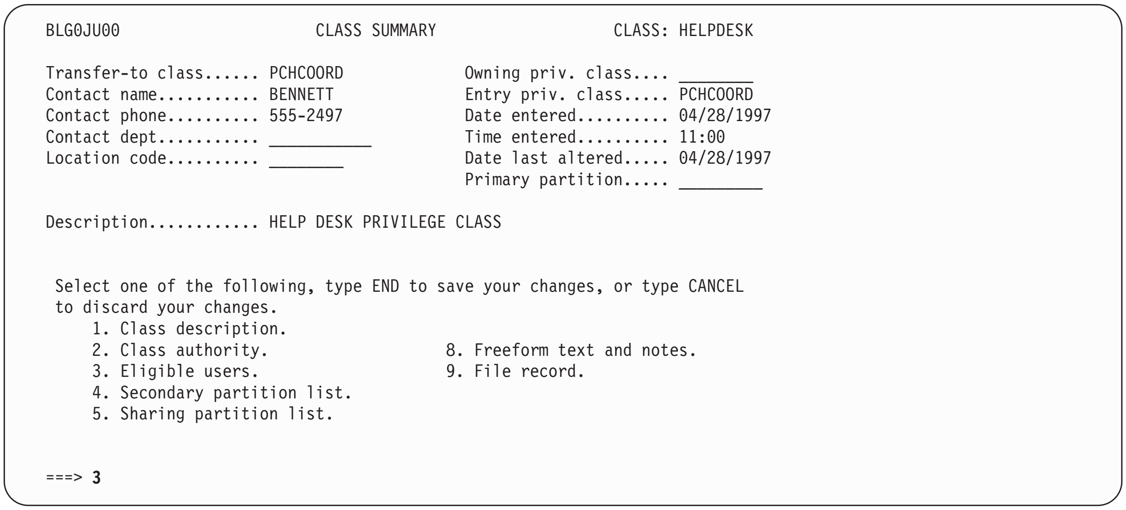
In this scenario, three employees are members of the HELPDESK privilege class. One of them, Margaret, no longer requires access to Tivoli Information Management for z/OS. Margaret's user ID is HDMARGE.

You can start from the Primary Options Menu for the Management application and type **UPD R HELPDESK** (to update using the privilege class identifier) and press **Enter**.



```
BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.

           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.



  ===> upd r helpdesk
```

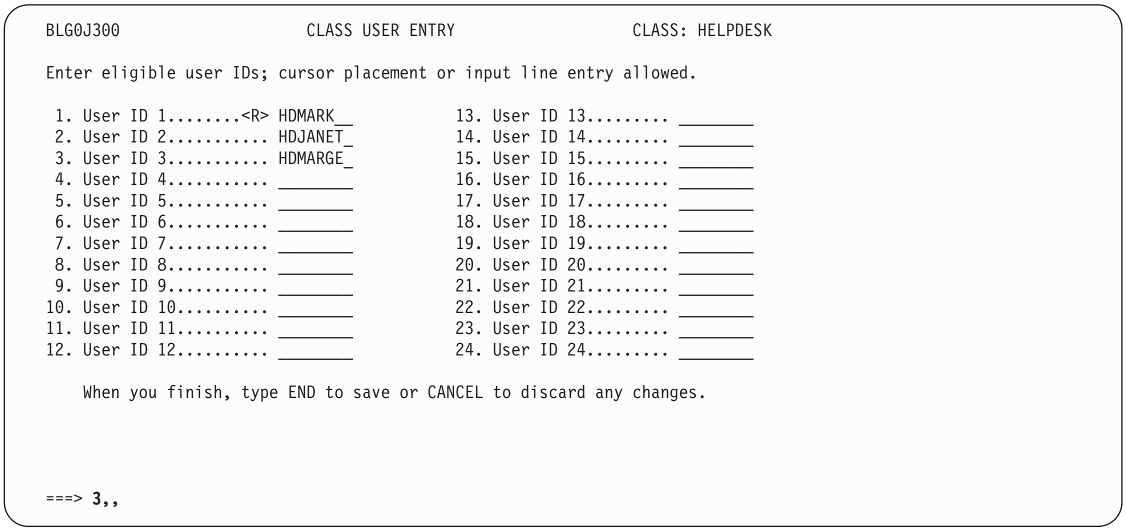
To modify eligible user information, select **Eligible users**. Type **3** and press **Enter**.



```
BLG0JU00                    CLASS SUMMARY                  CLASS: HELPDESK

Transfer-to class...... PCHCOORD           Owning priv. class.... ________
Contact name........... BENNETT            Entry priv. class..... PCHCOORD
Contact phone.......... 555-2497           Date entered.......... 04/28/1997
Contact dept........... ___________        Time entered.......... 11:00
Location code.......... ________           Date last altered..... 04/28/1997
                                           Primary partition..... _________

Description............ HELP DESK PRIVILEGE CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.
     2. Class authority.                 8. Freeform text and notes.
     3. Eligible users.                  9. File record.
     4. Secondary partition list.
     5. Sharing partition list.


===> 3
```

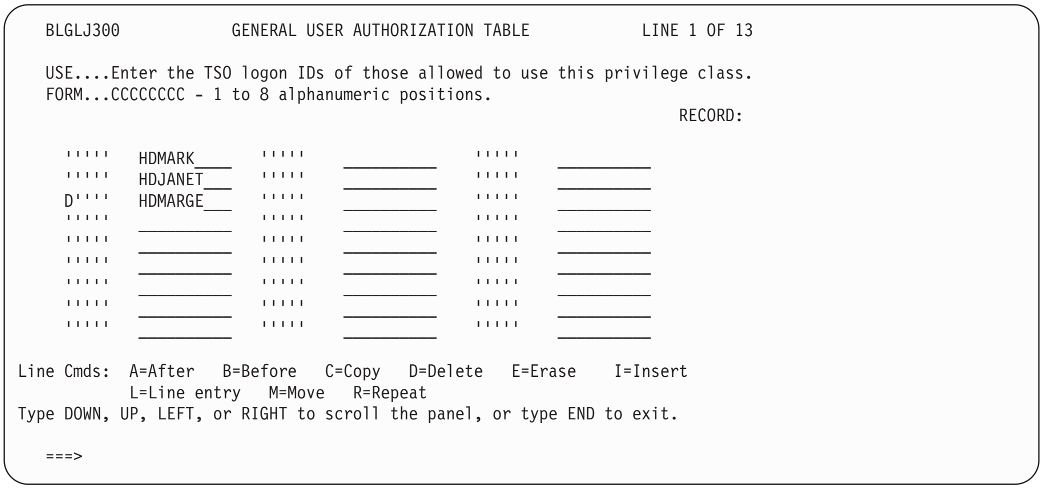
To delete Margaret’s user ID, type **3,,** and press **Enter**.

To leave the panel, type **end** and press **Enter**.



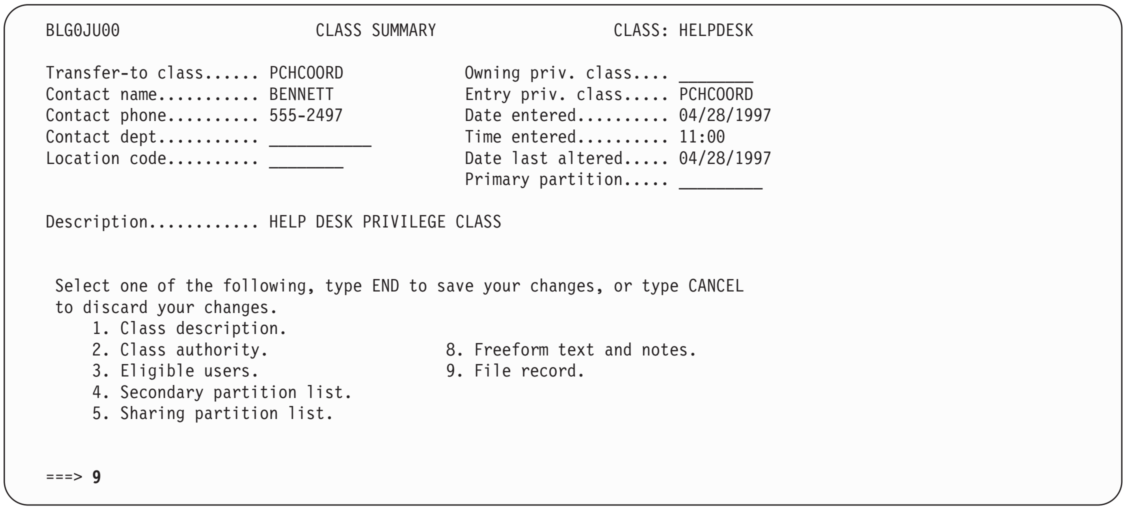
```
BLG0J300                   CLASS USER ENTRY                 CLASS: HELPDESK

Enter eligible user IDs; cursor placement or input line entry allowed.

 1. User ID 1........<R> HDMARK__          13. User ID 13......... ________
 2. User ID 2........... HDJANET_          14. User ID 14......... ________
 3. User ID 3........... HDMARGE_          15. User ID 15......... ________
 4. User ID 4........... ________          16. User ID 16......... ________
 5. User ID 5........... ________          17. User ID 17......... ________
 6. User ID 6........... ________          18. User ID 18......... ________
 7. User ID 7........... ________          19. User ID 19......... ________
 8. User ID 8........... ________          20. User ID 20......... ________
 9. User ID 9........... ________          21. User ID 21......... ________
10. User ID 10.......... ________          22. User ID 22......... ________
11. User ID 11.......... ________          23. User ID 23......... ________
12. User ID 12.......... ________          24. User ID 24......... ________

    When you finish, type END to save or CANCEL to discard any changes.




===> 3,,
```

Note: If you are removing a user from a record created in Tivoli Information Management for z/OS Version 7.1, or if the record was created before Tivoli Information Management for z/OS Version 7.1 and you have run TSX BLGTPRIV, then you would instead use the following panel to remove the user.



```
   BLGLJ300           GENERAL USER AUTHORIZATION TABLE           LINE 1 OF 13

   USE....Enter the TSO logon IDs of those allowed to use this privilege class.
   FORM...CCCCCCCC - 1 to 8 alphanumeric positions.
                                                                 RECORD:

     '''''   HDMARK____    '''''    __________    '''''    __________
     '''''   HDJANET___    '''''    __________    '''''    __________
     D''''   HDMARGE___    '''''    __________    '''''    __________
     '''''   __________    '''''    __________    '''''    __________
     '''''   __________    '''''    __________    '''''    __________
     '''''   __________    '''''    __________    '''''    __________
     '''''   __________    '''''    __________    '''''    __________
     '''''   __________    '''''    __________    '''''    __________
     '''''   __________    '''''    __________    '''''    __________

Line Cmds:  A=After   B=Before   C=Copy   D=Delete   E=Erase    I=Insert
            L=Line entry   M=Move   R=Repeat
Type DOWN, UP, LEFT, or RIGHT to scroll the panel, or type END to exit.

   ===>
```

To save the updated privilege class record, select **File record**. Type **9** and press **Enter**.



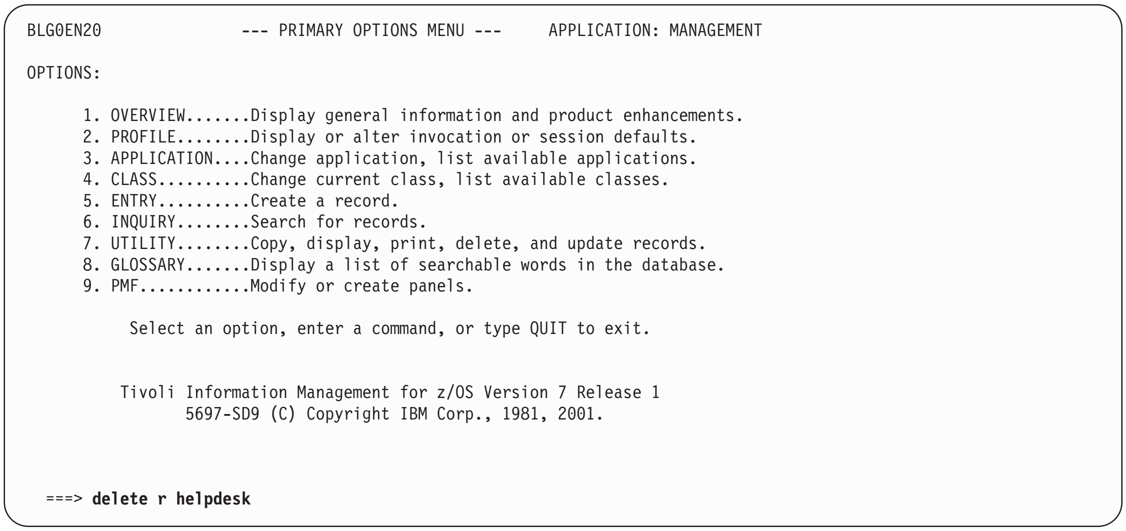
```
BLG0JU00                    CLASS SUMMARY                  CLASS: HELPDESK

Transfer-to class...... PCHCOORD            Owning priv. class.... ________
Contact name........... BENNETT             Entry priv. class..... PCHCOORD
Contact phone.......... 555-2497            Date entered.......... 04/28/1997
Contact dept........... ___________         Time entered.......... 11:00
Location code.......... ________            Date last altered..... 04/28/1997
                                            Primary partition..... _________

Description............ HELP DESK PRIVILEGE CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.
     2. Class authority.                  8. Freeform text and notes.
     3. Eligible users.                   9. File record.
     4. Secondary partition list.
     5. Sharing partition list.


===> 9
```

Deleting a Privilege Class

In this scenario, your company is reorganizing its operations and wants to eliminate the Help Desk department.

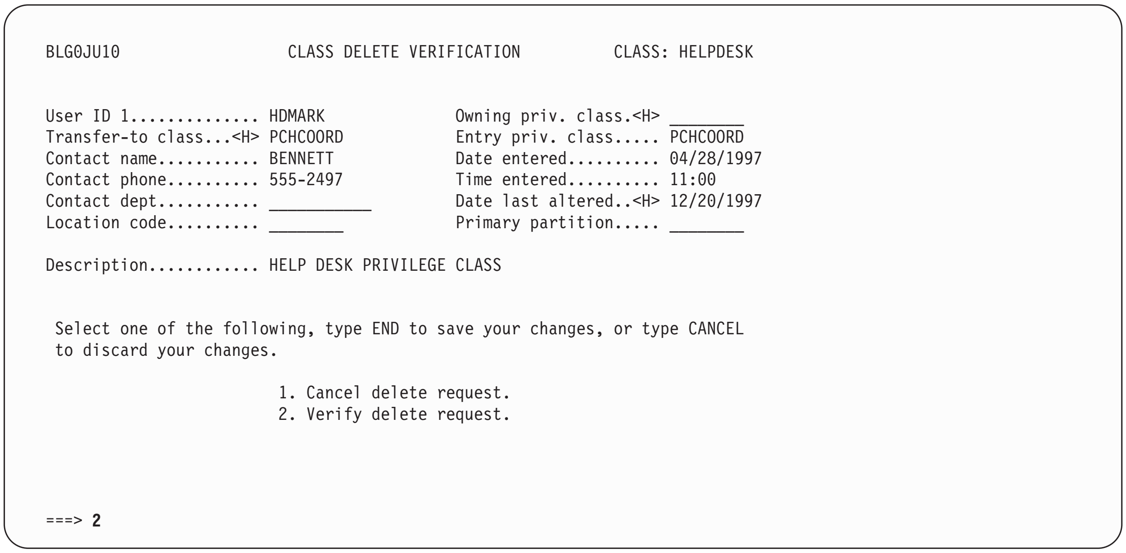
You have delete authority for the HELPDESK privilege class. All records that are owned by privilege class ID HELPDESK have been transferred or deleted. You know the privilege class record identifier, and you are currently using the Management application, so you begin at the Primary Options Menu after issuing the INITIALIZE command. It is not necessary that you start from this panel. You can delete by command from any panel.

From the Primary Options Menu, issue the DELETE command with the record ID. Type **delete r helpdesk** and press **Enter**.



```
BLG0EN20               --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.

           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.



  ===> delete r helpdesk
```

Verify the delete. Select **Verify delete request**. Type **2** and press **Enter**. A message tells you the record is deleted.



```
BLG0JU10                 CLASS DELETE VERIFICATION         CLASS: HELPDESK


User ID 1............. HDMARK             Owning priv. class.<H> ________
Transfer-to class...<H> PCHCOORD          Entry priv. class..... PCHCOORD
Contact name.......... BENNETT            Date entered.......... 04/28/1997
Contact phone......... 555-2497           Time entered.......... 11:00
Contact dept.......... ___________        Date last altered..<H> 12/20/1997
Location code......... ________           Primary partition..... ________

Description........... HELP DESK PRIVILEGE CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.

                        1. Cancel delete request.
                        2. Verify delete request.




===> 2
```

3

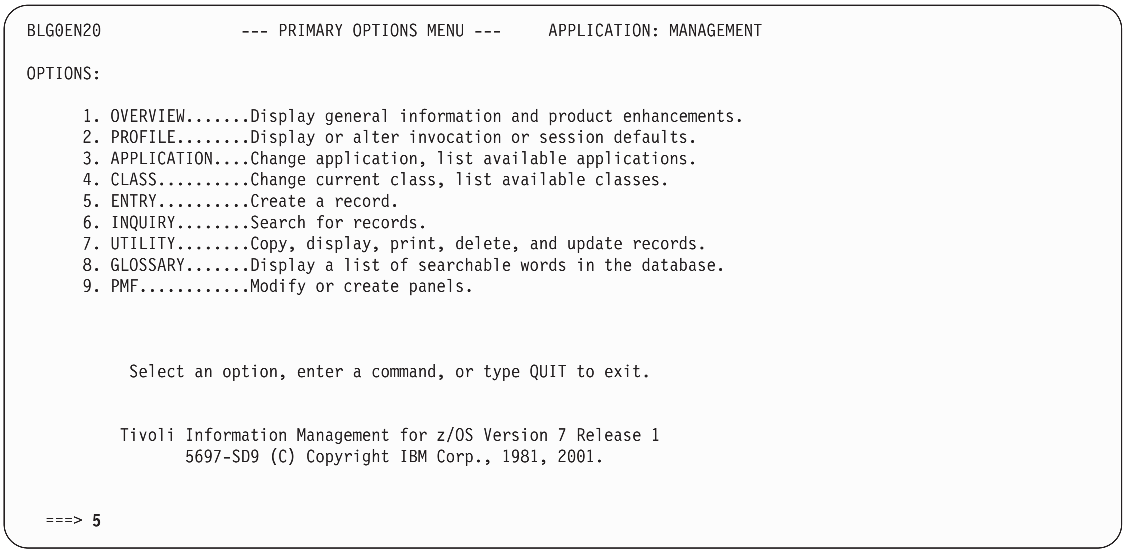
People Records

Both the Tivoli Information Management for z/OS Desktop (described in the *Tivoli Information Management for z/OS Desktop User's Guide*) and the Tivoli Service Desk Bridge (described in the *Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications*) use records of information about individual people. In order to maintain this information, a record type called the people record must be created for each person.

Creating People Records

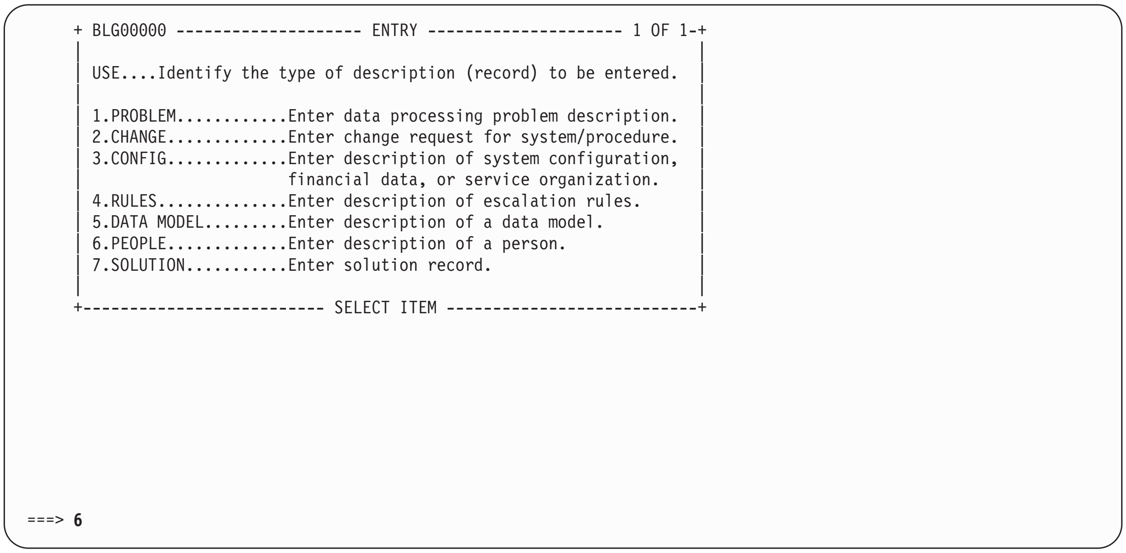
You must have the appropriate authority level in order to create people records. If you receive an error message indicating that you do not have the authority to perform this function, you will need to contact your program administrator so that you can be authorized for this function.

To create a people record, on BLG0EN20, the Primary Options Menu for the Management application, type **5** and press **Enter**.



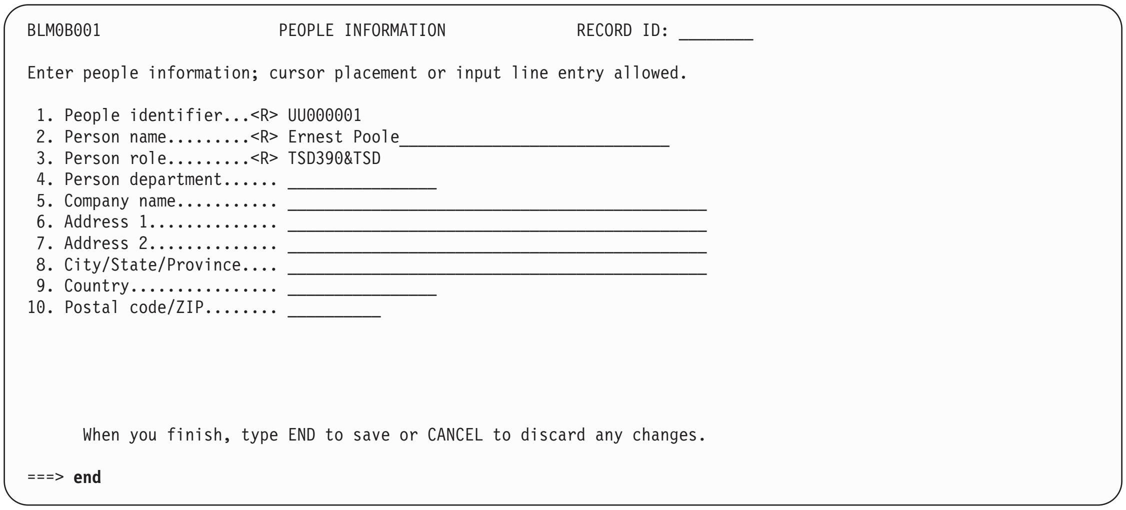
```
BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.



           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


  ===> 5
```

From panel BLG00000, type **6** and press **Enter** to select People Record.



```
     + BLG00000 -------------------- ENTRY --------------------- 1 OF 1-+
     |                                                                  |
     | USE....Identify the type of description (record) to be entered.  |
     |                                                                  |
     | 1.PROBLEM...........Enter data processing problem description.   |
     | 2.CHANGE............Enter change request for system/procedure.   |
     | 3.CONFIG............Enter description of system configuration,   |
     |                     financial data, or service organization.     |
     | 4.RULES.............Enter description of escalation rules.       |
     | 5.DATA MODEL........Enter description of a data model.           |
     | 6.PEOPLE............Enter description of a person.               |
     | 7.SOLUTION..........Enter solution record.                       |
     |                                                                  |
     +------------------------- SELECT ITEM ---------------------------+








===> 6
```

On panel BLM0B001, enter information about the person. The information that you provide in the **People identifier** field becomes the Record ID when the record is filed. In field 3, the **Person role**, you must identify the person as either **Customer**, **TSD390**, or **TSD390&TSD**. The designation of the person as **TSD390&TSD** or as **TSD390** is significant to the Tivoli Service Desk Bridge, described in *Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications*. When you have provided as much information as you like, type **end** and press **Enter** to save the record.



```
BLM0B001                PEOPLE INFORMATION             RECORD ID: ________

Enter people information; cursor placement or input line entry allowed.

 1. People identifier...<R> UU000001
 2. Person name.........<R> Ernest Poole____________________________
 3. Person role.........<R> TSD390&TSD
 4. Person department...... ________________
 5. Company name........... _____________________________________________
 6. Address 1.............. _____________________________________________
 7. Address 2.............. _____________________________________________
 8. City/State/Province.... _____________________________________________
 9. Country................ ________________
10. Postal code/ZIP........ __________





      When you finish, type END to save or CANCEL to discard any changes.

===> end
```

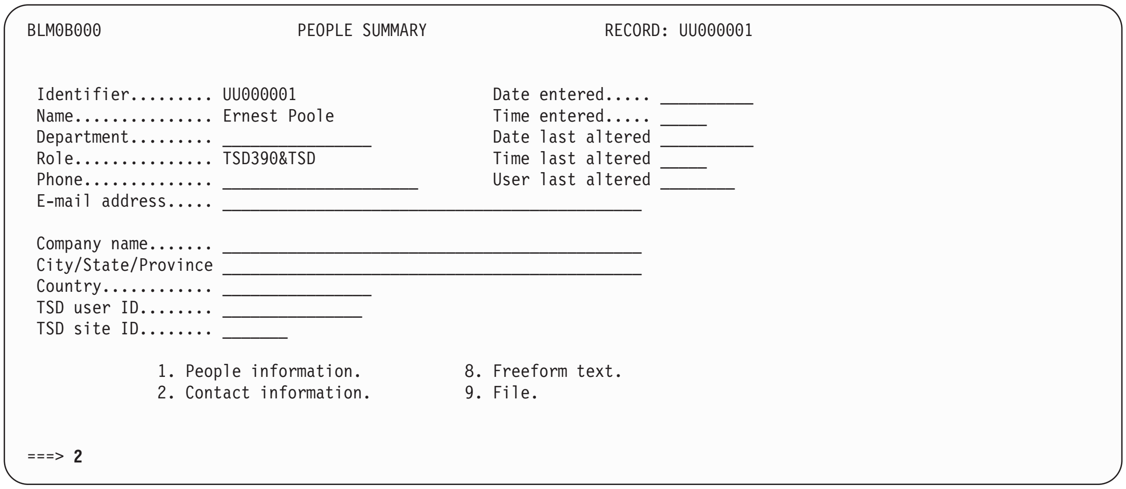
When you enter **end** on panel BLM0B001, panel BLM0B000 provides a summary of the data that you entered in the preceding step. You can select one of the options at the bottom of this panel to add additional information to the people record:

Type **1** and press **Enter** to go to panel BLM0B001 (see page 64) where you can add or modify data. Provide additional data, type **end**, and press **Enter** to return to BLM0B000.

Type **2** and press **Enter** to go to panel BLM0B002 (see page 66) where you can add contact information. Provide the contact information, type **end**, and press **Enter** to return to BLM0B000.

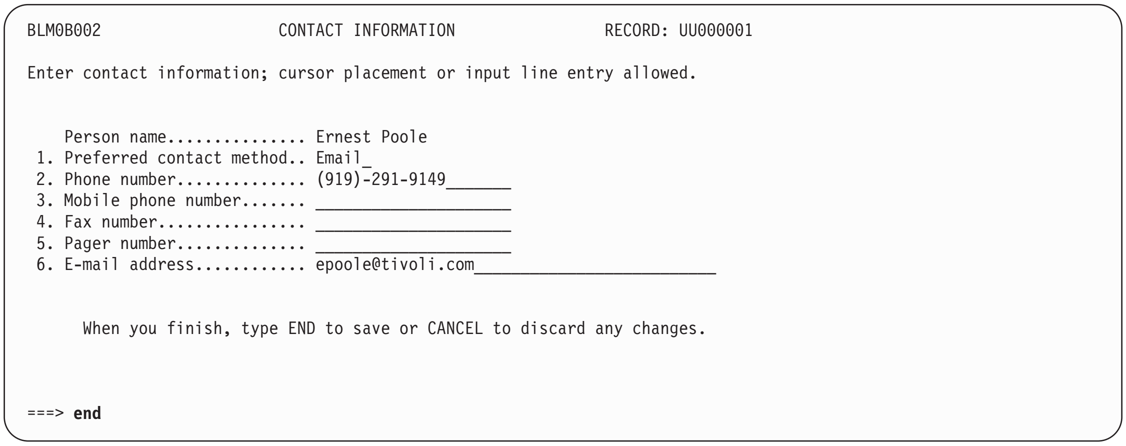
Type **8** and press **Enter**; if you are using the Tivoli Information Management for z/OS text editor (that is, you specified INFO in the Editor selection field in your Profile Summary), you will go to panel BLM1TDES (see page 66) where you can provide freeform information. Provide the freeform information on that panel, type **end**, and press **Enter** to return to BLM0B000.

Type **9** and press **Enter** to file the record.



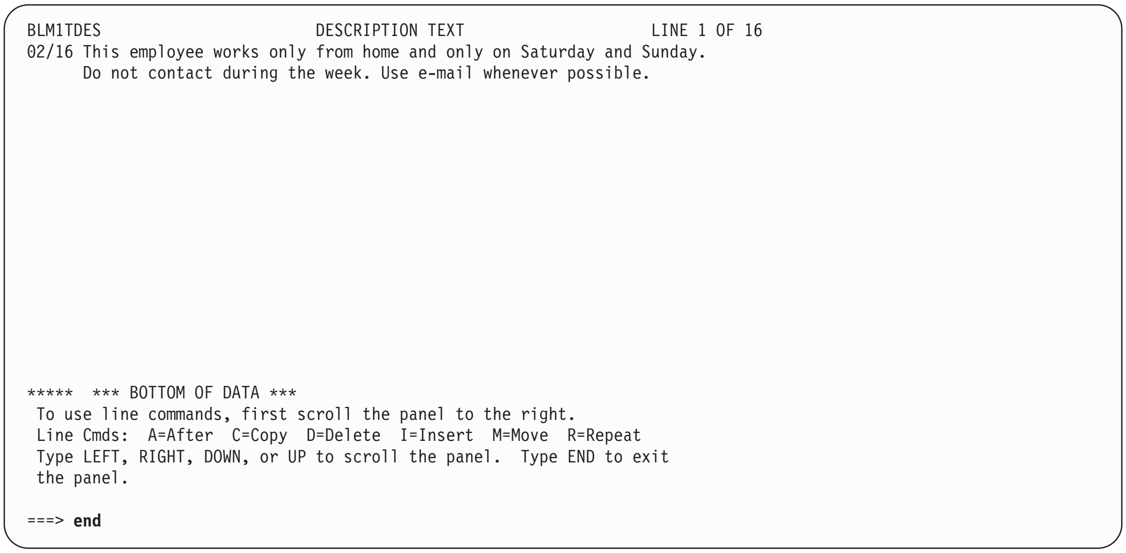
```
BLM0B000                    PEOPLE SUMMARY                   RECORD: UU000001


 Identifier......... UU000001                     Date entered..... __________
 Name............... Ernest Poole                 Time entered..... _____
 Department......... ________________             Date last altered __________
 Role............... TSD390&TSD                   Time last altered _____
 Phone.............. _____________________        User last altered ________
 E-mail address..... _____________________________________________

 Company name....... _____________________________________________
 City/State/Province _____________________________________________
 Country............ ________________
 TSD user ID........ _______________
 TSD site ID........ _______

             1. People information.           8. Freeform text.
             2. Contact information.          9. File.


===> 2
```

If you selected option **2** on panel BLM0B000, you can provide contact information on panel BLM0B002. Available choices for **Preferred contact method** are Phone, Email, Pager, Mobile, or FAX. After you enter the contact information, type **end** and press **Enter** to return to panel BLM0B000 (page 65).



```
BLM0B002                  CONTACT INFORMATION              RECORD: UU000001

Enter contact information; cursor placement or input line entry allowed.


    Person name............... Ernest Poole
 1. Preferred contact method.. Email_
 2. Phone number.............. (919)-291-9149______
 3. Mobile phone number....... ____________________
 4. Fax number................ ____________________
 5. Pager number.............. ____________________
 6. E-mail address............ epoole@tivoli.com__________________________


      When you finish, type END to save or CANCEL to discard any changes.



===> end
```

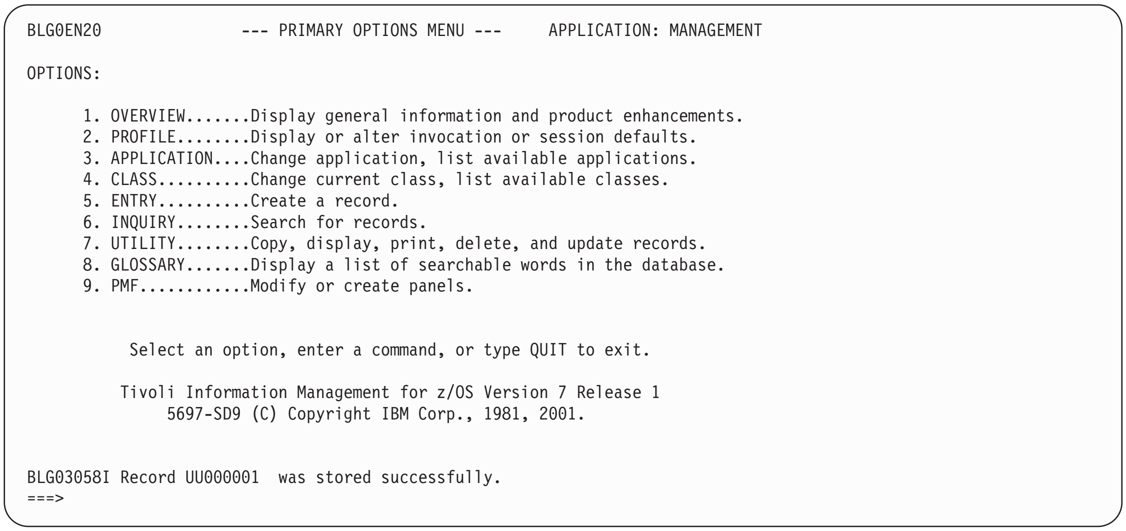
If you selected option **8** on panel BLM0B000, you can provide freeform text on panel BLM1TDES (assuming that you are using the Tivoli Information Management for z/OS text editor). After you enter the text, type **end** and press **Enter** to return to panel BLM0B000 (page 65).



```
BLM1TDES                      DESCRIPTION TEXT                   LINE 1 OF 16
02/16 This employee works only from home and only on Saturday and Sunday.
      Do not contact during the week. Use e-mail whenever possible.
















*****  *** BOTTOM OF DATA ***
 To use line commands, first scroll the panel to the right.
 Line Cmds:  A=After  C=Copy  D=Delete  I=Insert  M=Move  R=Repeat
 Type LEFT, RIGHT, DOWN, or UP to scroll the panel.  Type END to exit
 the panel.

===> end
```

If you selected option **9** on panel BLM0B000, you will receive message BLG03058I confirming that the record was successfully stored.

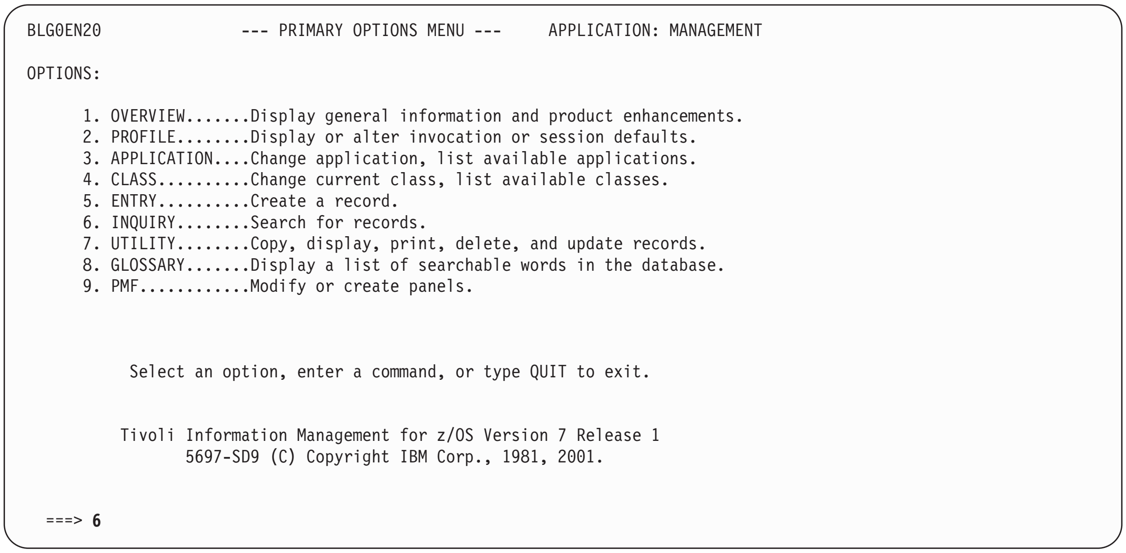
Note: If you replaced control panel BLM0B040 with panel BLM0B04Z (a process described in the *Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications*), when you file a people record with a role of **TSD390&TSD**, it is sent to the Tivoli Service Desk (TSD). Additional information can be found in the *Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications*.



```
BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.


           Select an option, enter a command, or type QUIT to exit.

          Tivoli Information Management for z/OS Version 7 Release 1
               5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


BLG03058I Record UU000001  was stored successfully.
===>
```

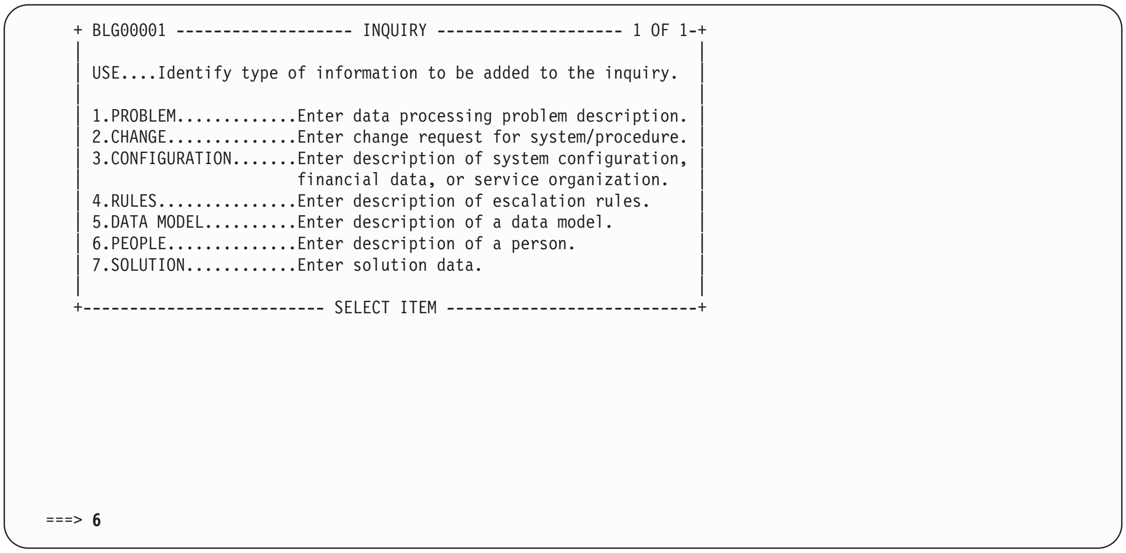
Searching for People Records

You can search for people records that have previously been entered. You will need to know some information about the people record in order to perform the search. From BLG0EN20, the Primary Options Menu, type **6** and press **Enter** to begin the search process.



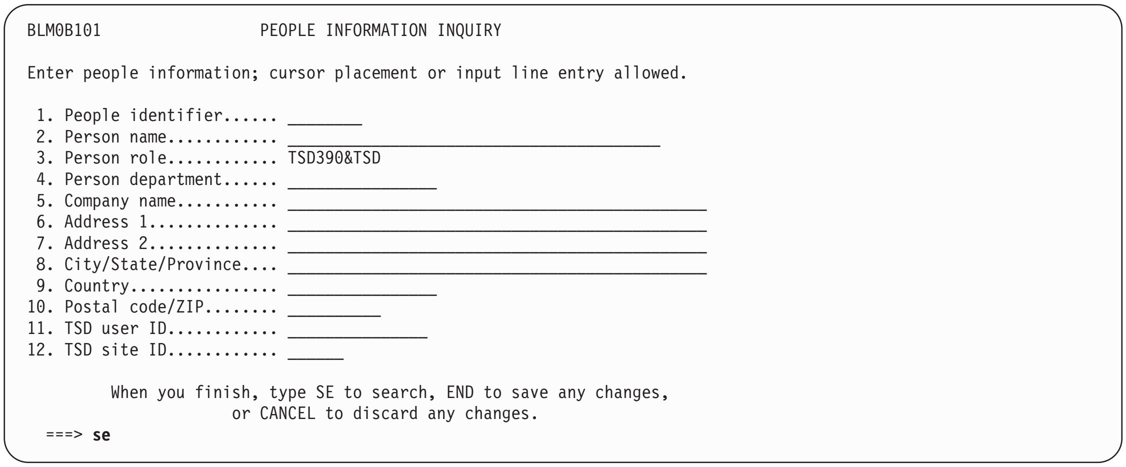
```
BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.



           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


  ===> 6
```

On panel BLG00001, type **6** and press **Enter** to restrict this search to people records.



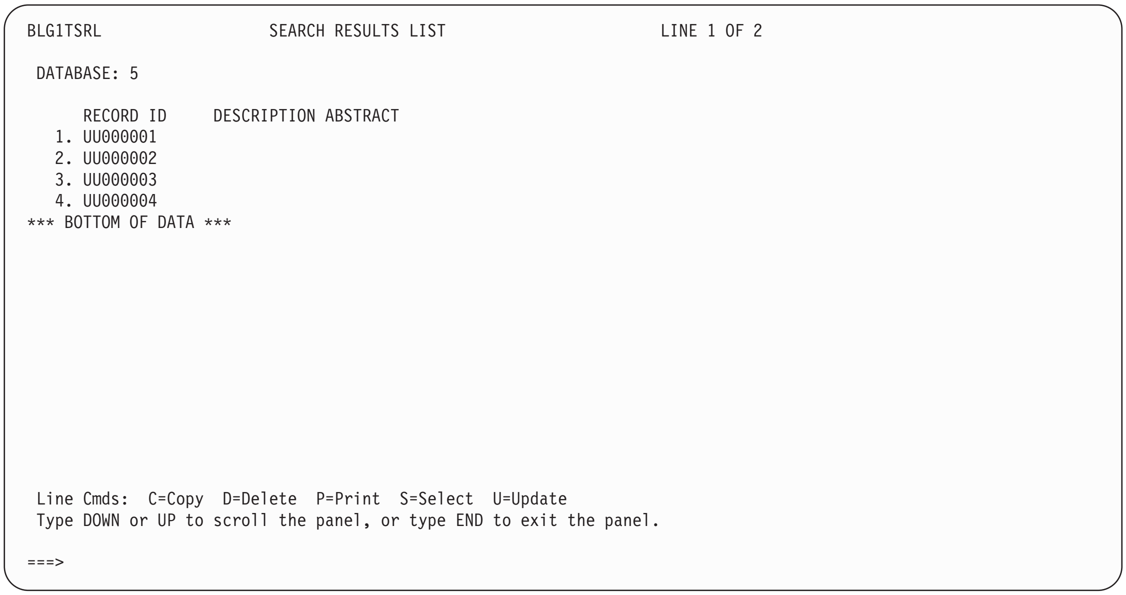
```
       + BLG00001 ------------------- INQUIRY -------------------- 1 OF 1-+
       |                                                                  |
       | USE....Identify type of information to be added to the inquiry.  |
       |                                                                  |
       | 1.PROBLEM............Enter data processing problem description.  |
       | 2.CHANGE.............Enter change request for system/procedure.  |
       | 3.CONFIGURATION......Enter description of system configuration,  |
       |                      financial data, or service organization.    |
       | 4.RULES..............Enter description of escalation rules.      |
       | 5.DATA MODEL.........Enter description of a data model.          |
       | 6.PEOPLE.............Enter description of a person.              |
       | 7.SOLUTION...........Enter solution data.                        |
       |                                                                  |
       +------------------------- SELECT ITEM --------------------------+







  ===> 6
```

On panel BLM0B101, enter some information to enable a search. Any data entered becomes the argument for the search. In this example, the Person role of **TSD390&TSD** is used as a search argument. After entering search data, type **se** on the command line and press **Enter** to begin the search. If you wish to save your changes, type **end** and press **Enter** to go to summary panel BLM0B100 (described in “Additional Search Capabilities” on page 70).



```
BLM0B101                 PEOPLE INFORMATION INQUIRY

Enter people information; cursor placement or input line entry allowed.

 1. People identifier...... ________
 2. Person name............ ________________________________________
 3. Person role............ TSD390&TSD
 4. Person department...... ________________
 5. Company name........... _____________________________________________
 6. Address 1.............. _____________________________________________
 7. Address 2.............. _____________________________________________
 8. City/State/Province.... _____________________________________________
 9. Country................ ________________
10. Postal code/ZIP........ __________
11. TSD user ID............ _______________
12. TSD site ID............ ______

         When you finish, type SE to search, END to save any changes,
                      or CANCEL to discard any changes.
  ===> se
```

Panel BLG1TSRL displays records that satisfy the search argument(s).



```
BLG1TSRL                  SEARCH RESULTS LIST                         LINE 1 OF 2

 DATABASE: 5

      RECORD ID     DESCRIPTION ABSTRACT
   1. UU000001
   2. UU000002
   3. UU000003
   4. UU000004
*** BOTTOM OF DATA ***









 Line Cmds:  C=Copy  D=Delete  P=Print  S=Select  U=Update
 Type DOWN or UP to scroll the panel, or type END to exit the panel.

===>
```

Additional Search Capabilities

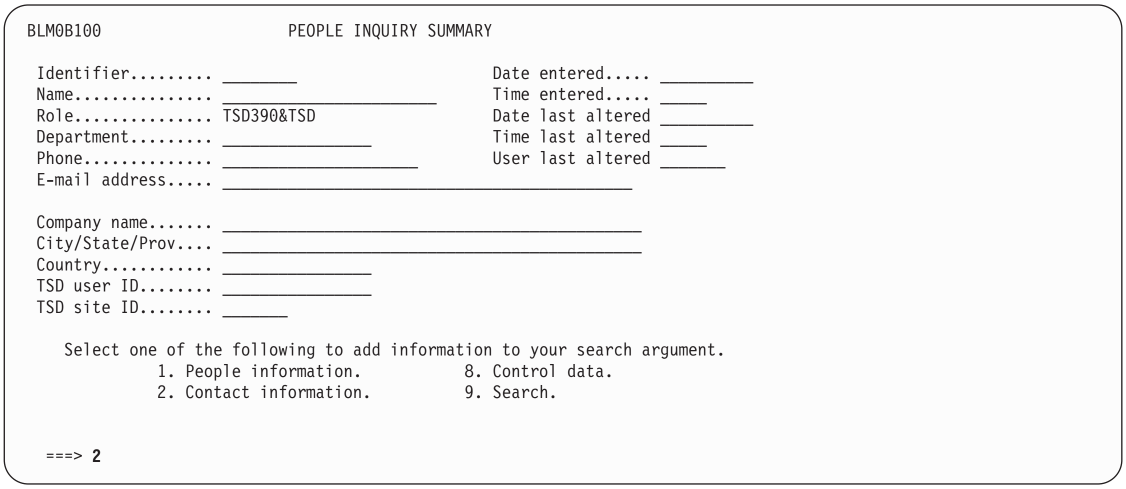
If you entered **end** from panel BLM0B101, the People Inquiry Summary panel, BLM0B100 is displayed. On this panel you can type one of the number choices at the bottom of the panel:

Type **1** and press **Enter** to return to panel BLM0B101 (seen earlier on page 69) where you can modify your search argument.

Type **2** and press **Enter** to go to panel BLM0B102 (see page 71) where you can add contact information.

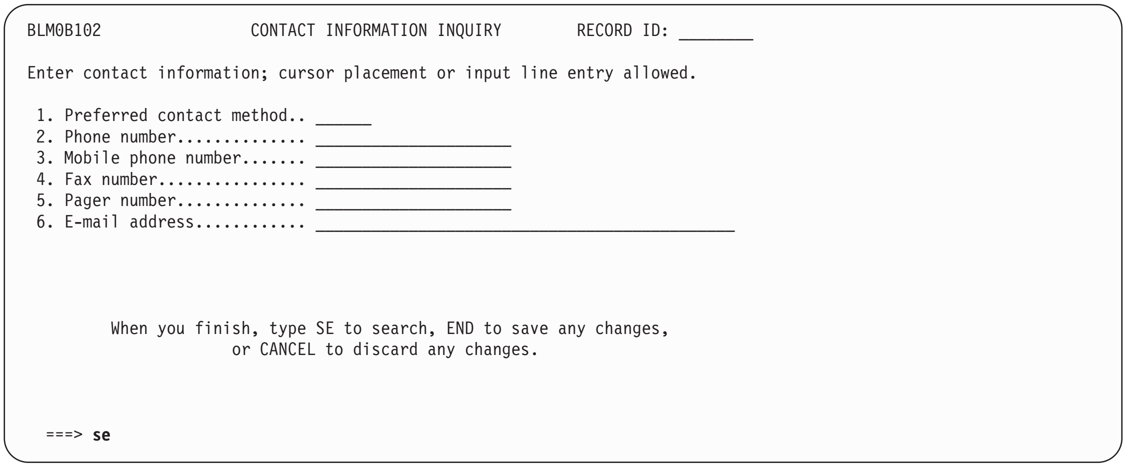
Type **8** and press **Enter** to go to panel BLM0B108 (see page 71) where you can add control information.

Type **9** and press **Enter** to do the search; the records satisfying your search arguments are displayed on the search results panel BLG1TSRL (see page 69).



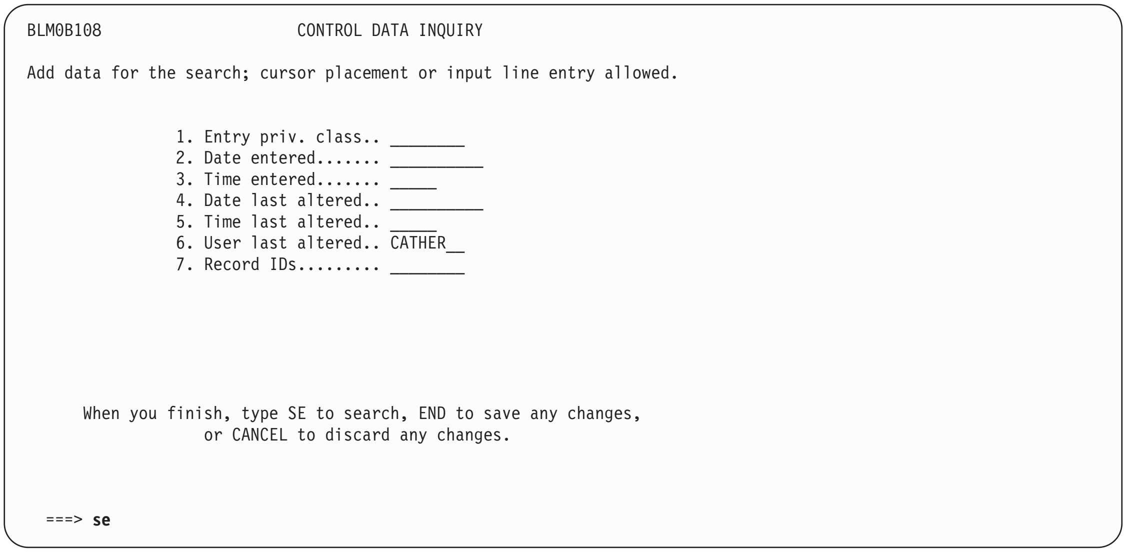
```
BLM0B100                    PEOPLE INQUIRY SUMMARY

 Identifier......... ________                    Date entered..... __________
 Name............... _______________________     Time entered..... _____
 Role............... TSD390&TSD                  Date last altered __________
 Department......... ________________            Time last altered _____
 Phone.............. _____________________       User last altered _______
 E-mail address..... ____________________________________________

 Company name....... _____________________________________________
 City/State/Prov.... _____________________________________________
 Country............ ________________
 TSD user ID........ ________________
 TSD site ID........ _______

    Select one of the following to add information to your search argument.
              1. People information.          8. Control data.
              2. Contact information.         9. Search.


  ===> 2
```

If you entered option **2** on panel BLM0B100, panel BLM0B102 provides the opportunity to add “contact” information to your search argument. Information entered here constitutes additional search parameters; type **se** and press **Enter** to begin the search with the information entered on this panel.



```
BLM0B102               CONTACT INFORMATION INQUIRY        RECORD ID: ________

Enter contact information; cursor placement or input line entry allowed.

 1. Preferred contact method.. ______
 2. Phone number............. _____________________
 3. Mobile phone number....... _____________________
 4. Fax number............... _____________________
 5. Pager number............. _____________________
 6. E-mail address........... _____________________________________________




         When you finish, type SE to search, END to save any changes,
                      or CANCEL to discard any changes.



  ===> se
```

If you entered option **8** on panel BLM0B100, panel BLM0B108 allows you to add control information. Information entered here constitutes additional search parameters; type **se** and press **Enter** to begin the search with the information entered on this panel.



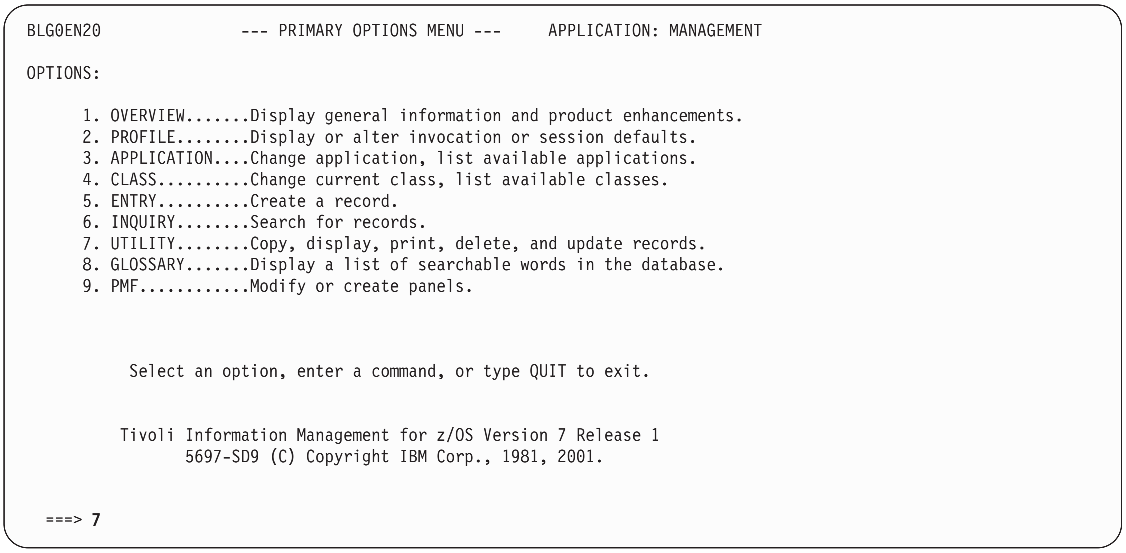
```
BLM0B108                    CONTROL DATA INQUIRY

Add data for the search; cursor placement or input line entry allowed.


               1. Entry priv. class.. ________
               2. Date entered....... __________
               3. Time entered....... _____
               4. Date last altered.. __________
               5. Time last altered.. _____
               6. User last altered.. CATHER__
               7. Record IDs......... ________






      When you finish, type SE to search, END to save any changes,
                   or CANCEL to discard any changes.



  ===> se
```

Deleting People Records

You can delete people records that have previously been entered. In order to do this, you must know the Person role that was assigned to the person. If the assigned Person role is **Customer** or **TSD390**, you can delete these either by using the Delete command (for example, **DEL R UU000001**) or by using the following panel process. However, if the assigned Person role is **TSD390&TSD**, you must update the record and change the Person role to either **Customer** or **TSD390** before you can delete the record. A process for doing this is described in "Deleting Records Having a 'Person Role' of TSD390&TSD" on page 75.

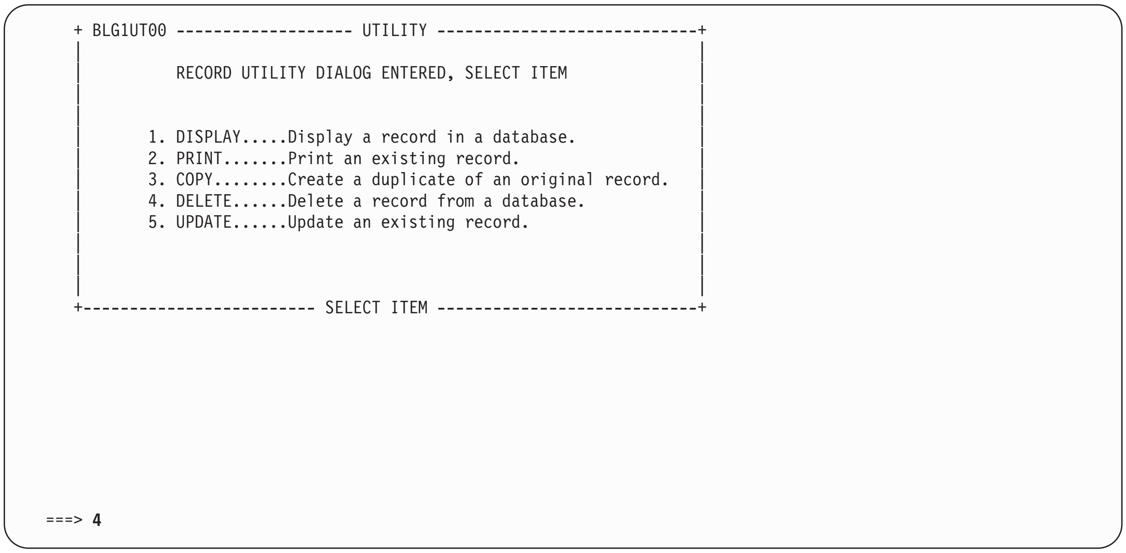
Note: A TSD390 user may also see people records having a Person role of **TSDUSER** or **TSDGROUP**. Records containing either of these Person roles are owned by TSD, and can be viewed, but not deleted or otherwise modified, by a TSD390 user.

To begin the process of deleting records, by typing **7** and pressing **Enter** on the Primary Options Menu.



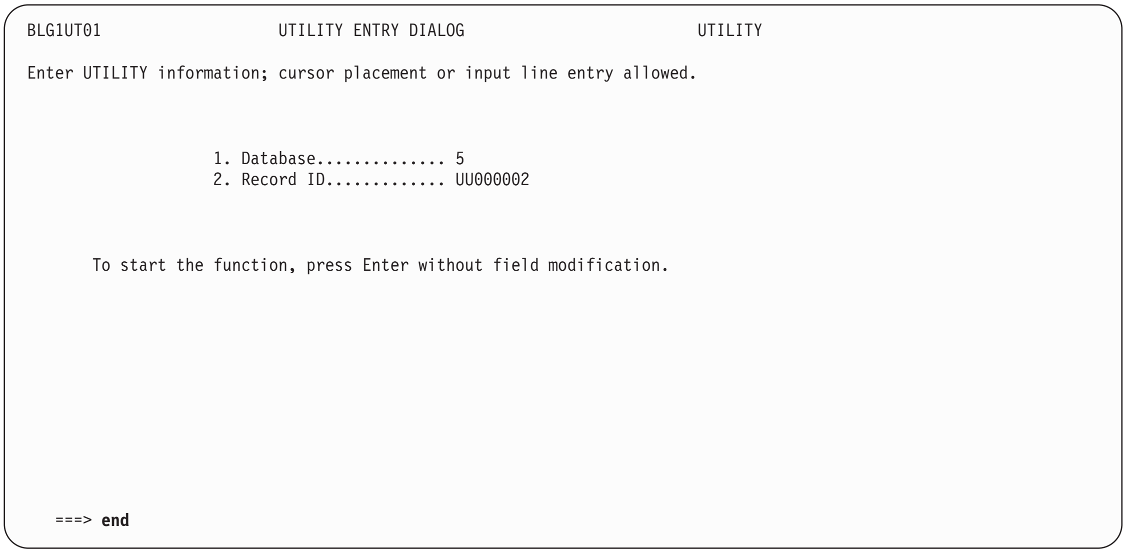
```
BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.



           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


  ===> 7
```

On panel BLG1UT00, type **4** and press **Enter** to go to panel BLG1TU01.



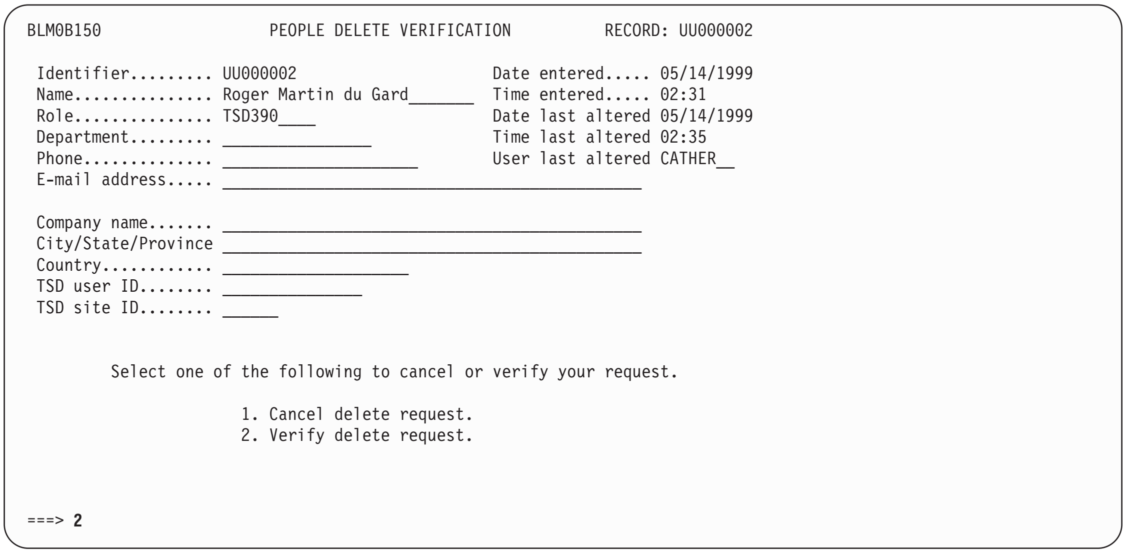
```
+ BLG1UT00 ------------------- UTILITY ----------------------------+
|                                                                  |
|          RECORD UTILITY DIALOG ENTERED, SELECT ITEM              |
|                                                                  |
|                                                                  |
|       1. DISPLAY.....Display a record in a database.             |
|       2. PRINT.......Print an existing record.                   |
|       3. COPY........Create a duplicate of an original record.   |
|       4. DELETE......Delete a record from a database.            |
|       5. UPDATE......Update an existing record.                  |
|                                                                  |
|                                                                  |
|                                                                  |
+------------------------- SELECT ITEM ----------------------------+




===> 4
```

On panel BLG1UT01, type the number of the record that you want to delete; then type **end** on the command line and press **Enter**.



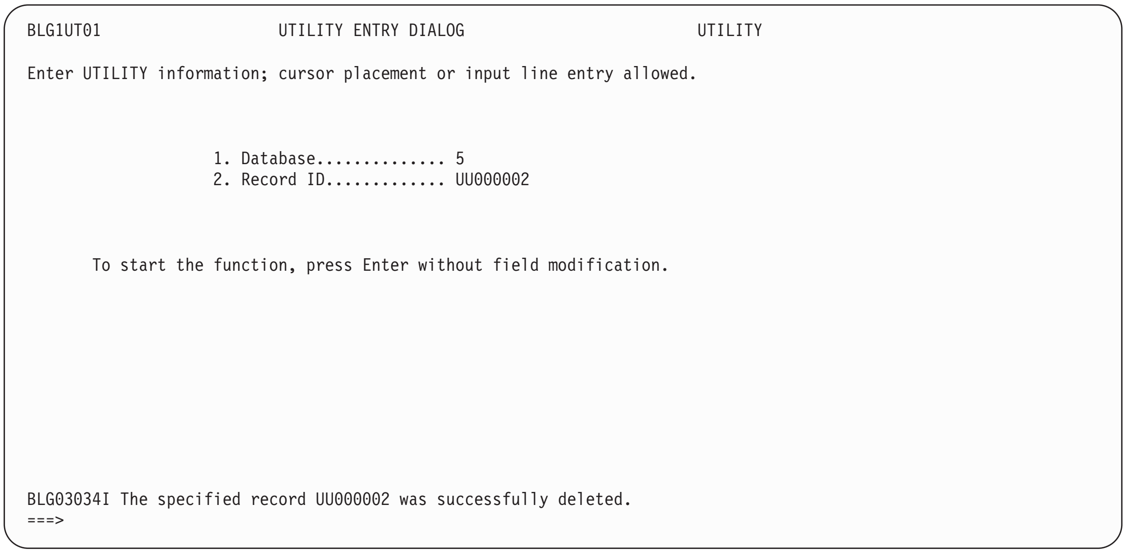
```
BLG1UT01                  UTILITY ENTRY DIALOG                         UTILITY

Enter UTILITY information; cursor placement or input line entry allowed.



                    1. Database............. 5
                    2. Record ID............ UU000002



       To start the function, press Enter without field modification.




   ===> end
```

On the verification panel, BLM0B150, type **2** and press **Enter** to confirm that you want to delete the record.



```
BLM0B150                  PEOPLE DELETE VERIFICATION          RECORD: UU000002

 Identifier......... UU000002                     Date entered..... 05/14/1999
 Name............... Roger Martin du Gard_______  Time entered..... 02:31
 Role............... TSD390____                   Date last altered 05/14/1999
 Department......... ________________             Time last altered 02:35
 Phone.............. _____________________        User last altered CATHER__
 E-mail address..... _____________________________________________

 Company name....... _____________________________________________
 City/State/Province _____________________________________________
 Country............ ____________________
 TSD user ID........ _______________
 TSD site ID........ ______


         Select one of the following to cancel or verify your request.

                       1. Cancel delete request.
                       2. Verify delete request.



===> 2
```

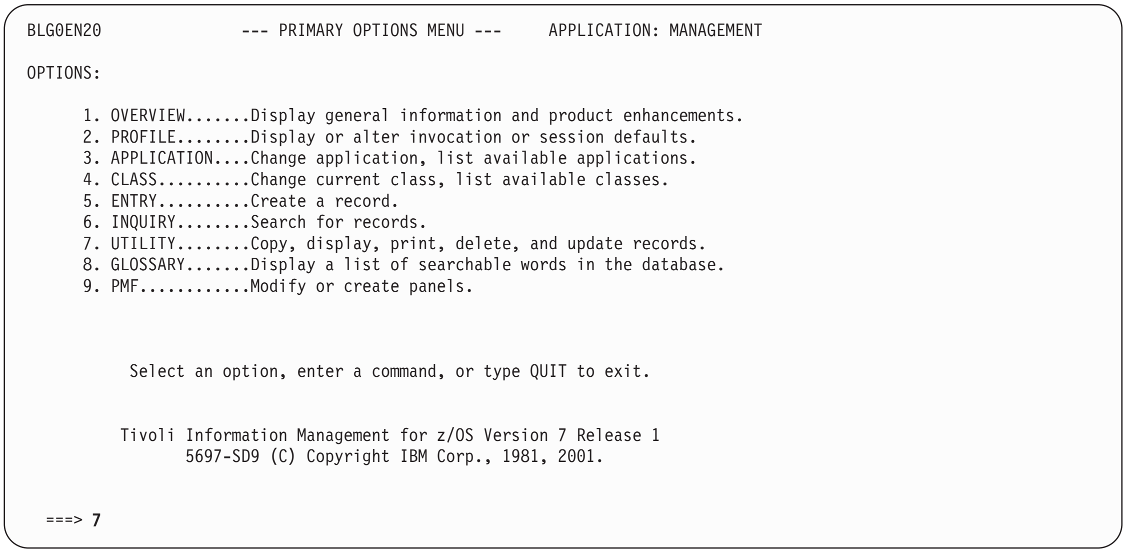
If the people record contains a Person role of **Customer** or **TSD390**, the record will be deleted.



```
BLG1UT01                   UTILITY ENTRY DIALOG                         UTILITY

Enter UTILITY information; cursor placement or input line entry allowed.



                    1. Database............. 5
                    2. Record ID............ UU000002



       To start the function, press Enter without field modification.









BLG03034I The specified record UU000002 was successfully deleted.
===>
```

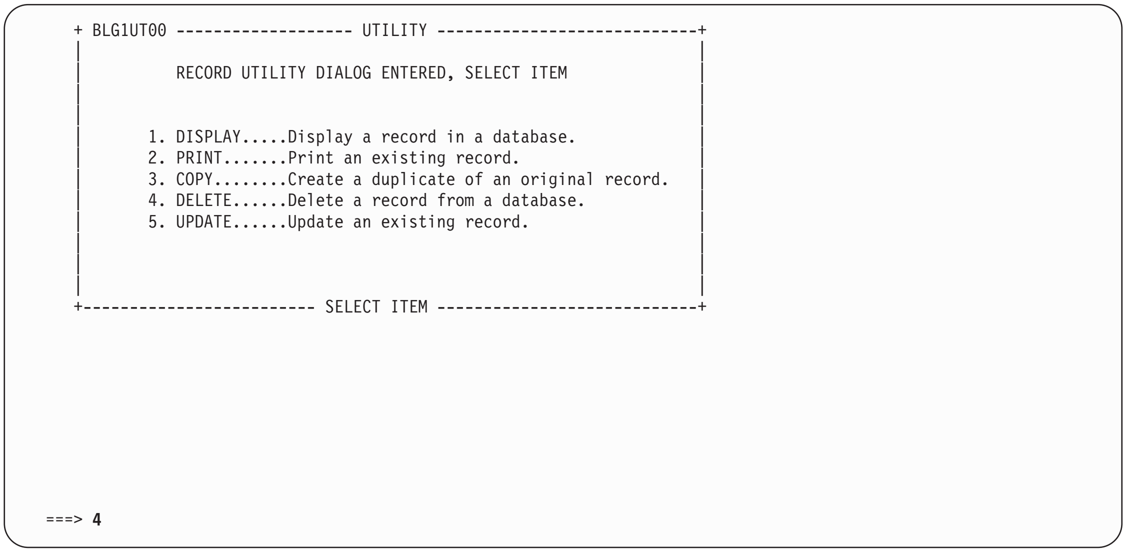
Deleting Records Having a ‘Person Role’ of TSD390&TSD

Records that have a Person role of **TSD390&TSD** cannot be deleted unless the Person role is changed to **Customer** or **TSD390**. The reasons for this have to do with the notification process in TSD, described in *Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications*. In the following example, assume that record UU000009 has been created with a Person role of **TSD390&TSD**. On the Primary Options Menu, type **7** and press **Enter**r.



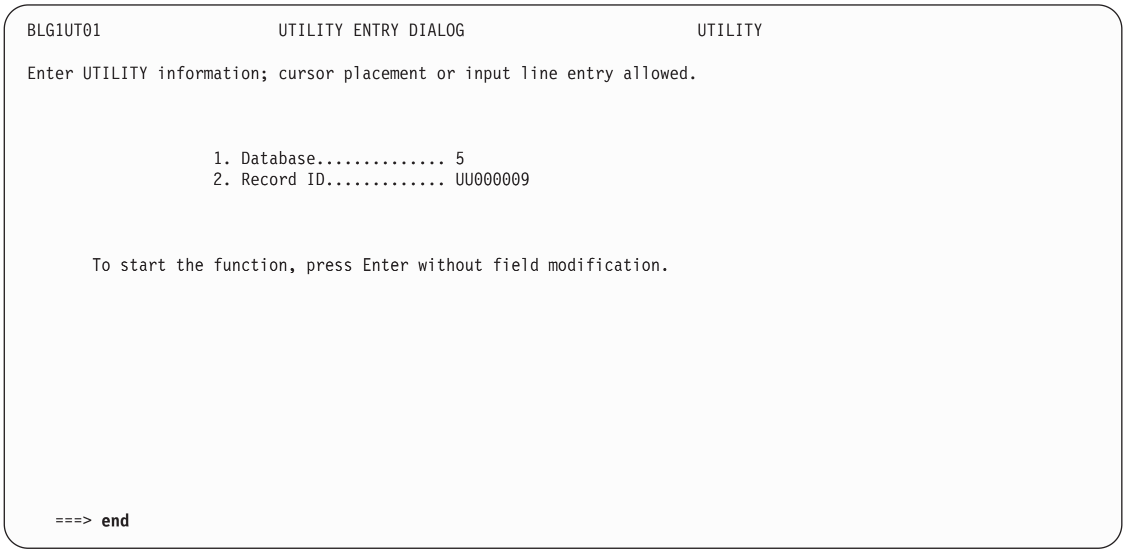
```
BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.



           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


  ===> 7
```

On panel BLG1UT00, type **4** and press **Enter** to go to panel BLG1TU01.



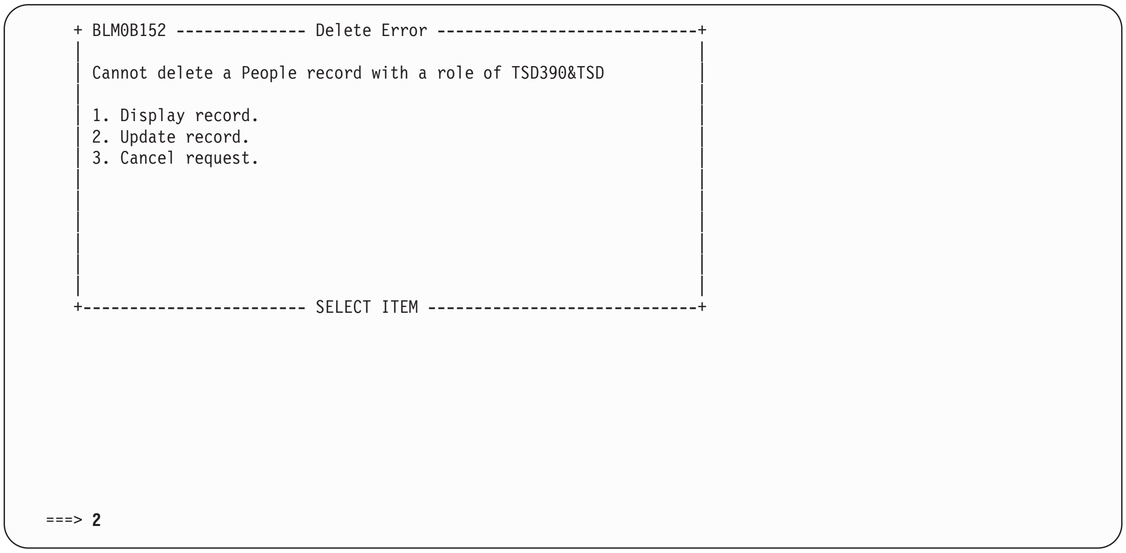
```
       + BLG1UT00 ------------------- UTILITY ----------------------------+
       |                                                                  |
       |          RECORD UTILITY DIALOG ENTERED, SELECT ITEM              |
       |                                                                  |
       |                                                                  |
       |       1. DISPLAY.....Display a record in a database.             |
       |       2. PRINT.......Print an existing record.                   |
       |       3. COPY........Create a duplicate of an original record.   |
       |       4. DELETE......Delete a record from a database.            |
       |       5. UPDATE......Update an existing record.                  |
       |                                                                  |
       |                                                                  |
       |                                                                  |
       +------------------------ SELECT ITEM -----------------------------+







  ===> 4
```

On panel BLG1UT01, type the number of the record that you want to delete; then type **end** on the command line and press **Enter**.



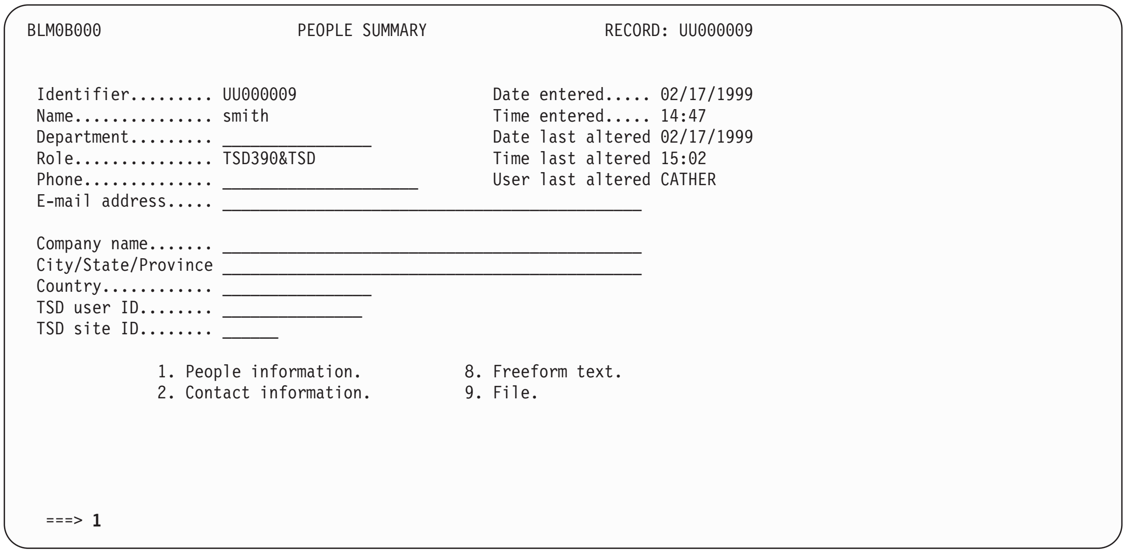
```
BLG1UT01                   UTILITY ENTRY DIALOG                         UTILITY

Enter UTILITY information; cursor placement or input line entry allowed.



                    1. Database............. 5
                    2. Record ID............ UU000009



       To start the function, press Enter without field modification.











   ===> end
```

If this people record contains a Person role of **TSD390&TSD**, error panel BLM0B152 explains that you cannot delete such a record. In order to delete this record, you must first change the Person role from **TSD390&TSD** to a different Person role so that you can delete the record; in this example, the record will be updated with a Person role of **TSD390**. On panel BLM0B2, enter **2** and press **Enter**.



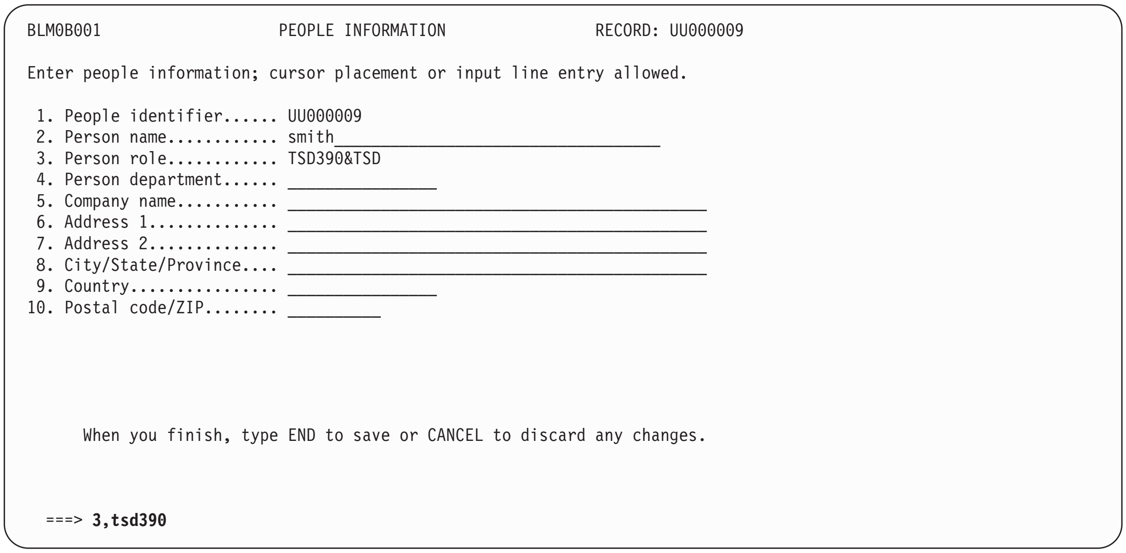
```
       + BLM0B152 -------------- Delete Error ----------------------------+
       |                                                                  |
       | Cannot delete a People record with a role of TSD390&TSD          |
       |                                                                  |
       | 1. Display record.                                               |
       | 2. Update record.                                                |
       | 3. Cancel request.                                               |
       |                                                                  |
       |                                                                  |
       |                                                                  |
       |                                                                  |
       |                                                                  |
       |                                                                  |
       +------------------------ SELECT ITEM -----------------------------+









  ===> 2
```

Notice that the record displayed, UU000009, contains a Person role of **TSD390&TSD**. Type **1** and press **Enter** to select People information, which contains the Person role.



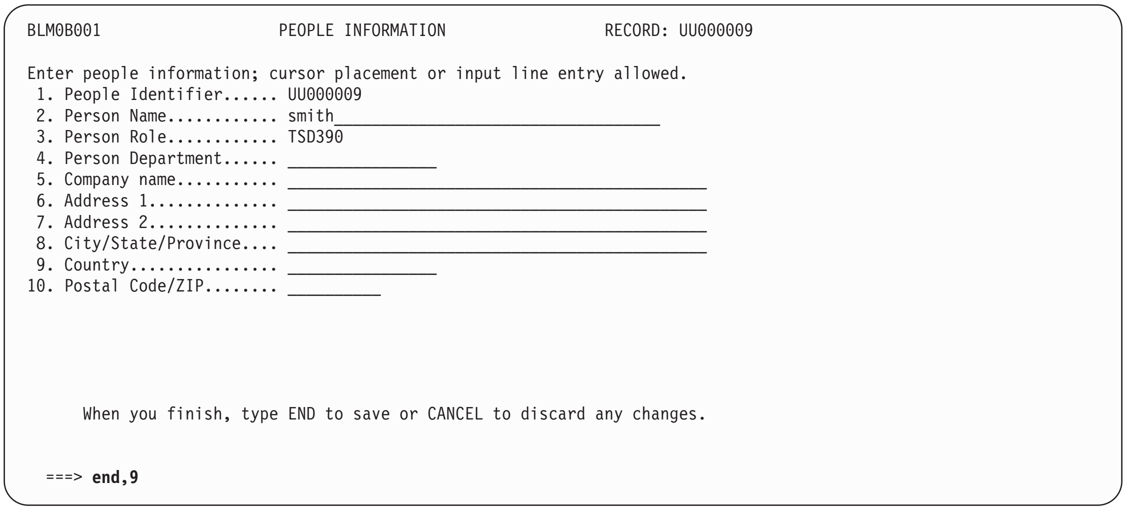
```
BLM0B000                    PEOPLE SUMMARY                  RECORD: UU000009


 Identifier......... UU000009                    Date entered..... 02/17/1999
 Name............... smith                       Time entered..... 14:47
 Department......... ________________            Date last altered 02/17/1999
 Role............... TSD390&TSD                  Time last altered 15:02
 Phone.............. _____________________       User last altered CATHER
 E-mail address..... _____________________________________________

 Company name....... _____________________________________________
 City/State/Province _____________________________________________
 Country............ ________________
 TSD user ID........ _______________
 TSD site ID........ ______

              1. People information.           8. Freeform text.
              2. Contact information.           9. File.





  ===> 1
```

Type **3,TSD390** and press **Enter** to modify the Person role.



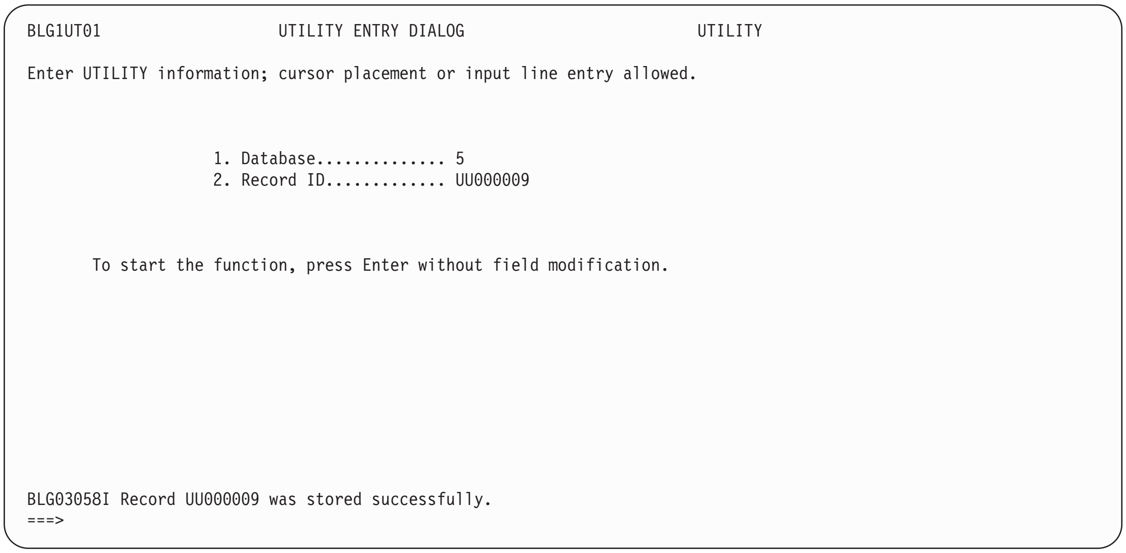
```
BLM0B001                  PEOPLE INFORMATION                RECORD: UU000009

Enter people information; cursor placement or input line entry allowed.

 1. People identifier...... UU000009
 2. Person name............ smith___________________________________
 3. Person role............ TSD390&TSD
 4. Person department...... ________________
 5. Company name........... _____________________________________________
 6. Address 1.............. _____________________________________________
 7. Address 2.............. _____________________________________________
 8. City/State/Province.... _____________________________________________
 9. Country................ ________________
10. Postal code/ZIP........ __________





      When you finish, type END to save or CANCEL to discard any changes.



  ===> 3,tsd390
```

The redisplayed record now shows a Person role of **TSD390**. Type **end,9** and press **Enter** to file the record.



```
BLM0B001                  PEOPLE INFORMATION               RECORD: UU000009

Enter people information; cursor placement or input line entry allowed.
 1. People Identifier...... UU000009
 2. Person Name............ smith___________________________________
 3. Person Role............ TSD390
 4. Person Department...... ________________
 5. Company name........... _____________________________________________
 6. Address 1.............. _____________________________________________
 7. Address 2.............. _____________________________________________
 8. City/State/Province.... _____________________________________________
 9. Country................ ________________
10. Postal Code/ZIP........ __________





      When you finish, type END to save or CANCEL to discard any changes.


  ===> end,9
```

A message confirms that the record (now with a Role of **TSD390**) has been stored successfully. This record can be deleted using the process described in “Deleting People Records” on page 72 once the notification data has been processed by TSD.



```
BLG1UT01                  UTILITY ENTRY DIALOG                       UTILITY

Enter UTILITY information; cursor placement or input line entry allowed.



                    1. Database............. 5
                    2. Record ID............ UU000009



       To start the function, press Enter without field modification.









BLG03058I Record UU000009 was stored successfully.
===>
```

4

Customizing the Enhanced Panel Style

This information is provided for the program administrator who is responsible for customizing the enhanced panel style for Tivoli Information Management for z/OS. To do the tasks explained in this chapter, you must be familiar with modifying ISPF panels and tables. For more information, refer to the following publications for the prerequisite version of ISPF:

- *ISPF User's Guide*
- *ISPF Dialog Devloper's Guide and Reference*
- *ISPF Edit and Edit Macros*
- *ISPF Dialog Tag Language and Reference*
- *ISPF Planning and Customizing*

Refer to the *ISPF Dialog Developer's Guide and Reference* for information about enabling applications for ISPF graphical user interface (GUI) mode.

Tivoli Information Management for z/OS Panel Styles

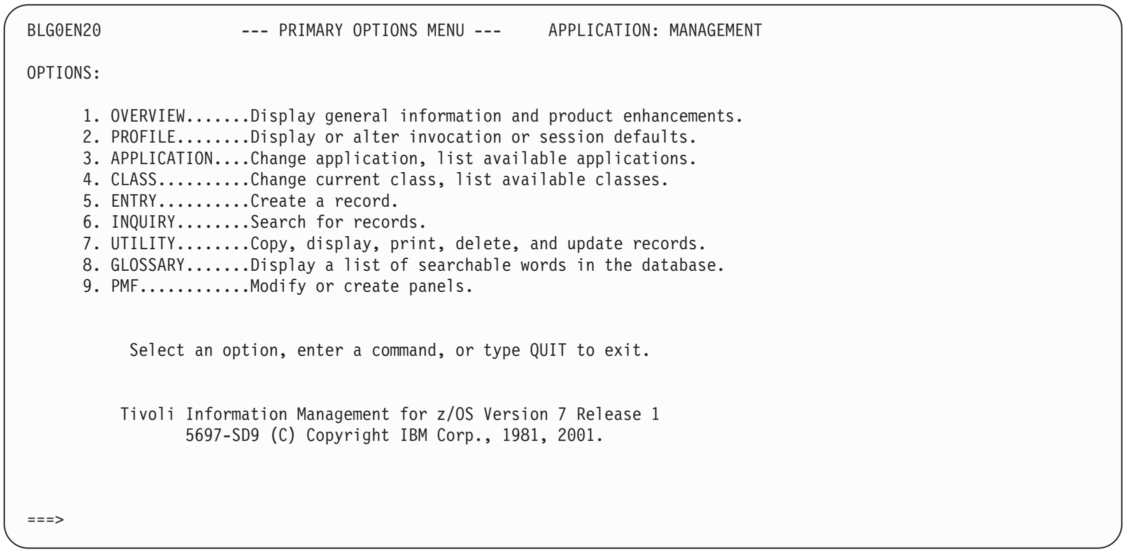
Tivoli Information Management for z/OS provides the following panel styles:

- The *standard panel style*, which provides a command line. See “Standard Panel Style” on page 80 for an illustration.
- The *enhanced panel style*, which provides an action bar, pull-down menus, and can display context-specific function keys, accelerator keys, and separator bars. A set of enhanced panel style windows is provided in the sample data set. See “Enhanced Panel Style” on page 80 for illustrations. Refer to the *Tivoli Information Management for z/OS User's Guide* for information on using the enhanced panel style.

Tivoli Information Management for z/OS provides the WINDOW command to enable you to change the panel style you are using. If you use the enhanced panel style, you can choose between automatic window selection and manual window selection. Refer to the *Tivoli Information Management for z/OS User's Guide* for command syntax and details.

Standard Panel Style

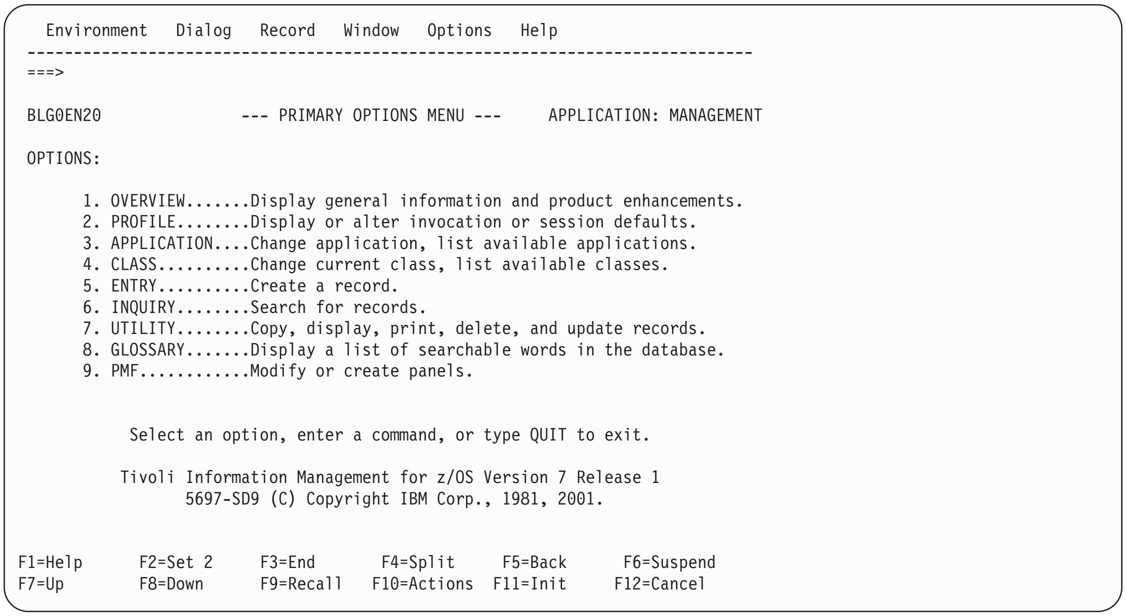
This is the Primary Options Menu displayed using the standard panel style window, BLGISPFD.



```
BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.



===>
```

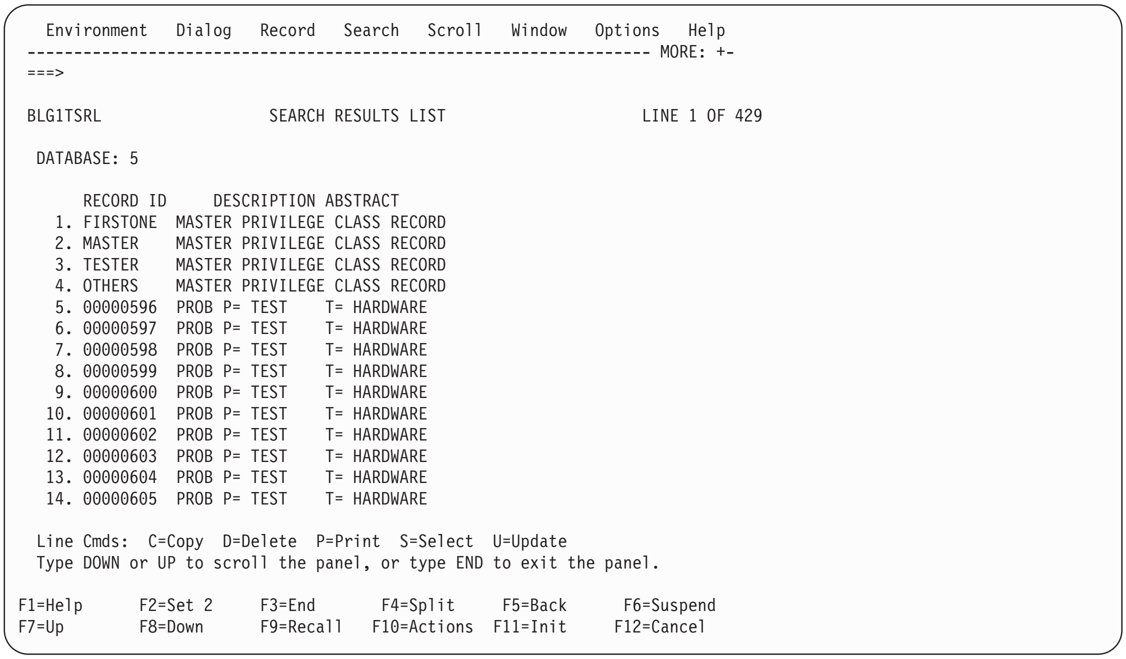
Enhanced Panel Style

This is the Primary Options Menu displayed using the enhanced panel style window BLGISPFE, the primary window.



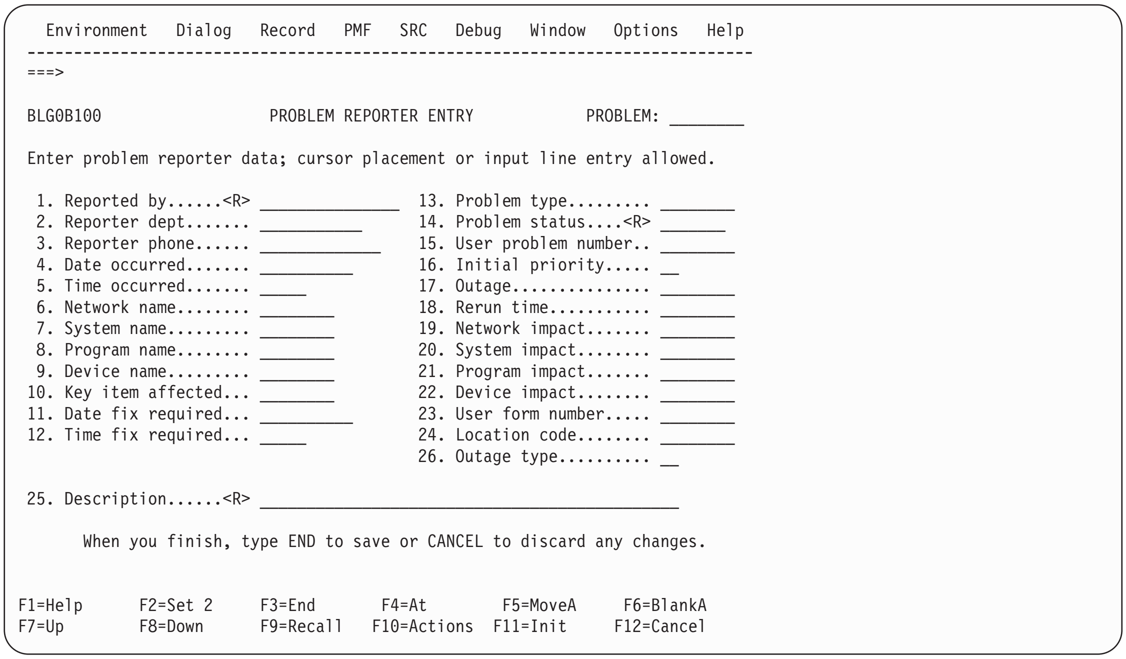
```
  Environment   Dialog   Record   Window   Options   Help
------------------------------------------------------------------------------
===>

BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.


           Select an option, enter a command, or type QUIT to exit.

          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


F1=Help      F2=Set 2     F3=End       F4=Split     F5=Back      F6=Suspend
F7=Up        F8=Down      F9=Recall   F10=Actions  F11=Init     F12=Cancel
```

This is a search results list displayed using the enhanced panel style window BLGISPFI, the inquiry window.



```
   Environment   Dialog   Record   Search   Scroll   Window   Options   Help
 ------------------------------------------------------------------- MORE: +-
 ===>

 BLG1TSRL                  SEARCH RESULTS LIST                     LINE 1 OF 429

  DATABASE: 5

       RECORD ID     DESCRIPTION ABSTRACT
    1. FIRSTONE  MASTER PRIVILEGE CLASS RECORD
    2. MASTER    MASTER PRIVILEGE CLASS RECORD
    3. TESTER    MASTER PRIVILEGE CLASS RECORD
    4. OTHERS    MASTER PRIVILEGE CLASS RECORD
    5. 00000596  PROB P= TEST    T= HARDWARE
    6. 00000597  PROB P= TEST    T= HARDWARE
    7. 00000598  PROB P= TEST    T= HARDWARE
    8. 00000599  PROB P= TEST    T= HARDWARE
    9. 00000600  PROB P= TEST    T= HARDWARE
   10. 00000601  PROB P= TEST    T= HARDWARE
   11. 00000602  PROB P= TEST    T= HARDWARE
   12. 00000603  PROB P= TEST    T= HARDWARE
   13. 00000604  PROB P= TEST    T= HARDWARE
   14. 00000605  PROB P= TEST    T= HARDWARE

  Line Cmds:  C=Copy  D=Delete  P=Print  S=Select  U=Update
  Type DOWN or UP to scroll the panel, or type END to exit the panel.

F1=Help      F2=Set 2     F3=End       F4=Split     F5=Back      F6=Suspend
F7=Up        F8=Down      F9=Recall   F10=Actions  F11=Init     F12=Cancel
```

This is an example of panel externals displayed when updating a panel in PMF. This display is using the enhanced panel style window BLGISPFA, the administration window.



```
   Environment   Dialog   Record   PMF   SRC   Debug   Window   Options   Help
 ------------------------------------------------------------------------------
 ===>

 BLG0B100                  PROBLEM REPORTER ENTRY            PROBLEM: ________

 Enter problem reporter data; cursor placement or input line entry allowed.

  1. Reported by......<R> _______________  13. Problem type......... ________
  2. Reporter dept....... ___________      14. Problem status....<R> _______
  3. Reporter phone...... _____________    15. User problem number.. ________
  4. Date occurred....... __________       16. Initial priority..... __
  5. Time occurred....... _____            17. Outage............... ________
  6. Network name........ ________         18. Rerun time........... ________
  7. System name......... ________         19. Network impact....... ________
  8. Program name........ ________         20. System impact........ ________
  9. Device name......... ________         21. Program impact....... ________
 10. Key item affected... ________         22. Device impact........ ________
 11. Date fix required... __________       23. User form number..... ________
 12. Time fix required... _____            24. Location code........ ________
                                           26. Outage type.......... __

 25. Description......<R> _____________________________________________

       When you finish, type END to save or CANCEL to discard any changes.


F1=Help      F2=Set 2     F3=End       F4=At        F5=MoveA     F6=BlankA
F7=Up        F8=Down      F9=Recall   F10=Actions  F11=Init     F12=Cancel
```

As a program administrator, you may want to modify one or more of the enhanced panel style windows to meet the requirements of your users. For example, you may want to:

- Enable the windows in the enhanced panel style set for ISPF GUI mode
- Change one of the windows in the enhanced panel style set
- Create additional windows
- Change the function key definitions

If your organization has installed the enhanced panel style and some of your users want to use the standard panel style, you can accommodate that group of users. See “Changing the Default ISPF Panel” on page 83 for details.

Understanding the ISPF Panels

From a program administrator’s point of view, Tivoli Information Management for z/OS windows are actually ISPF panels. The ISPF panels that you use must reside in a data set that is in your ISPPLIB concatenation. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for installation instructions. The following ISPF panels are shipped in the SBLMSAMP data set:

BLGISPFD
Contains the standard panel style

BLGISPFE
Contains the primary window for the enhanced panel style

BLGISPFI
Contains the inquiry window for the enhanced panel style

BLGISPFA
Contains the administration window for the enhanced panel style

You must also have ISPF panel BLGISPFM in a data set that is in your ISPPLIB concatenation. This panel contains the default panel style for your installation. This style was chosen at the time Tivoli Information Management for z/OS was installed. It is a copy of either BLGISPFD or BLGISPFE.

ISPF Tables

Because Tivoli Information Management for z/OS commands and ISPF commands can have the same name, Tivoli Information Management for z/OS provides the following ISPF tables:

BLG0CMDS
Contains a mapping of alias names for Tivoli Information Management for z/OS commands to the proper Tivoli Information Management for z/OS syntax.

BLG0KEYS
Contains the keylists that map the alias names to function keys.

If you use the enhanced panel style, the data sets containing these tables must be in your ISPTLIB concatenation. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for installation instructions.

Variables for Automating Window Selection

The ISPF panels that display the enhanced panel style windows use the following variables to determine which of the enhanced windows to display:

BLG#PTYP
Set to the type of the Tivoli Information Management for z/OS panel. (Refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide* for descriptions and illustrations of Tivoli Information Management for z/OS panel types.) The following values are used:

pmf Specifies a PMF panel type

select Specifies a selection panel type

assisted
Specifies an assisted-entry panel type

data Specifies a data-entry panel type

table Specifies a table panel type

other Specifies that the panel type does not match any of the other values. The panel could be a control panel or a terminal simulator panel.

If the value of BLG#PTYP is **pmf**, the administration window (BLGISPFA) is displayed. If the value is **table**, the values of the next two variables determine whether to display the inquiry window or the primary window.

BLG#PNAM

Set to the panel name. This variable is used to determine whether to display the primary window or the inquiry window when the value of BLG#PTYP is **table**. The inquiry window is displayed if the value of BLG#PNAM is one of the following:

- The same as the value of BLG#TNAM
- BLG1TSRL
- BLG1TMSQ
- BLG1TSQL

If the value is anything else, the primary window is displayed.

BLG#TNAM

Set to the name of the Tivoli Information Management for z/OS panel that is used to display a search results list. The following sources are used, in the order listed, to set this variable:

1. The last TABLE command issued during this session
2. The profile setting for the **Default panel** field in the Search options part of the Session Control panel

The WINDOW command can be used to override automatic window selection. Refer to the *Tivoli Information Management for z/OS User’s Guide* for details.

Changing the Default ISPF Panel

The default ISPF panel is the one that is used for the first panel after initializing Tivoli Information Management for z/OS.

To change the default ISPF panel for a set of users, copy the desired ISPF panel into BLGISPFM in the first data set in the users’ ISPPLIB concatenation.

An individual user can use the WINDOW command to select a default ISPF panel. For example a Tivoli Information Management for z/OS panel developer can change the default ISPF panel to administration with the following command:

```
WINDOW BLGISPFA,WINDOW SAVE
```

Refer to the *Tivoli Information Management for z/OS User’s Guide* for more information on using the WINDOW command.

Changing the Code for an ISPF Panel

You can change the code in an ISPF panel that displays an enhanced panel style window to:

- Add new actions to the action bar and corresponding pull-down choices
- Change the shipped action bar actions or corresponding pull-down choices
- Change the function keys.

See “Changing Keylists” on page 91 for information on changing the function keys.

To do the first two tasks, locate the ISPF panel you want to change. For this example, change BLGISPFE stored in INFO.SPECIAL.PANELS. Note that BLGISPFE is an ISPF panel, not a Tivoli Information Management for z/OS panel. Therefore, you cannot use PMF to make these changes. You must update this panel in the way you usually update data sets. For more information on updating ISPF panels, refer to the *ISPF Dialog Developer’s Guide and Reference*.

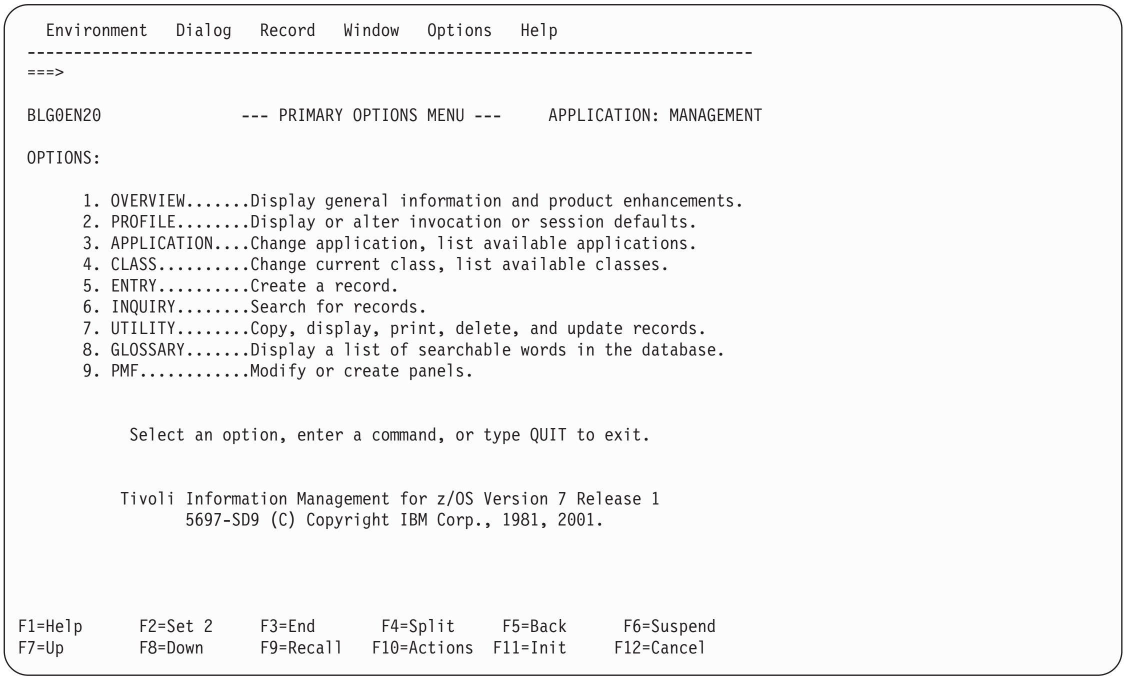
Note: You can change the panels using ISPF Dialog Tag Language. See “Using DTL to Change an Action Bar” on page 90 for more information.

Adding an Action to the Action Bar

In this scenario, you want to add an action called TSO to the action bar. The pull-down menu for TSO has three choices: **Time**, **Mail**, and **Profile**. **Time** is the default choice.

This is how ISPF panel BLGISPFE appears when you select it as your panel style. This is the way the panel is shipped with Tivoli Information Management for z/OS.

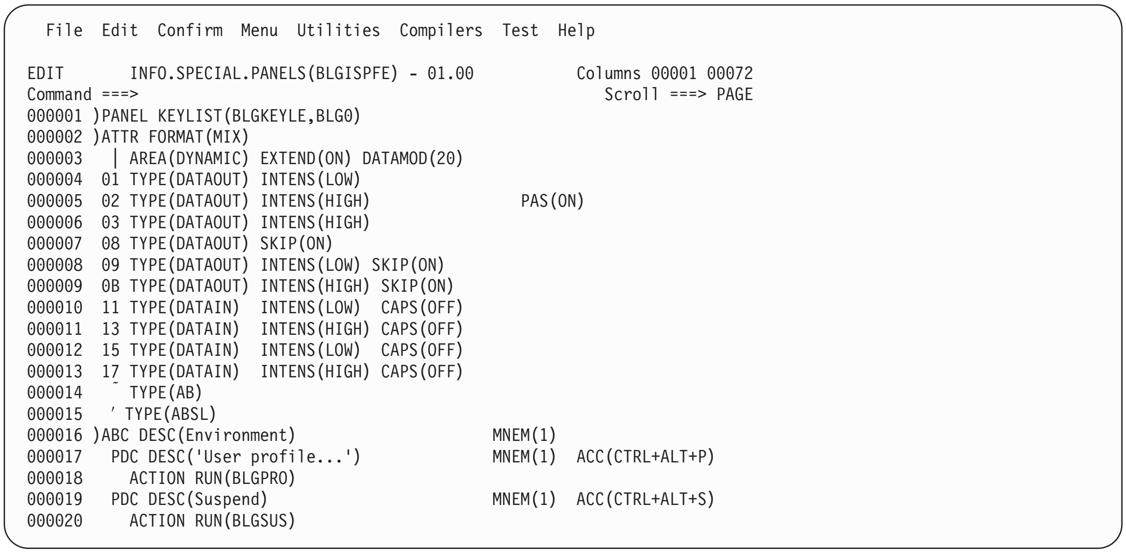
Refer to the *Tivoli Information Management for z/OS User’s Guide* for instructions on using the enhanced panel style.



```
  Environment   Dialog   Record   Window   Options   Help
 ------------------------------------------------------------------------------
 ===>

 BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

 OPTIONS:

       1. OVERVIEW.......Display general information and product enhancements.
       2. PROFILE........Display or alter invocation or session defaults.
       3. APPLICATION....Change application, list available applications.
       4. CLASS..........Change current class, list available classes.
       5. ENTRY..........Create a record.
       6. INQUIRY........Search for records.
       7. UTILITY........Copy, display, print, delete, and update records.
       8. GLOSSARY.......Display a list of searchable words in the database.
       9. PMF............Modify or create panels.


            Select an option, enter a command, or type QUIT to exit.


           Tivoli Information Management for z/OS Version 7 Release 1
                  5697-SD9 (C) Copyright IBM Corp., 1981, 2001.




F1=Help      F2=Set 2     F3=End       F4=Split     F5=Back      F6=Suspend
F7=Up        F8=Down      F9=Recall   F10=Actions  F11=Init     F12=Cancel
```

This is how the code for panel BLGISPFE appears on your screen when you are in PDF edit mode. This is the way the panel is shipped with Tivoli Information Management for z/OS.

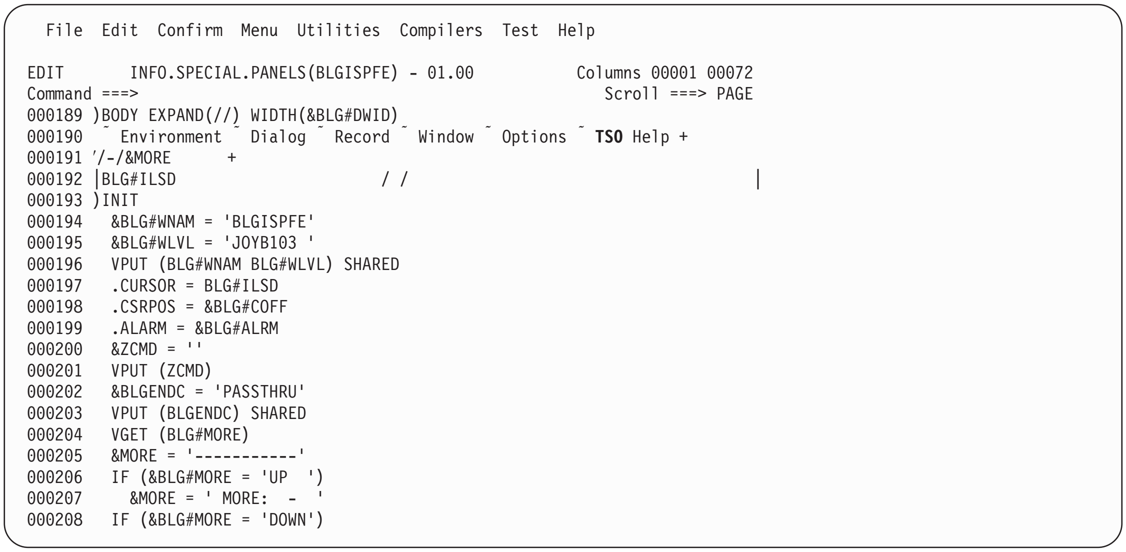
Locate the line that begins with **)BODY**.



```
  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help

EDIT       INFO.SPECIAL.PANELS(BLGISPFE) - 01.00          Columns 00001 00072
Command ===>                                                 Scroll ===> PAGE
000001 )PANEL KEYLIST(BLGKEYLE,BLG0)
000002 )ATTR FORMAT(MIX)
000003   | AREA(DYNAMIC) EXTEND(ON) DATAMOD(20)
000004  01 TYPE(DATAOUT) INTENS(LOW)
000005  02 TYPE(DATAOUT) INTENS(HIGH)              PAS(ON)
000006  03 TYPE(DATAOUT) INTENS(HIGH)
000007  08 TYPE(DATAOUT) SKIP(ON)
000008  09 TYPE(DATAOUT) INTENS(LOW) SKIP(ON)
000009  0B TYPE(DATAOUT) INTENS(HIGH) SKIP(ON)
000010  11 TYPE(DATAIN)  INTENS(LOW)  CAPS(OFF)
000011  13 TYPE(DATAIN)  INTENS(HIGH) CAPS(OFF)
000012  15 TYPE(DATAIN)  INTENS(LOW)  CAPS(OFF)
000013  17 TYPE(DATAIN)  INTENS(HIGH) CAPS(OFF)
000014    ˜ TYPE(AB)
000015   ' TYPE(ABSL)
000016 )ABC DESC(Environment)                   MNEM(1)
000017   PDC DESC('User profile...')            MNEM(1)  ACC(CTRL+ALT+P)
000018     ACTION RUN(BLGPRO)
000019   PDC DESC(Suspend)                      MNEM(1)  ACC(CTRL+ALT+S)
000020     ACTION RUN(BLGSUS)
```

On the line following the one that begins with **)BODY**, insert the text that you want to add to the action bar. Be sure to include attribute characters that tell ISPF how to display the text. For this example, you want **TSO** before **Help**, so insert this text before **˜ Help**:

˜ TSO



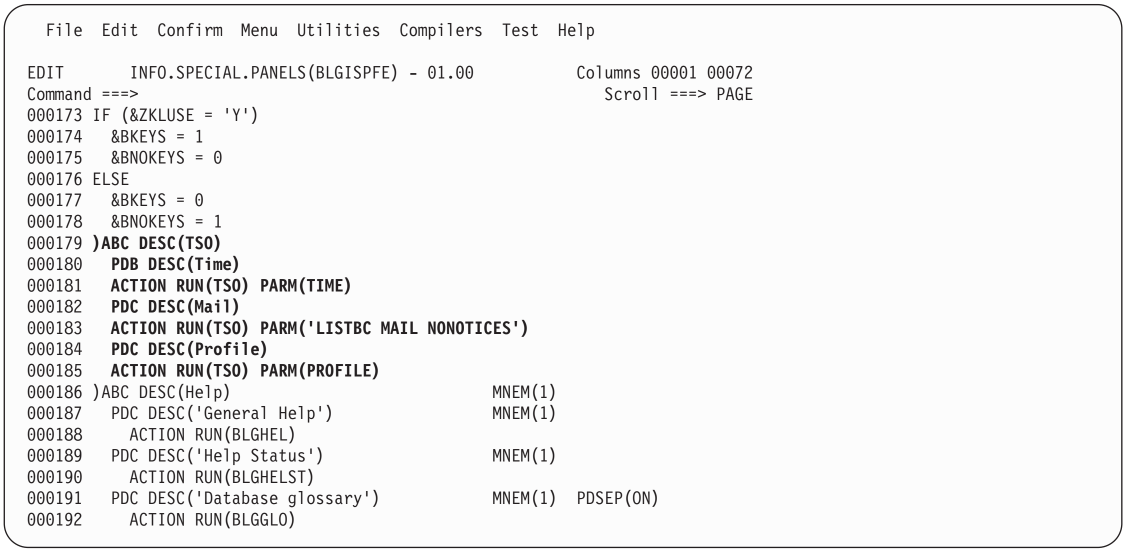
```
  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help

EDIT       INFO.SPECIAL.PANELS(BLGISPFE) - 01.00          Columns 00001 00072
Command ===>                                                 Scroll ===> PAGE
000189 )BODY EXPAND(//) WIDTH(&BLG#DWID)
000190   ˜ Environment ˜ Dialog ˜ Record ˜ Window ˜ Options ˜ TSO Help +
000191 '/-/&MORE      +
000192 |BLG#ILSD                    / /                                    |
000193 )INIT
000194   &BLG#WNAM = 'BLGISPFE'
000195   &BLG#WLVL = 'JOYB103 '
000196   VPUT (BLG#WNAM BLG#WLVL) SHARED
000197   .CURSOR = BLG#ILSD
000198   .CSRPOS = &BLG#COFF
000199   .ALARM = &BLG#ALRM
000200   &ZCMD = ''
000201   VPUT (ZCMD)
000202   &BLGENDC = 'PASSTHRU'
000203   VPUT (BLGENDC) SHARED
000204   VGET (BLG#MORE)
000205   &MORE = '-----------'
000206   IF (&BLG#MORE = 'UP  ')
000207     &MORE = ' MORE:  -  '
000208   IF (&BLG#MORE = 'DOWN')
```

Now you must add **)ABC, PDC**, and **ACTION** statements to create the pull-down menu. The **)ABC** statement must include exactly the same text that you just inserted before **Help**, but without the attribute characters. The **PDC** statements identify the choices on the pull-down menu. The ACTION statements say what to do when the user selects a particular choice.

The value of the DESC parameter on each PDC statement must be the exact text that you want to appear in the list on the pull-down menu. If the text contains multiple words, enclose the text in quotation marks. If the action is to run a Tivoli Information Management for z/OS command, the RUN() value must be **BLGxxxxx** (maximum of 8 characters) as defined in the BLG0CMDS table. If the action to be run requires parameters, use the PARM operand to pass them. For example, the ACTION statement for Time would be **ACTION RUN(tso) PARM(time)**. Use quotes when the PARM operand contains multiple parameters that are separated by spaces, for example **ACTION RUN(tso) PARM('listbc mail nonotices')**.

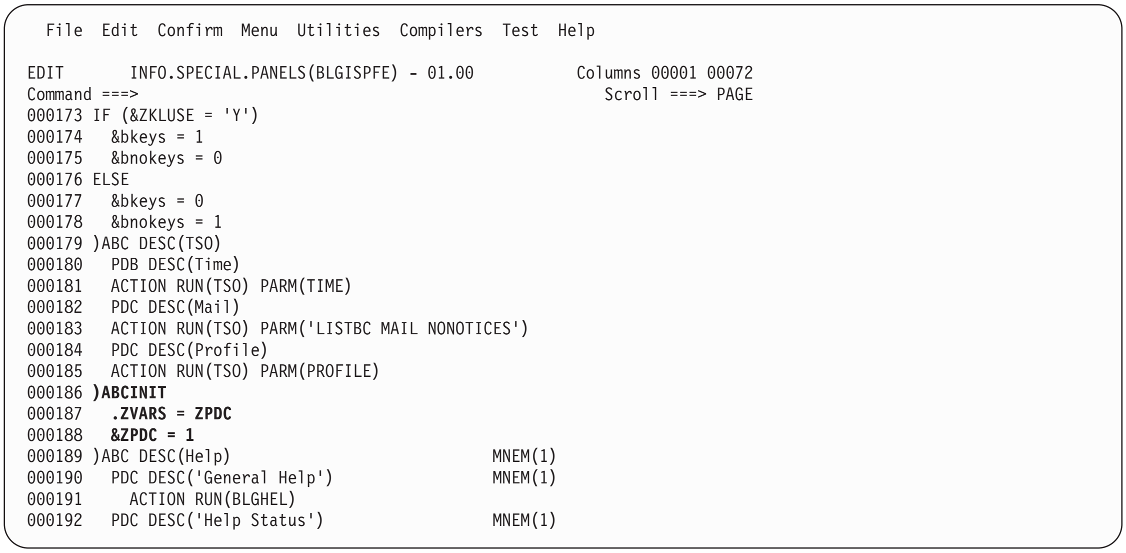
For this example, the **)ABC DESC** value is **TSO**. Each choice on the pull-down menu requires a PDC statement. And each PDC statement requires an ACTION statement to specify what to run when each choice is made.

Add these statements before the corresponding)abc section for Help. The statements are illustrated in the following panel. These statements were added by copying and modifying the corresponding **)ABC** section for **Help**. If you decide to enable the panel for GUI mode, verify that the values of the **MNEM** parameters on the PDC statements you add are appropriate. Refer to the *ISPF Dialog Developer’s Guide and Reference* for more information.



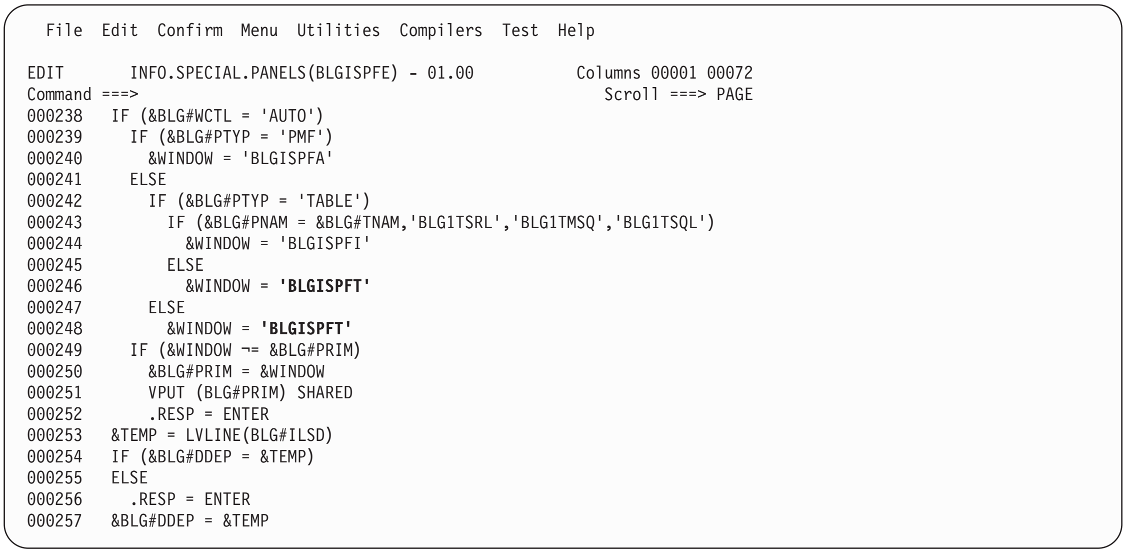
```
EDIT       INFO.SPECIAL.PANELS(BLGISPFE) - 01.00           Columns 00001 00072
Command ===>                                                  Scroll ===> PAGE
IF (&ZKLUSE = 'Y')
  &BKEYS = 1
  &BNOKEYS = 0
ELSE
  &BKEYS = 0
  &BNOKEYS = 1
)ABC DESC(TSO)
  PDB DESC(Time)
  ACTION RUN(TSO) PARM(TIME)
  PDC DESC(Mail)
  ACTION RUN(TSO) PARM('LISTBC MAIL NONOTICES')
  PDC DESC(Profile)
  ACTION RUN(TSO) PARM(PROFILE)
)ABC DESC(Help)                            MNEM(1)
  PDC DESC('General Help')                 MNEM(1)
    ACTION RUN(BLGHEL)
  PDC DESC('Help Status')                  MNEM(1)
    ACTION RUN(BLGHELST)
  PDC DESC('Database glossary')            MNEM(1)  PDSEP(ON)
    ACTION RUN(BLGGLO)
```

Follow the)ABC section you just added with an **)ABCINIT** section that duplicates the ZVARS and ZPDC assignment statements that are included on the other actions. The value that you assign to the **ZPDC** variable is the default choice value. In this example, assign a value of **1** to ZPDC to make **Time** the default choice. Add these statements before the **)ABC** section for HELP.



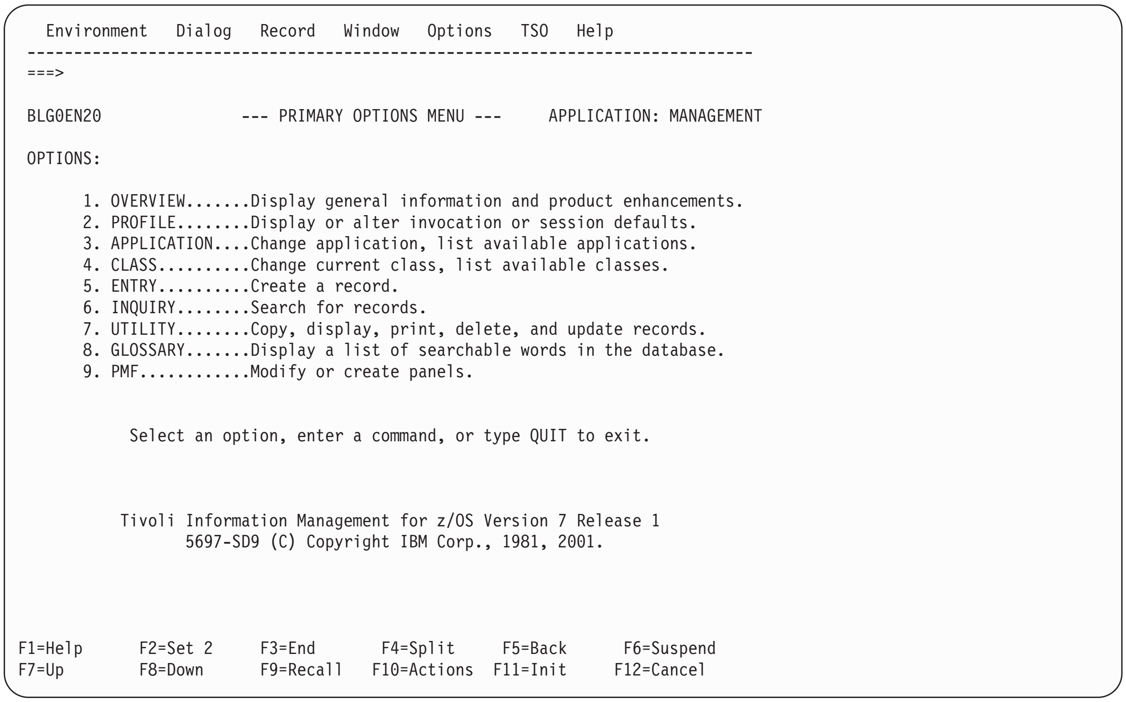
```
  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help

EDIT       INFO.SPECIAL.PANELS(BLGISPFE) - 01.00           Columns 00001 00072
Command ===>                                                  Scroll ===> PAGE
IF (&ZKLUSE = 'Y')
  &bkeys = 1
  &bnokeys = 0
ELSE
  &bkeys = 0
  &bnokeys = 1
)ABC DESC(TSO)
  PDB DESC(Time)
  ACTION RUN(TSO) PARM(TIME)
  PDC DESC(Mail)
  ACTION RUN(TSO) PARM('LISTBC MAIL NONOTICES')
  PDC DESC(Profile)
  ACTION RUN(TSO) PARM(PROFILE)
)ABCINIT
  .ZVARS = ZPDC
  &ZPDC = 1
)ABC DESC(Help)                            MNEM(1)
  PDC DESC('General Help')                 MNEM(1)
    ACTION RUN(BLGHEL)
  PDC DESC('Help Status')                  MNEM(1)
```

You can save the ISPF panel as **BLGISPFE** or save it under a different name. If you plan to save it under a different name, you must also update the statements that enable automatic selection of this window. See “Variables for Automating Window Selection” on page 82 for more information about these variables. For this example, save the panel as BLGISPFT. Assume that some of your users plan to use this as their primary window, so change all occurrences of **BLGISPFE** to **BLGISPFT** in this ISPF panel and in the other ISPF panels that automatically flow to this one.



```
EDIT       INFO.SPECIAL.PANELS(BLGISPFE) - 01.00          Columns 00001 00072
Command ===>                                                 Scroll ===> PAGE
IF (&BLG#WCTL = 'AUTO')
  IF (&BLG#PTYP = 'PMF')
    &WINDOW = 'BLGISPFA'
  ELSE
    IF (&BLG#PTYP = 'TABLE')
      IF (&BLG#PNAM = &BLG#TNAM,'BLG1TSRL','BLG1TMSQ','BLG1TSQL')
        &WINDOW = 'BLGISPFI'
      ELSE
        &WINDOW = 'BLGISPFT'
    ELSE
      &WINDOW = 'BLGISPFT'
  IF (&WINDOW ¬= &BLG#PRIM)
    &BLG#PRIM = &WINDOW
    VPUT (BLG#PRIM) SHARED
    .RESP = ENTER
&TEMP = LVLINE(BLG#ILSD)
IF (&BLG#DDEP = &TEMP)
ELSE
  .RESP = ENTER
&BLG#DDEP = &TEMP
```

If you copy **BLGISPFT** into **BLGISPFM**, you can make it the default for a group of users. Or an individual user can select it with the WINDOW command. Regardless of how the panel is selected, the adjacent window appears.



```
  Environment   Dialog   Record   Window   Options   TSO   Help
 -----------------------------------------------------------------------------
 ===>

 BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

 OPTIONS:

       1. OVERVIEW.......Display general information and product enhancements.
       2. PROFILE........Display or alter invocation or session defaults.
       3. APPLICATION....Change application, list available applications.
       4. CLASS..........Change current class, list available classes.
       5. ENTRY..........Create a record.
       6. INQUIRY........Search for records.
       7. UTILITY........Copy, display, print, delete, and update records.
       8. GLOSSARY.......Display a list of searchable words in the database.
       9. PMF............Modify or create panels.


            Select an option, enter a command, or type QUIT to exit.



           Tivoli Information Management for z/OS Version 7 Release 1
                  5697-SD9 (C) Copyright IBM Corp., 1981, 2001.




F1=Help      F2=Set 2     F3=End       F4=Split     F5=Back      F6=Suspend
F7=Up        F8=Down      F9=Recall   F10=Actions  F11=Init     F12=Cancel
```

If you want the TSO action to appear on the other windows in the enhanced panel style set (BLGISPFA and BLGISPFI), change those panels the same way you just changed BLGISPFE.

Changing the Action Bar Action or Pull-Down Choice

If you want a change to apply to more than one window, you must change each ISPF panel that you want to reflect the change. For example, if you want a change to appear only in the primary window, change only ISPF panel BLGISPFE. If you want a change to appear in the primary window and the inquiry window, change both BLGISPFE and BLGISPFI.

To change an existing action bar item, change the text on the line following the line that begins with **)BODY** and change the corresponding **)ABC DESC** statement. For example, to change the text of Environment to Profile, locate Environment in both places and replace it with Profile.

To change the text of a pull-down choice, change the text in the appropriate PDC DESC statement. For example, to change the text **Quit** to **Bye**, locate **Quit** and replace it with **Bye**.

To change the default choice for action, change the **ZPDC** assignment value within the appropriate **)ABCINIT** section. For example, to change the default Dialog choice from Recall to End, change the value of **ZPDC** in **)ABC DESC(Dialog)** section from **1** to **3**.

To delete an action, delete all the lines beginning with the **)ABC** statement that contains the action you want to delete and delete the lines in the corresponding)ABCINIT section. You must also delete the action from the line following the line that begins with **)BODY**.

To delete an individual choice from a pull-down menu, delete the two lines that contain the PDC and the ACTION statements. Note that this changes the ordinal choice number of any choices that follow the deleted choice. Therefore, if the ZPDC default was set for a following choice, decrement the value assigned to **ZPDC**.

For more information on changing ISPF panels, refer to the following publications:

- *ISPF Dialog Management Guide and Reference*
- *ISPF/PDF Edit and Edit Macros*
- *ISPF/PDF Guide and Reference*

Using DTL to Change an Action Bar

You can change the DTL source files for the ISPF panels, action bars, command table, and keylists instead of actually changing each individual ISPF panel. The source files are included in the sample library (SBLMSAMP) as follows:

GML Part	Source For:
BLGIGMLA	Panel BLGISPFA
BLGIGMLD	Panel BLGISPFD
BLGIGMLE	Panel BLGISPFE
BLGIGMLI	Panel BLGISPFI
BLGABDBG	Debug action bar
BLGABDIA	Dialog action bar
BLGABENV	Environment action bar
BLGABHLP	Help action bar
BLGABOPT	Options action bar
BLGABPMF	PMF action bar
BLGABREC	Record action bar
BLGABSCR	Scroll action bar
BLGABSER	Search action bar
BLGABSRC	SRC action bar
BLGABWIN	Window action bar
BLGIATTR	Common panel)ATTR section
BLGIINIT	Common panel)INIT section
BLGIPROC	Common panel)PROC section
BLG0EPSC	Tivoli Information Management for z/OS command table
BLG0EPSK	Tivoli Information Management for z/OS keylist table

Use the ISPF/PDF editor to modify these source files. Refer to the *ISPF Dialog Tag Language Guide and Reference* for complete information about changing the tags in these files.

Adding a New Action Bar

To add a new action bar definition to the BLGISPFE panel using the GML source files, follow these steps:

1. Create a new GML source file for the new action bar definition and use these tags: **AB, ABC, PDC**.
2. Update the BLGIGMLE source file (for BLGISPFE) by adding the include statement for the new action bar definition.
3. Use the DTL conversion utility (ISPDTLC) to convert the BLGIGMLE source file to an ISPF table.

4. Update the BLG0EPSC source file (for the Tivoli Information Management for z/OS command table, BLG0CMDS) by adding any new commands.
5. Use ISPDTLC to convert the BLG0EPSC source file to the command table, BLG0CMDS.
6. Test your modifications.
7. Move the modified BLGISPFE panel to your production ISPPLIB data set.
8. Move the modified BLG0CMDS command table to your production ISPTLIB data set.

Changing an Existing Action Bar

The steps required to change an action bar definition are similar to those for adding a new definition when you use the GML source files. For this example, you will see how to change the Options definition.

1. Update the BLGABOPT source file (for the Options definition).
2. Use ISPDTLC to convert the BLGIGMLA, BLGIGMLE, and BLGIGMLI source files.
3. Update the BLG0EPSC source file (for BLG0CMDS) by adding any new commands; use ISPDTLC to convert the BLG0EPSC source file, if necessary.
4. Test your modifications.
5. Move the modified BLGISPFA, BLGISPFE, and BLGISPFI panels to your production ISPPLIB data set.
6. If you changed BLG0CMDS, move it to your production ISPTLIB data set.

Changing Keylists

ISPF provides the ability to use stored sets of function keys, called *keylists* on specific ISPF panels. Therefore, each ISPF panel can have a different keylist, thereby enabling you to provide context-specific function keys. You can change which keylist a particular panel uses, or you can change the contents of a keylist.

Changing Which Keylist a Panel Uses

To change which keylist a particular panel uses, change the name after **KEYLIST** in the **)PANEL** statement in the ISPF panel. To use the enhanced panel style without a keylist, edit the **)PANEL** statement and delete the **KEYLIST** parameter. Without specifying a keylist, the panel uses the profile function key settings.

Note: You can use DTL to change the keylist a specific panel uses. Change or remove the keylist attribute on the PANEL tag and then convert the source file using the DTL conversion utility.

Changing the Contents of a Keylist

If you are changing the command to be issued when the key is pressed, determine whether the command is a Tivoli Information Management for z/OS command. If the command is not a Tivoli Information Management for z/OS command, follow the procedure in "Changing Function Keys on an ISPF Panel" on page 92. If the command is a Tivoli Information Management for z/OS command, you must first assign an alias to the command as described in "Assigning an Alias to a Tivoli Information Management for z/OS. Command" on page 92.

Note: You can change the Tivoli Information Management for z/OS command table and keylists by using ISPF Dialog Tag Language. See "Using DTL to Change an Action Bar" on page 90 for more information.

Assigning an Alias to a Tivoli Information Management for z/OS. Command

To assign an alias to a Tivoli Information Management for z/OS command, doing the following tasks:

1. Allocate both **DDNAME ISPTABL** and **DDNAME ISPTLIB** to the data set that contains the Tivoli Information Management for z/OS BLG0KEYS and BLG0CMDS tables.
2. Select the COMMANDS option from the UTILITIES panel (**3.9** if you are using the default ISPF panels). Enter **BLG0** as the application ID. You see a table of Tivoli Information Management for z/OS command aliases.

   Note: Do not change the BLG part of an entry. BLG associates the command with Tivoli Information Management for z/OS.
3. Look for an existing alias to use. For example, to use the Tivoli Information Management for z/OS PRINT command, locate:
   ```
   ALIAS BLG ;PRINT
   ```
   The alias name is BLGPRI. Use the alias name in the keylist **KEYnDEF** variable.
4. If the command you want to use does not already have an alias, add one to the table by doing the following:
   a. Insert a line into the command table.
   b. Create a new alias name (VERB). The name must be unique to the BLG0CMDS table, and the first 3 characters must be **BLG**.
   c. The ACTION must be of the following form:
   ```
   ALIAS BLG ;your command and parms
   ```
   Include the semicolon so that the command can be used when data is expected.
   d. Use the END command to exit the COMMANDS option. You can now use the alias in your keylist.
5. Use the END command to exit the TABLES option.

Changing Function Keys on an ISPF Panel

To change the contents of the function keys that are displayed on a specific ISPF panel, do the following tasks:

1. Identify the KEYLIST name and APPLID by looking at the **)PANEL KEYLIST** statement in the appropriate panel. For example, for the enhanced panel style, BLGISPFE, the keylist name is **BLGKEYLE** and the application ID is **BLG0**.
2. Allocate both DDNAME ISPTABL and DDNAME ISPTLIB to the data set that contains the Tivoli Information Management for z/OS BLG0KEYS and BLG0CMDS tables.
3. Use the **DIALOG TEST** option on the main ISPF panel, then the **TABLES** option (this is **7.4** if you use the default ISPF panels), find the row of the BLG0KEYS table that has KEYLISTN equal to the desired keylist name.

4. Type **3** to modify a row. To view the KEYLISTN value for the row, complete the panel as follows:
   a. Type the name of the keylist in the Table name field.
   b. Type **write** in the Open table field.
   c. Specify a value in the **By row number** field.
5. Locate the key you want to change. Each key has 3 variables:

   KEYnDEF
   Specifies the command to process when this key is pressed.

   KEYnLAB
   Specifies the label to display at the bottom of the screen.

   KEYnATR
   Specifies when or whether to display the label. This field is case sensitive; be sure to use all uppercase when you type your choice of label display.

   LONG
   Display the label only when all the keys are displayed.

   SHORT
   Always display the label.

   NO Never display the label.
6. Make your change and issue the **END** command to save it. The change you just made is effective for everyone who uses this table. If individual users want to change the function keys on a panel for their personal use, they can use the ISPF KEYLIST command.

Refer to the *ISPF Dialog Management Guide and Reference* for more information.

Adding Accelerator Keys and Separator Bars

You can add or change the accelerator keys and separator bars included in the panels. See "Using DTL to Change an Action Bar" on page 90 for instructions on changing action bar definitions.

ISPF has certain rules and recommendations for accelerator keys that you must adhere to when you create or change the definitions:

- You cannot assign the following single keys as accelerator keys. They must be used in combination with another key or keys.
  Ctrl
  Shift
  Alt
  Alphabetic characters A through Z
  Alphabetic characters a through z
  Numeric characters 0 through 9
- Only one key in a combination can be a function key.
- You cannot assign the Shift key with any alphabetic or numeric character (A-Z, a-z, 0-9).
- If you only use two keys for a combination, one key must be from the following list. If you use three keys for a combination, two of the keys must be from this list.
  Ctrl
  Alt
  Shift

  For either two- or three- key combinations, the other key must be one of these:

Insert
Delete
Backspace
Function keys F1 through F12
Alphabetic characters A through Z
Alphabetic characters a through z
Numeric characters 0 through 9

- You should avoid using the Alt key combined with only one character. (Use this combination for mnemonic access only.)
- You should avoid using a function key alone or Shift + a function key. (These combinations tend to cause confusion for users.)

Creating an Enhanced Panel Style Window

To create an enhanced panel style window, copy the BLGISPFE panel, or any other panel that was created using BLGISPFE as a model, to a new member in the appropriate ISPF panel data set. Give the new member the name that you want to use for selecting the panel. Then change the new member as described in “Changing the Code for an ISPF Panel” on page 83.

You can also add a keylist for a new panel. Refer to the *ISPF Dialog Management Guide and Reference* for details on how to do that.

Using the Standard Panel Style

To change the global default panel style to standard, copy BLGISPFD into BLGISPFM in the ISPPLIB concatenation.

An individual user can change his or her default to standard by issuing the following command:

```
WINDOW STANDARD,WINDOW SAVE
```

An individual user can temporarily switch to the standard style by issuing the following command:

```
WINDOW STANDARD
```

Refer to the *Tivoli Information Management for z/OS User’s Guide* for more information on the WINDOW command.

5

Sending E-mail Messages

This chapter describes how you can send e-mail messages through Tivoli Information Management for z/OS. The information provided here is for the program administrator responsible for implementing message notification in Tivoli Information Management for z/OS. You must understand terminal simulator EXECs (TSXs) and should know how to write REXX EXECs to perform the tasks described in this chapter. Additionally, you may need to become familiar with the use of remote data resources in a Tivoli Information Management for z/OS environment, and know how to use the operator commands that maintain remote data resources. (Remote data resources are described in this manual in “Remote Data Resources in Tivoli Information Management for z/OS” on page 335.)

Note: The type of message notification described in this chapter is different from the notification management methods described for earlier releases of Tivoli Information Management for z/OS. For a description of the former type of notification management (called ″immediate notification″ and ″problem escalation,″ see “Using Notification Management” on page 447). The type of message notification described in this chapter offers several advantages over the former methods of notification, and for that reason, is described here as the recommended way to implement message notification. If you still want to use the former methods, see “Using Notification Management” on page 447 for more information. If you are using the Integration Facility and using the former method of immediate notification, note that the information in this chapter does not apply to how messages are processed by that facility.

For more information about using TSXs in general, refer to the *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference*. For a description of the Integration Facility, refer to the *Tivoli Information Management for z/OS Integration Facility Guide*.

Message Notification

E-mail messaging, or ″message notification″ as described in this chapter, is a Tivoli Information Management for z/OS function that you can implement by creating model message records and writing or customizing TSXs to invoke messaging under conditions you specify.

By default, Tivoli Information Management for z/OS does not implement message notification in the Management application. However, you can create customized model message records by copying the sample records provided with Tivoli Information Management for z/OS. You can also customize the message notification TSXs or write your own based on the samples provided with Tivoli Information Management for z/OS.

The message notification function enables you to define various types of messages that can be sent for various purposes. For example, you can define messages to notify people of the following:

- A record has been assigned.
- Something has happened.
- Something is wrong or in need of attention (alert).
- A service level agreement (SLA) is being breached.
- A problem is being escalated.

Messages can be sent two ways:

- Immediately
- Through a message queue

With immediate message notification, a message is sent immediately to a specified user or group of users. You can implement immediate notification for various types of records, including problem, change, or activity records, or other records of your choice.

Through use of a Tivoli Information Management for z/OS remote data resource, messages can also be placed on a queue for processing. Message queues can be used in situations where you have a large number of messages to process and need to offload processing from the user's address space, or where immediate delivery is not a requirement. Tivoli Information Management for z/OS supports two message notification queues—a high-priority queue, and a low-priority (normal) queue.

Sample TSXs are provided with Tivoli Information Management for z/OS to produce different types of messages for various purposes. The TSXs and model message records provided create the following types of generic messages:

- “Record is assigned” message
- Escalation message

These TSXs are listed in “Message Processing with TSXs” on page 98. You can use the TSXs as provided, or create your own versions of them to suit your particular messaging needs.

Note: For performance reasons, it is recommended that you use the remote data resource to process all messages.

As provided, the message notification function is used only when you perform the tasks described in this chapter to implement it. You must customize your environment to send messages through use of the message notification function.

Features of the message notification function include the following:

- You can use TSXs to perform notification functions. TSXs and the inherent flexibility of REXX programming provide an easy way to construct complex mail messages. TSXs also allow for easier maintenance.
- Messages are sent using TCP/IP Simple Mail Transfer Protocol (SMTP). A mail message is constructed using a model message record and data from a filed record, and the

message is sent using TCP/IP SMTP. Messages routed through an SMTP server can be processed by a locally installed electronic mail package such as Lotus® Notes™. The SMTP server can reside on the mainframe (MVS) or in a distributed environment such as AIX®.

- Messages can be sent directly or added to a queue from which they subsequently will be sent.
- If necessary, message notification TSXs can also be called from TSPs.
- Message notification processing can occur for records filed through API processing.

Advantages of the message notification function over the former methods of notification management include:

- Ability to use an SMTP server on a platform other than MVS
- Flexibility in performing message processing
- Convenience of having different types of message queues
- Ease of defining the message content

Table 5 summarizes the basic functions provided by Tivoli Information Management for z/OS message notification.

Table 5. Message notification functions in Tivoli Information Management for z/OS

Type of message notification	TSXs used	Type of communication	General purpose	Required records	API support
Immediate notification	TSXs: BLGTXSND **BLGTNCNM** BLGTNMSG BLGTMRN0 **BLGTNMAN**	TCP/IP SMTP (mainframe or other platforms)	Process messages immediately. Ideal for sending a few messages right away where performance or volume is not an issue.	PEOPLE or USERS record for user's address space Message text record **BLGNMNOT**	Through HL14
Queued notification	TSXs: BLGTXQUE **BLGTNCNM** BLGTNMSG BLGTMRN0 **BLGTNMAN** BLGTNRQM BLGTNPQM	TCP/IP SMTP (mainframe or other platforms)	Place messages on a high priority or low priority queue for processing (offloads processing from user's address space). Uses a remote data resource.	PEOPLE or USERS record for user's address space Message text record **BLGNMNOT**	Through HL14
*Items shown in bold are samples provided with Tivoli Information Management for z/OS that are intended for your use and customization. TSXs not shown in bold represent REXX EXECs not intended for customization.					

The generic text used to send messages can be created interactively through Tivoli Information Management for z/OS panels; however, the recommended way to build notifications is through the TSXs and model message records provided or through a high-level API transaction (HL14 Start User TSP or TSX) as described in the following sections.

Deciding on a Message Process

Because the message notification process described here relies on the use of REXX EXECs which can be easily copied and customized, you should decide on the following before implementing message notification:

- Who should receive the messages? Should anyone be copied on the messages?
- Do e-mail addresses exist in the Tivoli Information Management for z/OS database for these people? Is there a people or users record that can be used to extract this information?
- Under what circumstances should the people be notified? When a particular type of record is filed? Or is there some other condition?
- It is recommended that you queue messages. What is the criteria for placing messages on a high-priority queue versus a normal queue?
- What should the message text say (what is the generic message text)? How many model message text records do you need?
- What type of information should be extracted from a record and assembled into the message text record? What types of records are needed?
- What SMTP server will be used to send the notification to the recipient?

After you have answered these questions, you will have a better understanding of what your business process needs are, and you can begin implementing the message notification process with Tivoli Information Management for z/OS.

Message Processing with TSXs

The following overview describes the basic steps of message processing.

1. When a record (such as a problem record) is filed, the file time control panel for the record calls a TSP which sets parameters and links to a TSX which starts the message notification process. Tivoli Information Management for z/OS provides sample TSXs which can be called for this purpose (BLGTXSND for immediate notification, and BLGTXQUE for queued notification). You can use these samples to create messages for various situations.

Table 6. Sample TSXs provided to create messages (available in SBLMSAMP sample library)

TSX Sample	Purpose
BLGTXSND	Invokes the immediate notification process
BLGTXQUE	Invokes the queued notification process
BLGTNCEM	Creates an escalation message
BLGTNCNM	Creates a notification message

2. A message is created through a message creation TSX. Tivoli Information Management for z/OS provides two TSXs for this purpose: BLGTNCNM, which creates a basic notification message, and BLGTNCEM, which can be used to create an escalation message. For the purposes of this message processing overview, the use of the BLGTNCNM TSX is described here.

The BLGTNCNM TSX creates a notification message record. It does this by copying the model notification message record (BLGNMNOT) provided with Tivoli Information Management for z/OS and updating information in the copy to create a new message text record.

The BLGTNCNM TSX passes the record number ID to the BLGTNMSG TSX, which actually does most of the work. The BLGTNMSG TSX is a utility TSX that constructs the message record using data it retrieves from the record number ID and text from the message text record model (BLGNMNOT). It also gets and sets a unique record number ID for the message record being created. Message record IDs are generated with an ″at″ sign (@) as the first character.

3. The BLGTNMAN TSX connects to an SMTP server and sends the notification message using TCP/IP. The address of the SMTP server must be specified in this TSX. The address is the SMTP server hostname or the IP address, and the port number. You can call this EXEC interactively from a TSX or TSP, or through an HL14 (Start User TSP or TSX) transaction.

4. If queueing is used, messages are added to a queue through use of a Tivoli Information Management for z/OS remote data resource.

   A remote data resource is required and must be active for message queueing to occur. A remote data resource is opened when the BLGTNRQM TSX is run. The BLGTNRQM TSX is a TSX that is invoked by starting Tivoli Information Management for z/OS as a batch job. When this TSX is started, it opens a remote data resource in Tivoli Information Management for z/OS and then constantly monitors the queues for work to appear. The following takes place:

   - The BLGTNPQM TSX puts message information on a remote data resource for queueing. This information can consist of record number IDs for the message text records, or the data required to create a message record. For example, a record number ID for a problem record can be passed to the BLGTNPQM TSX with the name of the queue for the message, and the name of the TSX which should be called to create the message (and any additional parameters required by that TSX). For example: `RUN BLGTNPQM` *`rnid qname`* `(`*`tsxname tsxparms`*.

     Two types of queues are provided: a high-priority (hot) queue, and a low-priority (normal) queue. If the message is considered ″hot″, it is placed in the high-priority queue; if it is a low-priority message (or if no queue is specified), it is placed in the normal queue. The record number IDs are placed in the appropriate queue for the BLGTNRQM batch process to pick up and send the messages.

   - The BLGTNRQM TSX queries the queues to read the messages, retrieves the message data, and processes the data in batch mode. Data from the high-priority queue is processed before the normal queue. If the record read from the queue is a message text record ID, the BLGTNMAN TSX is called to send the message. If the record read contains the name of a TSX, that TSX is called to perform additional processing. When the TSX ends, the BLGTNMAN TSX is called to send the message created by the TSX.

5. After messages are processed, they are sent to the recipient by the SMTP server through a TCP/IP connection. TCP/IP SMTP routes the mail using your installation’s mail configuration. After messages reach this point, Tivoli Information Management for z/OS no longer has control of the message.

Figure 3 on page 100 provides an illustration of how messages are processed for immediate delivery when the samples provided with Tivoli Information Management for z/OS are used.

In contrast, Figure 4 on page 102 shows how messages can be queued using the TSXs provided.

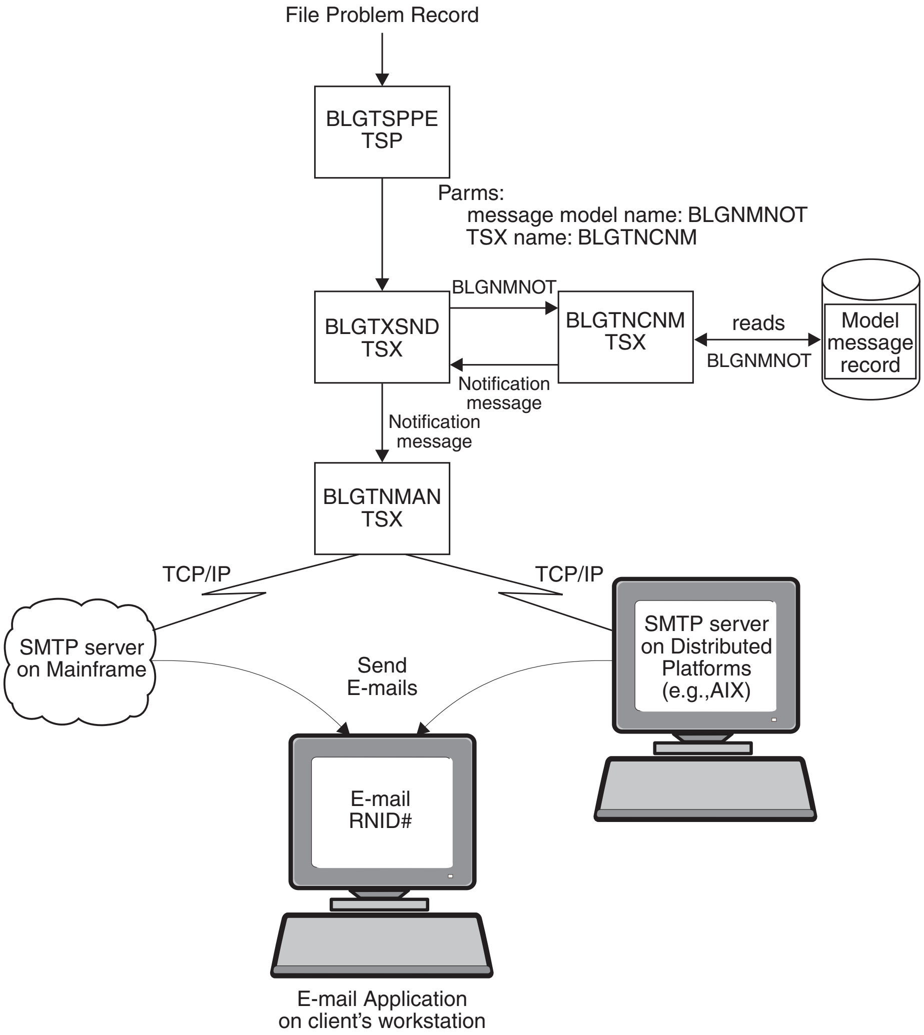


Figure 3. Immediate or "on demand" message notification processing with samples provided by Tivoli Information Management for z/OS

In Figure 3, the control panel runs TSP BLGTSPPE when a problem record is filed. The sample BLGTXSND TSX (or your equivalent) is called when the record is filed. A record number ID (RNID) is passed to the BLGTNCNM TSX (also a sample), which creates the notification message. BLGTNCNM creates the message by copying the model message text record (BLGNMNOT or one you specify) and adding address information to the message. The TO data in the message record is derived from the assignee name in the problem record and people records are checked to see if an e-mail address exists for that assignee. The CC and BCC fields can be derived from the message text record if address values are supplied in the BLGNMNOT record, or you can alter the sample BLGTNCNM TSX to have it obtain the information from another source. (Your business process should dictate how you make use of these fields.) The sample BLGTNCNM TSX as provided does not use CC or BCC data, so those fields are deleted from the message record. The BLGTNMAN TSX is then called to send the message to the SMTP server. After the message appears on the server, it is processed using your local e-mail application (such as Lotus Notes).

In Figure 4 on page 102, messages are processed through use of a remote data resource queue. Sample TSX BLGTXQUE runs when problem records are filed. It passes a record number ID, queue name, and the name of a TSX to construct the message (here, BLGTNCNM) to the BLGTNPQM TSX, which places the information on the specified queue. The BLGTNRQM TSX, which runs constantly, reads the record on the queue and processes it, calling any other TSXs as necessary. This is the recommended method for processing messages.

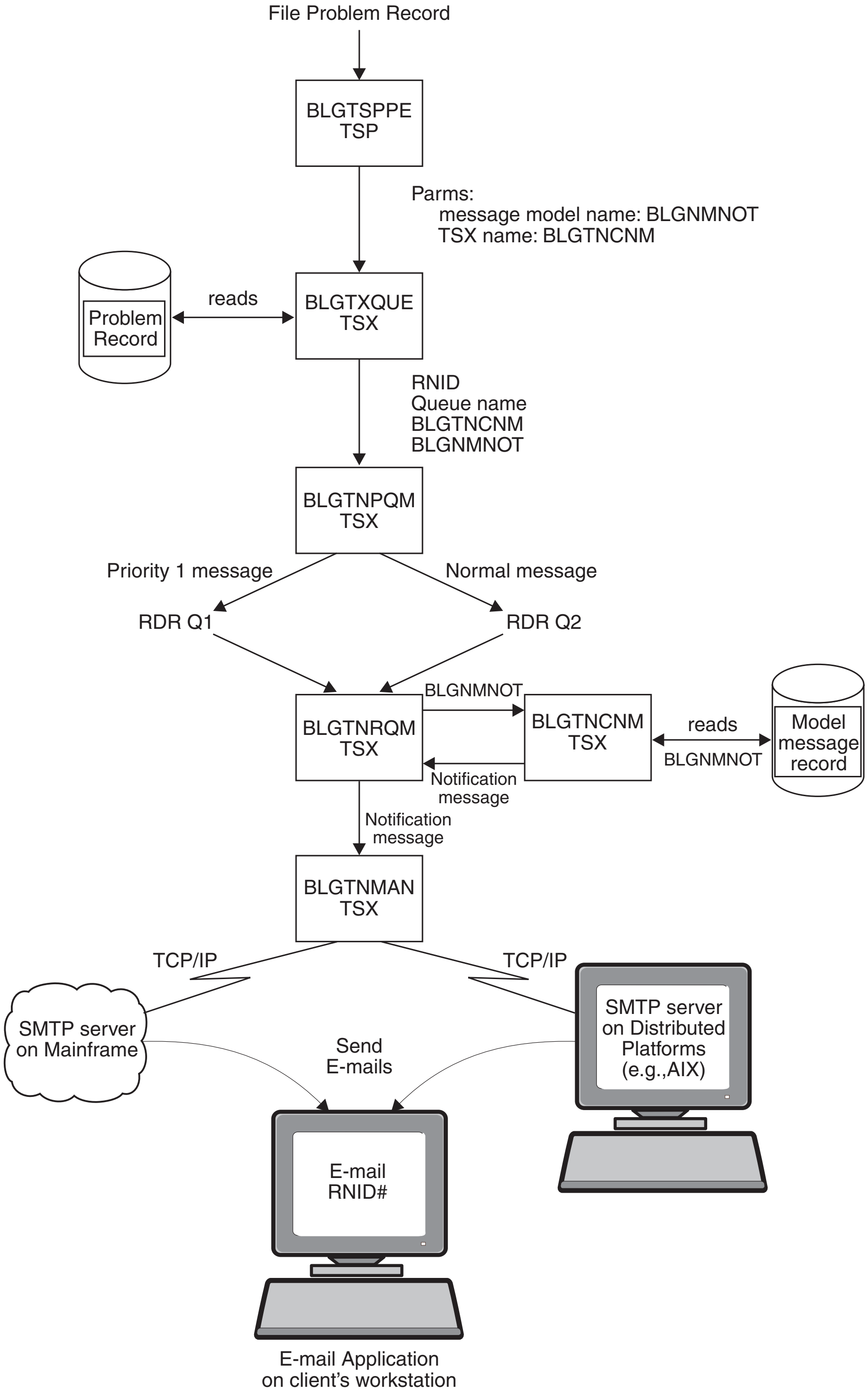


Figure 4. Queued message notification with samples provided by Tivoli Information Management for z/OS

Understanding Model Message Text Records

Model message text records are required to use the message notification TSXs. A model message text record is a special type of record in the Tivoli Information Management for z/OS database that stores notification information such as:

- Type of message (e-mail)

- E-mail addresses
- Priority of the message (01 – high, or 02 – normal/low priority)
- Text of the message and message inserts

The following model message text records are provided with Tivoli Information Management for z/OS:

- BLGNMMSG – Generic model message text record
- BLGNMNOT – Model message text record to inform someone that a problem record is assigned
- BLGNMCHG – Model message text record to inform someone that a change record is assigned
- BLGNMACT – Model message text record to inform someone that an activity record is assigned

These records are available in the SBLMRCDS data set and are loaded into your database when you load data model records. (For a description of how to load records provided with Tivoli Information Management for z/OS, refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference*.)

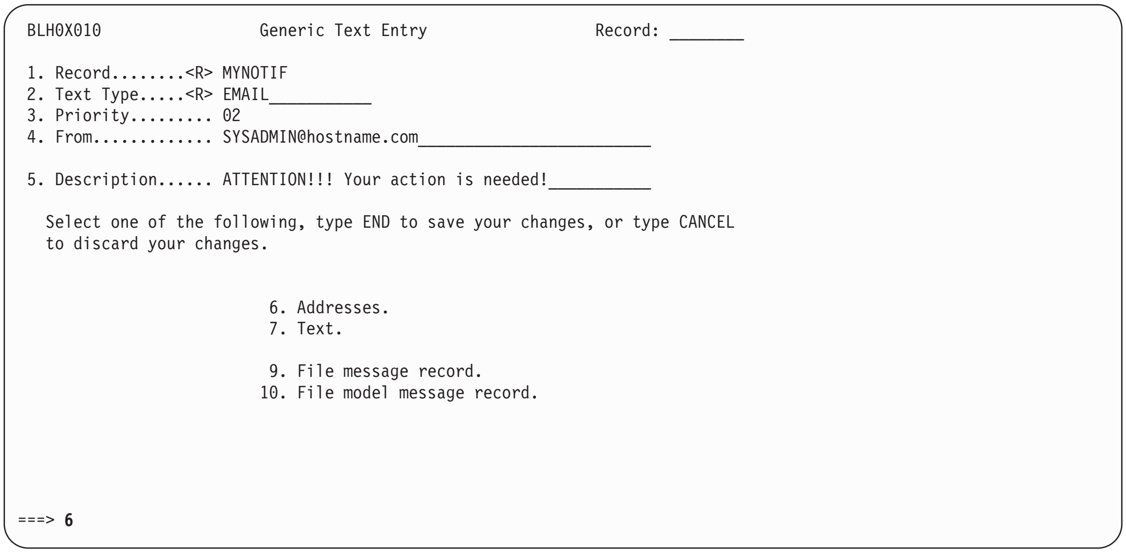
You can copy these model message text records using the Tivoli Information Management for z/OS COPY command and create new message text records from the models. You can assign your own record ID to the new copy of the record. Do not use the ″at″ sign symbol (@) as the first character of your model records. The ″at″ sign is reserved for the actual messages that are sent to users.

Note: Although the model message text records remain in your database, the actual notification messages that are sent through the SMTP server to the recipient do not reside in the database. Notification messages are temporary in nature and can potentially take up a lot of space in the database if stored over time. Therefore, they are not stored as permanent records and cannot be manipulated the same way you can manipulate other records in the database.

To copy a model message text record, enter the COPY command at the Tivoli Information Management for z/OS command line. For example:

```
;COPY R BLGNMNOT
```

On the panel which is then displayed (BLH0X010, Generic Text Entry), type your own record number ID in the **Record** field. Specify the type of notification for which the record will be used (currently only EMAIL is supported). Optionally, enter data for other fields as desired by overtyping the field data displayed from the model. For example, you can specify the priority (01 = high, 02 = normal/low), enter an e-mail address for the sender of the message in the From field, and include a description (message subject) in the **Description** field.



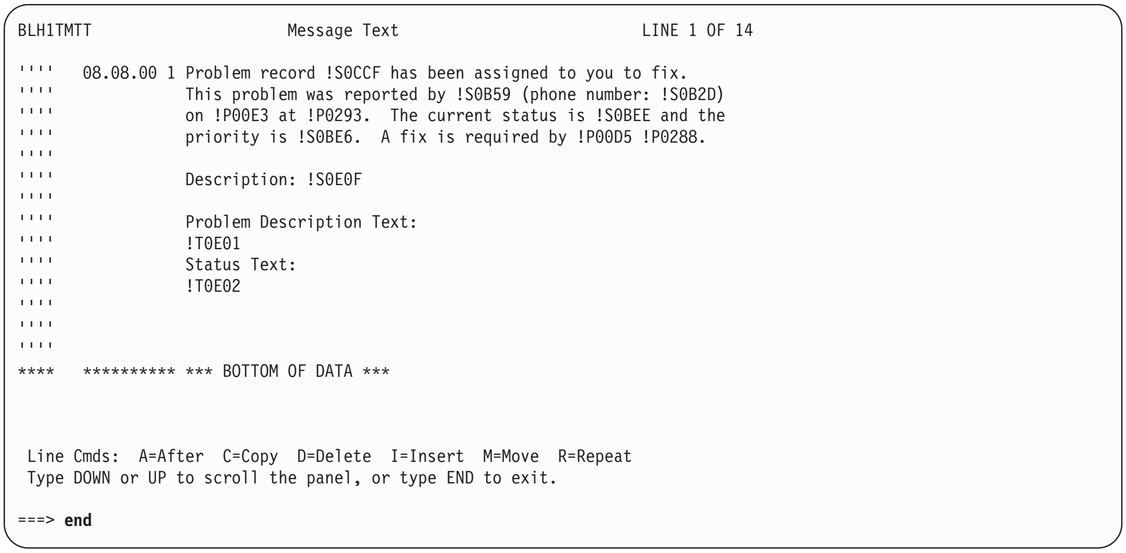
```
 BLH0X010               Generic Text Entry                Record: _______

 1. Record........<R> MYNOTIF
 2. Text Type.....<R> EMAIL__________
 3. Priority......... 02
 4. From............ SYSADMIN@hostname.com_______________________

 5. Description...... ATTENTION!!! Your action is needed!__________

   Select one of the following, type END to save your changes, or type CANCEL
   to discard your changes.


                         6. Addresses.
                         7. Text.

                         9. File message record.
                        10. File model message record.





===> 6
```

To supply text for the model, type **7** in the command line on panel BLH0X010 and press Enter. In this example, address information is selected.

Note: Do *not* change the address information (option 6) for the model record you are creating. This is a skeleton entry that is used to add the Send to, CC (carbon copy), and BCC (blind carbon copy) information by the TSX to the address data for the message record. Be sure to use option 10 to file your new model record.

On the resulting Message Text panel, supply the standard message text that should appear in messages using that model. Type the text you want displayed to the recipients and supply the necessary s-word or p-word index values to retrieve the appropriate pieces of information from the record.

The next screen example shows the message text as used in the BLGNMNOT model message text record. You can overtype the text and supply other s-word or p-word index values by entering !S*nnnn* and !P*nnnn* to indicate the values. Substitute the correct value for *nnnn*.



```
BLH1TMTT                    Message Text                        LINE 1 OF 14

''''   08.08.00 1 Problem record !S0CCF has been assigned to you to fix.
''''              This problem was reported by !S0B59 (phone number: !S0B2D)
''''              on !P00E3 at !P0293.  The current status is !S0BEE and the
''''              priority is !S0BE6.  A fix is required by !P00D5 !P0288.
''''
''''              Description: !S0E0F
''''
''''              Problem Description Text:
''''              !T0E01
''''              Status Text:
''''              !T0E02
''''
''''
''''
****   ********** *** BOTTOM OF DATA ***



 Line Cmds:  A=After  C=Copy  D=Delete  I=Insert  M=Move  R=Repeat
 Type DOWN or UP to scroll the panel, or type END to exit.

===> end
```

The BLGTNMSG TSX interprets the !S, !P, and !T and uses the associated s-words and p-words to retrieve the corresponding data from the record during message processing. !S and !P indicate that the FINDSDATA TSX control line should be used to retrieve the data. !T indicates that the GETTEXT control line should be used to retrieve the freeform text data.

The example showed a t-word index value for the Problem Description Text and Status Text, both of which are freeform text fields. T-word values do not really exist in Tivoli Information Management for z/OS; they are used here as symbols to represent that type of freeform text.

In this example, the actual message would be displayed as follows after processing takes place to substitute actual data for the items represented as s-word and p-word index values.

```
Problem record 123456 has been assigned to you to fix.
This problem was reported by HOLLOWAY (phone number: 555-8383)
on 03/15/2001 at 09:00.  The current status is OPEN and the
priority is 01.  A fix is required by 03/15/2001 15:00.

Description: Web server is down

Problem Description Text:
Server went down and customers cannot pull information for
recently announced widget product.
Status Text:
Called Bob Torres to preliminary investigation.
```

When you are finished entering text on panel BLH1TMTT, type **end** and press Enter to return to the Generic Text Entry panel, where you can type **9** to file your message text record. The model message text record is saved in the database with the record number ID you specified.

You can make copies of BLGNMNOT and assign different record number IDs to them for various purposes. For example, if you have five critical business applications with service-level agreements, you may want to set up five customized alarm notifications. Each generic message record could have a customized subject line (**Description** field) reflecting the name of the particular business application. This way, when recipients are going through their list of e-mail messages, they can immediately see what business application requires attention and take the appropriate action.

The fields that make up a generic message text record are shown in Table 7.

Table 7. Generic message text record fields

Field Name	Prefix	Prefix Index (P)	S-Word Index	Length
Generic text record type			S12D1	
Record ID	RNID/	P01EA	S0CCF	8
Text type	TTYP/	P0505 (EMAIL) P0506 (MESSAGE) P0507 (FAX) P0508 (PAGE)	S12D2	–
Priority	PRIO/	P01B7	S0BE7	2
From	FROM/	P0500	S12D3	45
To	TO/	P0501	S12D4	45

Table 7. Generic message text record fields (continued)

Field Name	Prefix	Prefix Index (P)	S-Word Index	Length
CC	CC/	P0502	S12D5	45
Pager PIN number	PPIN/	P0504	S12D7	16
Description (message subject)			S0E0F	45
Text of message			S0E09	(Freeform text)
Date entered	DATE/		S0C34	10
Time entered	TIME/		S0C61	5
Class created	CLAE/		S0BB1	8
Date modified	DATM/		S0C35	10
Time modified	TIMM/		S0C62	5
User modified	USER/		S0B5E	8

Setting Up for Message Notification

Before using message notification, you need to perform the following setup tasks described in this section:

- Define the location of the TCP/IP SMTP server to Tivoli Information Management for z/OS in the BLGTNMAN TSX.
- Set up a remote data resource (required only if message queues will be used).
- Create message text records that will be used as models or templates for the actual messages.
- Create TSXs to create messages from the models.
- Update file time control panels to use the message TSXs.

Defining TCP/IP SMTP Server Information (Editing the BLGTNMAN TSX)

To define the address of the TCP/IP SMTP server, you need to edit the BLGTNMAN TSX. (It is recommended that you make a copy of this TSX for backup purposes, so that if you need to return to the original TSX for any reason, you still have an original version of the TSX.) Customize the TSX to specify the SMTP server hostname (or the IP address) and the port number for the server. The BLGTNMAN TSX is provided in the SBMLTSX data set. Edit the TSX and change the following area:

```
/******************* USER MODIFIABLE AREA ****************************/
/* Beginning of user modifiable area.                                */
/*********************************************************************/

  smtpServer = '000.000.000.000'  /* Set SMTP server address         */
  smtpPort = '25'                 /* SMTP default port number        */

  /*******************************************************************/
  /* These values may need to be adjusted to handle slow networks    */
  /* and SMTP server response time.                                  */
  /*******************************************************************/
  waittime = 1                    /* Seconds to wait for reply from
                                           READSOCKET                */
  maxwait  = 20                   /* Max seconds to wait for reply
```

```
                                   from READSOCKET before trying to
                                   resend the data               */
  sendtries = 2                 /* Number of times to try sending
                                   message                       */

/*********************************************************************/
/*           End of user modifiable area.                            */
/*********************************************************************/
```

Change the smtpServer line to specify an SMTP hostname or IP address (for example, `123.45.678.9` or `mysystem.mycompany.com`). Include only one TCP/IP SMTP server address.

Specify the port number for the SMTP server. The default port number is 25.

You may need to adjust the values for **waittime**, **maxwait**, and **sendtries** to handle slow response times from the network and the SMTP server. **Waittime** specifies how many seconds to wait between READSOCKET control line calls when a reply is expected from the SMTP server. **Maxwait** specifies the maximum number of times to issue the READSOCKET control line without receiving a response from the SMTP server before a ″time out″ condition is reached. **Sendtries** specifies the number of times to try sending a message to the SMTP server.

After completing the remaining setup tasks, you can call the BLGTNMAN TSX as described in “Sending the Notification (Using the BLGTNMAN TSX)” on page 110.

Defining the Escalation Message Address (Editing the BLGTNCEM TSX)

To define the source address of escalation messages, you need to edit the BLGTNCEM TSX. The address source you specify does not need to a valid e-mail address; its only purpose is to let the user know who sent the escalation message. For example, `ESCALATION@MVS1` might be an appropriate source address. Customize the BLGTNCEM TSX to specify the sender of the escalation message. The BLGTNCEM TSX is provided in the SBLMTSX data set. Copy the TSX to your user TSX dataset and edit the TSX to change the following area:

```
/*********************************************************************/

/* Modify 'from@address' data. from should be changed to contain an  */
/* addresss indicating where the Escalation message is coming from.  */
/* This information is included in the E-mail message.               */

/*********************************************************************/
   from = 'from@address'                /* Set from addres value   */
```

Note: You must provide a new escalation message source address. Do *not* leave the default. Leaving the default (`'from@address'`) as the escalation message source address will result in an error in the BLHTMVAL TSX.

Setting Up the Remote Data Resource (Using the BLGTNRQM TSX)

If you want to queue messages or define messages as high priority or normal messages, you must set up a remote data resource in Tivoli Information Management for z/OS. Setting up a remote data resource for message queueing involves the following tasks:

- Start Tivoli Information Management for z/OS in batch mode and specify that the BLGTNRQM TSX should run after Tivoli Information Management for z/OS is initialized. The BLGTNRQM TSX runs through batch job submission.

For example, your batch mode startup JCL would contain the following statements:

```
PROFILE PREFIX(userid)
ISPSTART PGM(BLGINIT) +
PARM('SESS(00) TSP(BLGTNRQM) IRC (QUIT)')
```

Note: If you are using partitions, you must also specify a value for `CLASS()`.

You can copy the BLGTNRQJ sample JCL from the SBLMSAMP sample library. When you run BLGTNRQJ, it runs BLGTNRQM as a batch job, which invokes ISPF under TSO. The BLGTNRQM TSX runs and then quits after it is finished running. The BLGTNRQM TSX should be a member of a partitioned data set allocated to the BLGTSX DD statement.

In this example JCL, the BLGSES00 session member is used.

For more information on starting Tivoli Information Management for z/OS in batch mode, refer to the section on starting Tivoli Information Management for z/OS in the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference*.

If you need to terminate the BLGTNRQM batch job, you can use the Tivoli Information Management for z/OS operator command RDR. The RDR command stops the job and flushes the queues used for message notification. For example:

```
/F procname,RDR,FLUSH=BLGTNMQ1
/F procname,RDR,FLUSH=BLGTNMQ2
```

procname is the name of the started catalog procedure that identifies the BLX-SP to which the BLGTNRQM job is connected.

BLGTNMQ1 is the queue for high-priority messages. BLGTNMQ2 is the queue for all other (normal/low priority) messages.

For more information about the RDR operator command, refer to the *Tivoli Information Management for z/OS Operation and Maintenance Reference*.

Create Model Message Text Records

As described in “Understanding Model Message Text Records” on page 102, create the model message text records that will meet your business process needs. You can copy the sample models provided and adjust them as necessary to create your own models.

Create TSXs to Support Your Message Notification Process

It is suggested that you create TSXs that create the messages you want to send. TSX BLGTNCNM is a sample TSX that creates notification messages from model message records. You can modify this sample and make any necessary changes to suit your business needs. If you decide to modify this sample, it is suggested that you first make a copy of the sample TSX so that you have a backup copy available for later reference.

The following section describes how you can use the BLGTNCNM TSX.

Creating Messages with the BLGTNCNM TSX

You can create a notification message to inform someone of an assigned problem by calling the BLGTNCNM TSX from a TSX. For example, a TSX can call the BLGTNCNM TSX through a LINK control line:

```
Call BLGTSX 'LINK','BLGTNCNM',rnid
```

rnid is the record number ID for the record being filed. Your TSX can pass the BLGTNCNM TSX this value.

You need to update the TSPs that run when a record is filed in order to set the TSX parameter values and invoke the correct TSX. Use the following tables to determine what changes to make. As shipped, the immediate notification TSPs perform queued message processing. They use TSX BLGTXQUE to queue the messages and TSX BLGTNCNM to create the message records.

Table 8. Tivoli Information Management for z/OS Immediate Notification TSPs and TSX Equivalents

TSP name	TSX name	Purpose	Invoked by
BLGTSPPE	BLGNMNOT	Problem create notification	BLG1A111
BLGTSPPU	BLGNMNOT	Problem update notification	BLG1A111
BLGTSPCE	BLGNMCHG	Change create notification	BLG1A121
BLGTSPCU	BLGNMCHG	Change update notification	BLG1A121
BLGTSPAE	BLGNMACT	Activity create notification	BLG1A131
BLGTSPAU	BLGNMACT	Activity update notification	BLG1A131

If you want to change the parameter values, update the MOVEVAR line and change the message model name and the message create TSX name.

If you want to use immediate notification processing, update the LINK control line and specify TSX BLGTXSND instead of BLGTXQUE.

Specifying Use of a Queue (Using the BLGTNPQM TSX)

To cause a notification message to be routed to a remote data resource and queued for batch processing, you can use the BLGTNPQM TSX. With this TSX, you can specify the ID of a record in the database (such as a problem record), and the name of a TSX to be called to create the message.

The TSX name specified when running the BLGTNPQM TSX creates the message with the desired information (FROM, TO, CC, SUBJECT data).

You can call the BLGTNPQM TSX from a TSP, a TSX, through the RUN command, or from the API through the HL14 Start User TSP or TSX transaction.

From a TSX or TSP

A TSX can call the BLGTNPQM TSX through a LINK control line. For example:

```
Call BLGTSX 'LINK','BLGTNPQM',rnid qname (tsxname tsxparms)
```

Additional parameters you can include in the LINK control line are:

- *rnid* – Specify the record number ID of the record to be placed on the message queue. This can be the message text record ID or another type of record ID in the database (such as a problem record ID). If a TSX name is also specified in the LINK statement, the record number ID is passed as an input parameter to the TSX so that information from the record can be used to construct the message. The record number ID is required.
- *qname* – Specify the queue name for the message (optional):

- Q1 = high-priority queue
- Q2 = low-priority (normal) queue

If no queue name is specified, the message is placed on the normal queue.

- *tsxname* – Specify the name of a TSX to be called to create a message text record. For example, you can specify BLGTNCNM as the TSX if you wanted to use the BLGTNCNM sample to send a notification message. This TSX should set a zero (0) return code in the TSCAVDA if the message record was created. The TSX name is optional.
- *tsxparms* – Specify any additional parameters to be passed to the TSX specified here. TSX parameters are optional, but if you specify them, you must also specify a TSX name.

Using the RUN Command

Enter the RUN command and specify a record number ID. You can optionally specify additional parameters as previously described.

RUN BLGTNPQM *rnid qname (tsxname tsxparms)*

From the HLAPI

Use the HL14 Start User TSP or TSX transaction and pass the following input parameters:

- TSP_NAME – Specify BLGTNPQM as the TSX name. The value BLGTNPQM is required.
- RNID_SYMBOL – Specify the record ID of the record to be placed on the message queue. This can be the message text record ID or another type of record ID in the database (such as a problem record ID). The record ID is required.
- Q_NAME – Specify the queue name (optional).
  - Q1 = high-priority queue
  - Q2 = low-priority (normal) queue

  If no queue name is specified, the message is placed on the normal queue.
- TSX_NAME – Specify the name of the TSX to be called (optional).
- TSX_PARAMETER_DATA – Specify any additional parameters to be passed to the TSX specified in TSX_NAME. TSX parameter data is optional.

Sending the Notification (Using the BLGTNMAN TSX)

The BLGTNMAN TSX is provided by Tivoli Information Management for z/OS so that you can communicate with the SMTP server to send the message. It contains the TCP/IP address for the server and can be called from a TSP, a TSX, through the RUN command, or from the API through the HL14 Start User TSP or TSX transaction.

From a TSX

A TSX can call the BLGTNMAN TSX through a LINK control line. For example:

```
Call BLGTSX 'LINK',BLGTNMAN,msgrnid
```

msgrnid is the message text record ID (required).

Using the RUN Command

Enter the RUN command and specify a message record ID (required):

```
RUN BLGTNMAN msgRNID
```

From the HLAPI

Use the HL14 Start User TSP or TSX transaction and pass the following input parameters:

- TSP_NAME – Specify BLGTNMAN as the TSX name. The value BLGTNMAN is required.
- RNID_SYMBOL – Specify the record ID of the message record you want to use.

Sending a Notification through the HLAPI

Portions of notification processing can be controlled through applications using the Tivoli Information Management for z/OS HLAPI. If you have applications that communicate with the Tivoli Information Management for z/OS database through the HLAPI, you can use HLAPI transaction HL14 to do the following:

- Send a notification message by starting the BLGTNMAN TSX.
- Queue messages on a remote data resource by starting the BLGTNPQM TSX.

More information about using the Tivoli Information Management for z/OS HLAPI is available in the *Tivoli Information Management for z/OS Application Program Interface Guide*.

Understanding Problem Escalation

You can use the problem escalation facility of Tivoli Information Management for z/OS to inform interested users of a problem's status and its priority until the problem is closed. Escalation processing runs in batch mode and at the intervals you set. The escalation of problem records is controlled by a list of criteria that you define in rules records, a record type defined specifically for problem escalation.

The escalation facility uses the rules records criteria to create a search argument that identifies problem records that you want escalated. For each problem that is identified, Tivoli Information Management for z/OS sends an alert message to one or more user IDs or mail addresses, as specified in the rules record. The **Current priority** field and the **Escalation level** field of the problem record are updated each time a message is sent.

Escalation messages can be processed using TCP/IP SMTP mail. A temporary file is created and then, after use, it is deleted.

Implementing Problem Escalation

Before you can implement escalation processing, you need to understand the process and perform the following tasks:

1. Read this section for an overview of the problem escalation process and the implementation tasks.

2. Establish your installation's escalation criteria within the problem management process. If you want to use escalation criteria that are different from those shipped by Tivoli, use PMF to update the rules record panels.
3. Define the rules records for each set of criteria that you want to monitor.
4. Create the JCL to run the escalation batch job and set the interval at which this job is to be run. Some sample JCL, called BLMESCAL, is provided for you in the SBLMSAMP data set.

Understanding the Escalation Process

The escalation facility uses RULES records to create a search argument to identify problem records to escalate. For each problem that is escalated, an alert message is sent to one or more user IDs specified in the RULES record. You can also use each RULES record user **ID** field as an "alias" that is defined along with one or more mail addresses in the USERS record. The alias can be a privilege class name or a name of your choice to identify one or more people that should be sent an escalation message. If you use this method, the escalation message is sent to each address defined for the alias. **Current Priority** and **Escalation Level** fields in the problem record are also updated.

The following steps outline the processing performed by TSP BLGESCAL. See "Notification Management User Exits" on page 499 for information about the escalation control block (ESCB) and the user exits for notification management.

1. Search for all RULES records and obtain a search results list. For complete information about creating RULES records, see "Creating Rules Records" on page 114.
2. Select each RULES record, one at a time from the search results list, and do the following:
   a. Move selected information from the RULES record to the ESCB.
   b. Move the search criteria from the RULES record to the ESCB.
   c. Store the escalation command template in the ESCB. The template contains the command and an outline of the format used to send the escalation alert message.
   d. Build a search argument to find all nonclosed problem records in the database. (A nonclosed problem is a problem with any status other than closed.)
   e. Determine which of the escalation criteria exist in the RULES record. Add them to the search argument. For example, if Key item affected is one of the search criteria and the RULES record contains an entry of REPORTS in the **Key item affected** field, then COMK/REPORTS is added to the search argument (COMK/ is the p-word for Key item affected). If the **Key item affected** field in the RULES record is blank, this p-word is not added to the search argument.
   f. Issue a SEARCH command to get a list of all nonclosed problem records that are candidates for escalation according to this RULES record.
   g. Process each problem record:
      1) Subtract the **Date** and **Time** fields in the problem record from the current date and time. The result is the calculated duration. The escalation level in the problem record determines whether the Date entered and **Time entered** fields or the **Date last altered** and **Time last altered** fields are used for the subtraction operation.

2) Compare the calculated duration against the duration in the RULES record to determine whether to issue an alert. The escalation level in the problem record determines whether the level 1, 2, or 3 duration should be used.

3) Build the alert message, using either the data in the variable data area for the Terminal Simulator Communications Area (TSCA) or a specified message panel.

4) Substitute the alert message and the appropriate information from the ESCB in the escalation command template.

5) Send the alert message.

6) If the **Priority adjustment** field in the RULES record is not 0, update the **Current priority** field in the problem record.

7) Update the Escalation level in the problem record.

8) File the record.

   Note: When problem records are updated because of escalation processing, immediate notification messages are *not* sent.

9) Processing is complete for one problem record. Repeat steps 2.g.1 through 2.g.8 until all problem records in the search results list are processed.

h. Processing is complete for one RULES record. Repeat steps 2.a through 2.g until all RULES records in the search results list created in step 1 have been processed.

3. Processing is complete. Print the messages on the chain and exit from Tivoli Information Management for z/OS.

The Escalation TSPs and TSXs

The BLGESCAL TSP controls the escalation facility. This TSP calls other TSPs and TSXs to perform different escalation functions. The following table names each TSP and its purpose. The processing steps in this table refer to “Understanding the Escalation Process” on page 112.

TSP Name	Purpose of the TSP	Processing Step
BLGESCAL	Escalation facility control.	1, 2.h, 3
BLGESC01	Finds the escalation criteria in a RULES record and stores the criteria in the ESCB.	2.a, 2.b, 2.d, 2.e, 2.f
BLGESC02	Controls the problem record loop area.	2.g.8, 2.g.9
BLGESC03	Gets data from the RULES record and stores it in the ESCB.	2.a
BLGESC04	(Continuation of BLGESC03.) Also stores the notification command template in the ESCB.	2.a, 2.c
BLGESC05	Determines the escalation level of a problem record.	N/A
BLGESC06	Performs processing for a problem record based on the escalation level.	2.g.1 through 2.g.7
BLGESC07	Builds a notification message.	2.g.3

In addition to these TSPs, a TSX is available for escalation processing:

TSX Name	Purpose of the TSX
BLGTNCEM	Sends the escalation message via TCP/IP SMTP

See “Using the BLGTNCEM TSX with TCP/IP SMTP” on page 122 for more information.

Creating Rules Records

Familiarize yourself with the escalation criteria defined in the escalation facility and decide whether they fit your installation’s needs. You must create your rules records before escalation processing can be done. The security of your rules records is controlled by the privilege class records in your database. The Privilege Class Record panels include authority for rules records.

The criteria used in the escalation facility as shipped from Tivoli uses the following fields in the rules records to find matching fields in problem records:

- **Initial priority**
- **Problem type**
- **Location code**
- **Key item affected**

Create a rules record for each set of unique search criteria that you want escalated. Refer to the *Tivoli Information Management for z/OS User’s Guide* for information about creating records.

To establish your site’s escalation criteria, make a list of the possible combinations of escalation criteria that are valid for your site. For example, perhaps your location has the following:

- Four initial priority values: 01, 02, 03, and 04
- Two problem types: Hardware and Software
- One location: A10
- Two key items affected: TAPE01 and DISK03B

Because every problem reported has a location code of A10, you need not use the **Location code** field as part of your search argument. Therefore, this field is not an escalation criterion for your installation.

Continuing with this example, you come up with a list of 16 rules records that include every possible situation that can occur. The following table lists these rules records.

Table 9. Example Rules Records

Rules Record Number	Initial Priority	Key Item Affected	Problem Type
1	01	TAPE01	Hardware
2	01	TAPE01	Software
3	01	DISK03B	Hardware
4	01	DISK03B	Software
5	02	TAPE01	Hardware
6	02	TAPE01	Software

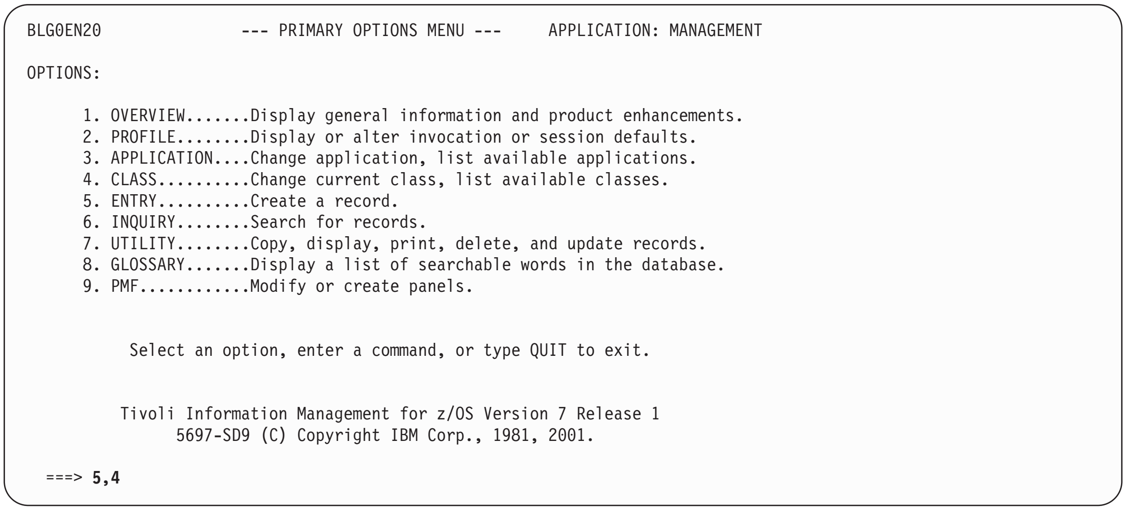
Table 9. Example Rules Records (continued)

Rules Record Number	Initial Priority	Key Item Affected	Problem Type
7	02	DISK03B	Hardware
8	02	DISK03B	Software
9	03	TAPE01	Hardware
10	03	TAPE01	Software
11	03	DISK03B	Hardware
12	03	DISK03B	Software
13	04	TAPE01	Hardware
14	04	TAPE01	Software
15	04	DISK03B	Hardware
16	04	DISK03B	Software

Scanning this list, you find that some combinations cannot logically occur in your installation. For example, you never open software problems with an initial priority of 01. Therefore, you do not need to create rules records numbered 2 and 4.

Suppose you have assigned only one person to monitor all problems with the key item TAPE01. You, therefore, need to create only one rules record for the key item affected TAPE01, leaving the **Initial priority** and **Problem type** fields blank. In this example, you do not need to create rules records numbered 1, 2, 5, 6, 9, 10, 13, and 14. The one rules record that you need to create uses the **Key item affected** field as the only escalation criterion.

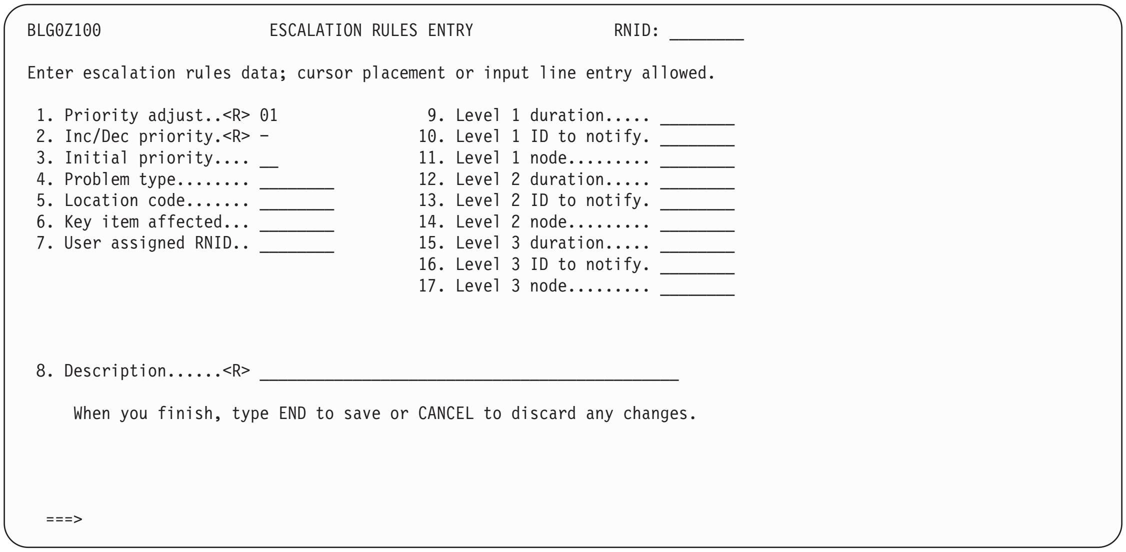
You can add, delete, or modify rules records at any time. A rules record that has no escalation criteria results in *all* nonclosed problem records being candidates for escalation. Rules records are Management application records, so be sure to start on Primary Options Menu, BLG0EN20, for the Management application. To get to the Escalation Rules Entry panel, type **5,4** and press **Enter**.



```
BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                5697-SD9 (C) Copyright IBM Corp., 1981, 2001.

  ===> 5,4
```

The Escalation Rules Entry panel contains several required fields. Some of these fields are primed. The **Current priority** field is primed with a value of 01, and the **Inc/Dec priority** field is primed with a minus sign (-). You can modify both fields.

When the criteria in this rules record cause a problem record to be escalated, the current priority in the problem record is adjusted based on the values in these fields. The values shown cause the **Priority adjust** field in the problem record to be decremented (because of the minus sign) by 1.



```
BLG0Z100                  ESCALATION RULES ENTRY               RNID: ________

Enter escalation rules data; cursor placement or input line entry allowed.

 1. Priority adjust..<R> 01              9. Level 1 duration..... ________
 2. Inc/Dec priority.<R> -              10. Level 1 ID to notify. ________
 3. Initial priority.... __             11. Level 1 node......... ________
 4. Problem type........ ________       12. Level 2 duration..... ________
 5. Location code....... ________       13. Level 2 ID to notify. ________
 6. Key item affected... ________       14. Level 2 node......... ________
 7. User assigned RNID.. ________       15. Level 3 duration..... ________
                                        16. Level 3 ID to notify. ________
                                        17. Level 3 node......... ________



 8. Description......<R> _____________________________________________

     When you finish, type END to save or CANCEL to discard any changes.




  ===>
```

The Level 1, 2, and 3 IDs can be the actual userids on the Level 1, 2, and 3 nodes or they can be aliases as defined in the USERS record. If they are aliases, do not use the node fields.

Understanding the Escalation Search Argument

The escalation facility initially searches for all rules records. It then builds a search argument for all nonclosed problems that match the criteria found in the first rules record escalation criteria fields. For example, if the **Initial priority** field in your first rules record has a value of 03, the search argument finds all nonclosed problems with an initial priority of 03. If the **Problem type** field also contains the entry SOFTWARE, the search argument finds all nonclosed problems with an initial priority of 03 and a problem type of software. This logic is continued for the **Location code** field. When the escalation function finishes processing the problem records that require escalation based on the first rules record, it continues to the next rules record and processes the appropriate problem records based on the next rules record criteria. This process continues until all of the rules records are used.

Table 10 illustrates how a search argument is built.

Note: The SUSPEND command is issued first so that when you are finished processing the problem records, you return to the next rules record in the list.

Table 10. Building the Search Argument

Initial Priority	Problem Type	Location Code	Key Item Affected	Search Argument
3				SUS,6,1,SE +¬STAC/CLOSED PRII/3
3	software			SUS,6,1,SE +¬STAC/CLOSED PRII/3 TYPE/SOFTWARE
3	software	A10		SUS,6,1,SE +¬STAC/CLOSED PRII/3 TYPE/SOFTWARE LOCC/A10
3	software	A10	DISK03B	SUS,6,1,SE +¬STAC/CLOSED PRII/3 TYPE/SOFTWARE LOCC/A10 COMK/DISK03B

All problem records have an **Escalation level** field that is not seen by the user. An escalation level of 1 is added to a problem record the first time the record is filed. The **Escalation level** field is incremented by 1 each time escalation occurs for this record (up to a maximum value of 4) to enable the escalation processor to determine which levels of escalation have been performed.

If you try to escalate a problem record that was created in your database on some version before 4.1 and the record has not been subsequently updated under 4.1 or a later version, the **Escalation level** field is added to the record the first time the record is updated.

The BLGESCAL TSP processes the rules records one at a time, locating all of the nonclosed problem records that match the criteria set in each rules record. Then BLGESCAL determines which records to escalate. The contents of the **Escalation level** field in each problem record determines which levels of duration, node, user ID, date set, and time set to use in the processing. If a record that meets the criteria does not contain an escalation level, an escalation level of 1 is assumed.

Next, the BLGESCAL TSP sends the escalation message, updates the **Current priority** field in the problem record according to the values in the rules record, and increments the number of the escalation level in the problem record. The maximum escalation level is 4. When this level is reached, it remains at 4 until the problem record is closed.

If the duration value in the rules record for the current escalation level is blank, the **Escalation level** field is incremented; however, the priority remains unchanged, and no message is sent.

Escalating Problems

Table 11 illustrates how two different rules records cause problem records to be escalated. These records illustrate how the escalation level and how long the problem has been active determine whether a problem is escalated and whether an alert message is sent.

Table 11. Values for Two Rules Records

Rules Record Field	Rules Record 1	Rules Record 2
Priority adjust	1	1
Inc/Dec Priority	-	+
Initial priority	20	01
Problem type	SWBATCH	
Location code		DALLAS
Key item affected	ANYITEM	ANYITEM
Level 1 duration	00:01:00	00:12:00
Level 1 ID to notify	USER01	USER01
Level 1 node	ABC01	DEF01
Level 2 duration	00:06:00	
Level 2 ID to notify	USER02	
Level 2 node	ABC01	
Level 3 duration	07:00:00	01:00:00
Level 3 ID to notify	USER03	USER03
Level 3 node	ABC02	DEF01

RULES Record 1

In this example:

- When a problem with a type of SWBATCH, an initial priority of 20, and a key item affected of ANYITEM has existed for at least one hour and has not been closed, USER01 at ABC01 is sent an alert message, and the **Current priority** field in the problem record is reduced by 1. The escalation level in the record is raised to 2.
- When the same problem has existed for at least 6 hours and has not been closed, USER01 and USER02 at ABC01 are sent a message, and the **Current priority** field in the problem record is reduced by 1. The escalation level in the record is raised to 3.
- When the same problem has existed for at least 7 days and has not been closed, USER01 and USER02 at ABC01 and USER03 at ABC02 are sent a message, and the **Current priority** field in the problem record is reduced by 1. The escalation level in the record is raised to 4.
- Alert messages are sent to all three users and the **Current priority** field is lowered by 1 every 7 days after that (until it reaches 1, where it stops) until the problem record is closed. If the problem record is updated during this time, alert messages are sent every 7 days from when the record was last updated and filed. The escalation level in the problem record, once it reaches 4, does not change.

RULES Record 2

In this example:

- When a problem with a location of DALLAS and an initial priority of 01 has existed for at least 12 hours and has not been closed, USER01 at DEF01 is sent an alert message, and the **Current priority** field in the problem record is raised by 1. The escalation level in the record is raised to 2.
- The next time the escalation batch job runs, Tivoli Information Management for z/OS looks at the **Level 2 duration** field in the rules record and finds a blank. This tells the system that no escalation message needs to be sent, and the current priority is not raised. The escalation level in the problem record is raised to 3.
- The system checks the **Level 3 duration** field on succeeding cycles, and, when the problem has existed for one day, a message is sent to USER03 at DEF01, as well as to USER01 at DEF01. The **Current priority** field in the record is increased by 1, and the escalation level is raised to 4.
- Assuming that no other updates are made to the problem record, alert messages are sent to both USER01 and USER03, and the **Current priority** field in the problem record is incremented by 1 every day after that until the problem record is closed. If the record is updated, alert messages are sent every day afterward, based on when the record was last updated and filed.

  Note: When a record is updated, the escalation clock restarts. For example, if the duration at escalation level 3 is 5 hours and the record is updated 4 hours after the last message was sent, the next message is sent when 9 hours have passed. If the record is not updated again, another message is sent in 5 hours.

  This example assumes you are not using the BLGESCX6 TSX to route escalation messages using TCP/IP SMTP.

Sending the Alert Message

Table 12 shows which users are notified at each escalation level.

Table 12. Users Notified at Each Escalation Level

Escalation Level	Duration Used	Date/Time Used	Messages Sent to
1	Level 1 duration	Date/time entered	Level 1 user ID or address
2	Level 2 duration	Date/time entered	Level 1 and 2 user IDs or addresses
3	Level 3 duration	Date/Time entered	Level 1, 2, and 3 user IDs or addresses
4	Level 3 duration	Date/Time last altered	Level 1, 2, and 3 user IDs or addresses

Note: If you want only one user notified in any rules record, specify that user as a level 3 user, leave levels 1 and 2 blank, and complete only the **Level 3 duration**, **Level 3 ID to notify**, and **Level 3 node** fields. You only need **Level 3 node** if your **Level 3 ID to notify** is a userid. If it is an alias, you do not need to specify anything in the **Level 3 node**. If the level 3 fields are blank, the escalation messages are sent only twice. After the level 1 and level 2 user IDs are notified, no other messages are sent.

Starting Escalation Processing

Problem escalation is designed to be started through a batch job submitted on a periodic basis to MVS. This job starts the BLGESCAL TSP that reads all the RULES records. If you have five different RULES records, the TSP performs five different searches against the nonclosed problem records in the database.

Creating the JCL for the Escalation Batch Job

BLMESCAL in the sample library (SBLMSAMP) contains JCL you can use to start the escalation batch job. Modify BLMESCAL as the prologue directs. The user ID you specify in the JCL must be the TSO logon ID dedicated to the Problem Escalation Facility. The user ID must be in the MASTER privilege class because that class has the authority to update all problem records, regardless of which class is the owning privilege class.

Be sure the JOB card matches the defined standards for your installation. You may need a JOBCAT card if the data sets are not in the master catalog. Also, you may not require a STEPLIB if the Tivoli Information Management for z/OS load modules are in the system link pack area.

Tivoli Information Management for z/OS creates a data set called &ZPREFIX; &ZPREFIX is the ISPF variable set when you enter PROFILE PREFIX(user ID). If a value cannot be found for &ZPREFIX, a data set called &ZUSER.$ESTEMP$ is created. Set the variable by using the TSO profile prefix command.

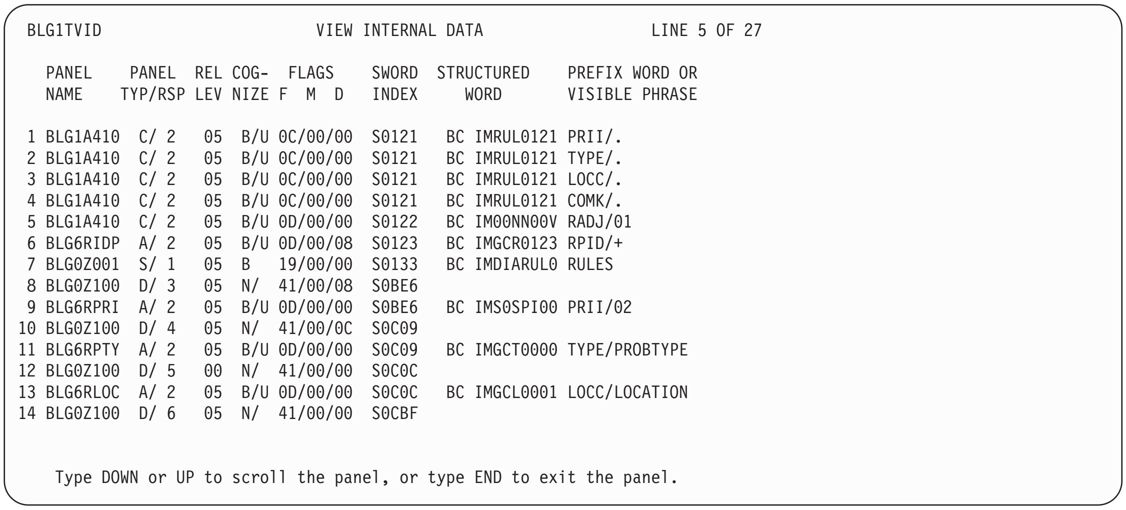
Tailoring the Escalation Facility

An s-word index of 0121 in a RULES record indicates that the prefix collected with it is used as escalation criteria. Escalation processing looks at the p-word associated with each 0121 s-word index and scans the rest of the RULES record for other occurrences of that prefix. If the prefix has data, that data is used in the search argument for escalation processing.

The following example illustrates how an escalation search argument is built.

Note: The line numbers in this example are for reference only.

This panel shows an internal view of a RULES record. The s-word index of 0121 for lines 1 through 4 shows the prefixes that are used as escalation criteria.



```
BLG1TVID                      VIEW INTERNAL DATA                  LINE 5 OF 27

  PANEL    PANEL  REL COG-  FLAGS    SWORD  STRUCTURED    PREFIX WORD OR
  NAME    TYP/RSP LEV NIZE F  M  D   INDEX     WORD       VISIBLE PHRASE

 1 BLG1A410  C/ 2   05  B/U 0C/00/00  S0121   BC IMRUL0121 PRII/.
 2 BLG1A410  C/ 2   05  B/U 0C/00/00  S0121   BC IMRUL0121 TYPE/.
 3 BLG1A410  C/ 2   05  B/U 0C/00/00  S0121   BC IMRUL0121 LOCC/.
 4 BLG1A410  C/ 2   05  B/U 0C/00/00  S0121   BC IMRUL0121 COMK/.
 5 BLG1A410  C/ 2   05  B/U 0D/00/00  S0122   BC IM00NN00V RADJ/01
 6 BLG6RIDP  A/ 2   05  B/U 0D/00/08  S0123   BC IMGCR0123 RPID/+
 7 BLG0Z001  S/ 1   05  B   19/00/00  S0133   BC IMDIARUL0 RULES
 8 BLG0Z100  D/ 3   05  N/  41/00/08  S0BE6
 9 BLG6RPRI  A/ 2   05  B/U 0D/00/00  S0BE6   BC IMS0SPI00 PRII/02
10 BLG0Z100  D/ 4   05  N/  41/00/0C  S0C09
11 BLG6RPTY  A/ 2   05  B/U 0D/00/00  S0C09   BC IMGCT0000 TYPE/PROBTYPE
12 BLG0Z100  D/ 5   00  N/  41/00/00  S0C0C
13 BLG6RLOC  A/ 2   05  B/U 0D/00/00  S0C0C   BC IMGCL0001 LOCC/LOCATION
14 BLG0Z100  D/ 6   05  N/  41/00/00  S0CBF


   Type DOWN or UP to scroll the panel, or type END to exit the panel.
```

In this example, the TSP first looks in the current record for an occurrence of the prefix PRII/. It finds PRII/02 (line 9) and adds it to the search argument. The TSP then adds TYPE/PROBTYPE (line 11) because of line 2, and it adds LOCC/LOCATION (line 13) because of line 3 to the search argument. Because of line 4, it looks for COMK/ next. No COMK/ p-word has an associated value, so this escalation criteria is not a part of the search argument in this RULES record.

Changing the Escalation Criteria

The escalation facility as shipped from Tivoli uses the following fields in the RULES records:

- Initial priority
- Problem type
- Location code
- Key item affected

To change the escalation criteria:

- Add the p-word (for example, PERH/) to the dictionary, if it is not already there.
- Update control panel BLG1A410 to include the ADD control lines with the s-word index 0121 and the required p-word index for the criteria you want to include in your search argument.
- Update the Escalation Rules panel (BLG0Z100) to reflect the same criteria. For example, if you add location code as an escalation criterion, add the **Location Code** field to panel BLG0Z100.

  You can also delete criteria in the same way. Although this task is optional if you delete criteria, we recommend that you make the modifications to prevent any confusion about which escalation criteria are used.
- Modify the Summary, Display, Inquiry, Utility, and Delete panels to reflect your changes.
- Modify the RFTs that print RULES records.

Changing the Escalation Message

The escalation notification message shipped with the Escalation Facility reads as follows:

ALERT: REC=*record id*, PRI=*current priority*, STAT=*current status*, DESC=*problem description*.

The BLGESC07 TSP builds this message. The BLGESNOT user exit assembles the string of data to be sent to the ISPEXEC module and substitutes the message into the position of the &DSN. variable in the escalation command template. You can modify the escalation message to meet your requirements either by changing TSP BLGESC07 or by building a message panel including the message you want to send.

You can specify a message panel and extract the message from it to create a simpler one, but you are limited to inserting only one piece of variable data into the message. If your message has inserted data, it is processed in the same manner as in the TSP MESSAGE function, and it requires the same input. Refer to the *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference* for more information about the MESSAGE function. Decide which variable you want to insert, such as the RNID, and do the following:

1. Create a message panel containing the new message.
2. Change TSP BLGESC06 by removing its LINK to TSP BLGESC07.
3. Update the USEREXIT line in BLGESC07 that calls BLGESNOT to include the message panel name in the **Panel name** field.

Using the BLGTNCEM TSX with TCP/IP SMTP

Tivoli Information Management for z/OS provides a TSX that can send escalation notification messages through TCP/IP SMTP.

The mail aliases are stored in RULES records that define the search criteria for problems to be escalated, and the people to alert at each escalation level. Specify the aliases in the **Level 1**, **2**, and **3 ID** fields and nothing in the node fields. The aliases represent the actual mail addresses which are in the USERS record. To use the escalation TSX, you must define a new message skeleton that includes a mail alias. TSX BLGTNCEM extracts mail addresses that are defined for the mail alias from the USERS record. You must define the mail aliases and addresses in the USERS record, or customize escalation processing to determine the mail address using an alternative method.

To use the BLGTNCEM TSX, you must first perform the following steps:

- Store a *mail alias* in the **Level 1**, **2**, and **3 ID** fields of the RULES record, instead of the userids and nodes. Then, define the mail aliases in the USERS record in the **ASSIGNEE NAME/ALIAS** field on BLGLUSER, the Mail Information Entry panel, and include their corresponding mail addresses.
- Ensure that the BLGTNMAN TSX contains your SMTP server information. See “Defining TCP/IP SMTP Server Information (Editing the BLGTNMAN TSX)” on page 106 for information on editing this TSX.
- Ensure that the BLGTNCEM TSX contains your escalation message source address information. See “Defining the Escalation Message Address (Editing the BLGTNCEM TSX)” on page 107 for information on editing this TSX.

Notification Management User Exits

The user exits described in this section are intended for use by the escalation facility. Eight of these user exits, BLGESFCB, BLGESGCB, BLGESGET, BLGESPUT, BLGESPUV, BLGNSYAL, BLGNSYFR, and BLGUSERS are also used by immediate notification. After you understand these exits and how they work together, you can use them for your own purposes.

Two possible input types are fields in the TSCA that are expected to be set before the user exit can run and data that is entered when the USEREXIT control line is created. In the first case, the input is described by using the TSCA field name. In the second case, the field name on the USEREXIT panel is specified. The exception to this is the variable data area, which does not have a TSCA name.

Note: User exits which require that information be passed in any field, other than the variable data area (VDA), cannot be run from a TSX. Refer to the Environment section of each of the user exits in this chapter to determine whether or not the user exit can be run from a TSX.

The Escalation Control Block

The escalation control block (ESCB) holds anchor information about a rules record while a batch escalation job is running.

Table 13 describes the contents of the control block; it also includes the field lengths and the offset of each field into the record. This information is provided for your use in diagnosing problems in the escalation facility. These fields must not be changed.

Table 13. Escalation Control Block Fields

Field Number	Description	Field Length (bytes)	Hex Offset
	Escalation control block	2BE	0
1	ESCB acronym	4	0
2	Rules record RNID	8	4
3	Reserved	2	C
4	Priority adjust amount	2	E
5	Adjust direction (+ or -)	1	10
6	Reserved	7	11
7	Level 1 duration	8	18
8	Level 1 ID to notify	8	20
9	Level 1 node	8	28
10	Level 2 duration	8	30
11	Level 2 ID to notify	8	38
12	Level 2 node	8	40
13	Level 3 duration	8	48
14	Level 3 ID to notify	8	50
15	Level 3 node	8	58
16	Form of command skeleton	50	60
17	The current date	8	B0

Table 13. Escalation Control Block Fields (continued)

Field Number	Description	Field Length (bytes)	Hex Offset
18	The current time	5	B8
19	Reserved	3	BD
	Everything that follows this field is used as escalation criteria and can be repeated a maximum of 30 times	1FE	C0
20	Length of prefix	1	C0
21	Actual prefix	10	C1

BLGESADD–Increment Counter

Adds 1 to the value in TSCAUFLD. This counter has leading zeros in its format.

Input TSCAUFLD as set by a SETFIELD control line

TSCAIPTR containing a pointer to a valid ESCB.

Output

TSCAUFLD is incremented by 1. Possible return and reason codes are listed in Table 51 on page 500.

Environment

TSP only.

Table 14. BLGESADD Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
4	4	A nonnumeric value was specified in TSCAUFLD.

The BLGESC01 TSP contains an example of this user exit.

BLGESCCL–Escalation Cleanup

Deletes the escalation load modules and frees the temporary data set.

Input TSCAIPTR containing a pointer to a valid ESCB.

Output

Escalation load modules are deleted, and the temporary data set is freed. Possible return and reason codes are listed in Table 52 on page 501.

Environment

TSP only.

Table 15. BLGESCCL Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
8	4	ESCB pointer is not valid.
8	8	Internal error was detected.

Table 15. BLGESCCL Return and Reason Codes (continued)

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
8	C	Error occurred freeing data set. Data set was not freed.

The BLGESCAL TSP contains an example of this user exit.

BLGESCKE–Check Escalation

Checks to see if an escalation job is in process.

Input None.

Output
Return and reason codes, as listed in Table 53 on page 501.

Environment
TSP only.

Table 16. BLGESCKE Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Escalation is in process.
4	4	Escalation is not in process.

The BLGNOTPU TSP contains an example of using this user exit.

BLGESCLR–Clear Control Block

Resets the ESCB fields to binary zeroes.

Input TSCAIPTR containing a pointer to a valid ESCB.

Output
ESCB is cleared. Possible return and reason codes are listed in Table 54 on page 502.

Environment
TSP only.

Table 17. BLGESCLR Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
8	4	ESCB pointer is not valid.

The BLGESC03 and BLGESC04 TSPs contain examples of using this user exit.

BLGESDAT–Date and Time

Puts the system date and time into the TSCA variable area.

Input The TSCA containing the contents of the **Get variable data** field in the Flag Field Specification panel of the USEREXIT control line.

YES The date and time is appended to the existing variable data.

NO The date and time overlay the existing variable data.

Output
TSCAVDAL containing the length of date and time data in the variable data area; variable data area containing the date in the YY/MM/DD format and the time in the HH:MM format. Return and reason codes are listed in Table 55 on page 502.

Environment
TSP only.

Table 18. BLGESDAT Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.

The BLGESC06 TSP contains an example of this user exit.

BLGESDUR–Duration

Locates the date and time fields and determines whether the duration (1, 2, or 3 depending on TSCAUFLD) specified by the current rules record is met for notifying a user ID. A problem record must be in update mode, and the escalation control block must contain the appropriate duration.

Input TSCAIPTR containing a pointer to a valid escalation control block.

TSCAUFLD containing the level of current escalation.

The **Structured word index** field in the Data Field Specification panel of the USEREXIT control line to use for comparison for escalation levels 1 to 3 (the **Date entered** field, as shipped for TSP BLGESC06).

The **New Structured word index** field in the Data Field Specification panel of the USEREXIT control line to use for comparing escalation levels (the **Time entered** field, as shipped for TSP BLGESC06).

Output
Return and reason codes as listed in Table 56 on page 503.

Environment
TSP only.

Table 19. BLGESDUR Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Time to escalate this problem record.
4	0	Not time to escalate this problem record.
4	4	Duration is blank in the rules record.
4	8	Date or time was not specified in the problem record.
8	4	ESCB pointer is not valid.
8	8	Internal error was detected.
8	C	Date or time conversion error.

Table 19. BLGESDUR Return and Reason Codes (continued)

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
8	10	Level number in TSCAUFLD is not valid.

The BLGESC06 TSP contains an example of this user exit.

BLGESFCB–Free Control Block

Frees the escalation control block. This must be the last user exit processed in the escalation function.

Input TSCAIPTR containing a pointer to a valid escalation control block.

Output
TSCAIPTR is set to 0. Possible return and reason codes are listed in Table 20.

Environment
TSP only.

Table 20. BLGESFCB Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
8	4	ESCB pointer is not valid.
8	8	Internal error was detected.

The following TSPs contain examples of using this user exit:
BLGESCAL
BLGNOTAE
BLGNOTAU
BLGNOTCE
BLGNOTCU

BLGESGCB–Get Control Block Storage

Gets storage for the ESCB and anchors it in the TSCA. This user exit must be called before any other escalation user exit is called.

Input None.

Output
TSCAIPTR containing a pointer to a valid ESCB. Possible return and reason codes are listed in Table 58 on page 504.

Environment
TSP only.

Table 21. BLGESGCB Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Escalation control block was set up successfully.
8	8	Internal error was detected.

The BLGESCAL TSP contains an example of this user exit.

BLGESGET–Get Control Block Field

Gets a field stored by BLGESPUT, BLGESPUV, or a user-written user exit from the ESCB and stores it in the TSCA variable data area.

Input TSCAIPTR containing a pointer to a valid ESCB.

The **Literal/Test data** field in the Data Field Specification panel of the USEREXIT control line calling this exit and containing the hex length of the data to be moved.

The **New data** field in the Data Field Specification panel of the USEREXIT control line calling this exit and containing the 1- to 7-byte hex offset within the escalation control block of the data to be moved.

The **Get variable data** field.

YES To append the data to the end of the variable data area.

NO To replace the data in the variable data area.

Output
TSCAVDAL containing the length of the data in the variable data area. The variable data area is updated with data from the ESCB. Possible return and reason codes are listed in Table 59 on page 504.

Environment
TSP only.

Table 22. BLGESGET Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
4	4	The length specified in the **Literal/Test data** field is not valid.
4	8	The offset specified in the **New data** field is not valid.
8	4	ESCB pointer is not valid.

The following TSPs contain examples of this user exit:
BLGESC01
BLGNOTAE
BLGNOTAU
BLGNOTCE
BLGNOTCU
BLGESC02

BLGESINI–Initialize

Loads the escalation user exits and allocates a temporary data set that you can use with the BLGESNOT user exit to send the escalation message.

Input TSCAIPTR containing a pointer to a valid ESCB.

Output
Environment for the escalation function.

Environment
TSP only.

Possible return and reason codes are listed in Table 23.

Table 23. BLGESINI Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
8	4	ESCB pointer is not valid.
8	8	Internal error was detected.
8	C	An error occurred while allocating data set.

The BLGESCAL TSP contains an example of this user exit.

BLGESLVL–Level Increment

Increases the escalation level by 1 in the current problem record ESCB, if possible.

Input TSCAIPTR containing a pointer to a valid ESCB.

Output
Escalation level incremented by 1 up to a maximum of 4. No processing occurs if the record has no escalation level or an escalation level of 4. Possible return and reason codes are listed in Table 24.

Environment
TSP only.

Table 24. BLGESLVL Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
4	4	No escalation level field was found in the problem record.
8	8	Internal error was detected.

The BLGESC06 TSP contains an example of this user exit.

BLGESNOT–Notify

Builds the command that notifies a user about a problem record. The command is built and stored in the TSCA variable data area, overlaying what was there. The message comes from one of the following:

- The message line in an input message panel
- Text in the input variable data area

To send a message from a message panel:

Input TSCAIPTR containing a pointer to a valid ESCB.

TSCAUFLD containing the level of current escalation set by a SETFIELD control line or the BLGESADD user exit.

The **Panel name** field in the Data Field Specification panel of the USEREXIT control line calling this user exit and containing the message panel name from which to extract a message.

TSCAVDAL containing the length of the send command.

Output
TSCAVDAL containing the length of the send command.

Variable data area containing the send command in a format that can be passed directly to ISPEXEC.

Environment
TSP only.

To send a message from a string of data:

Input TSCAIPTR containing a pointer to a valid ESCB.

TSCAUFLD containing the level of current escalation set by a SETFIELD control line in a TSP or the BLGESADD user exit.

TSCAVDAL containing the length of the message to be sent.

Variable data area containing the notification message if no message panel is used.

Note: The message must be created by any combination of a MOVEVAR control line, the BLGESGET user exit, and a user-written user-exit routine.

Output
TSCAVDAL containing the length of the send command.

The variable data area containing the send command in a format that can be passed directly to ISPEXEC.

Environment
TSP only.

Possible return and reason codes are listed in Table 62 on page 507.

Table 25. BLGESNOT Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Command was built successfully.
4	4	TSCAUFLD does not contain the current level of escalation.
8	4	ESCB pointer is not valid.
8	8	Internal error was detected.
8	C	Error occurred in processing data set.
8	10	Message text is supposed to come from the variable data area, but variable data length (TSCAVDAL) is zero.

Table 25. BLGESNOT Return and Reason Codes (continued)

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
8	14	Message text is supposed to come from a message panel, but the panel cannot be loaded.
8	18	Message text is supposed to come from a message panel, but the panel has no text.
8	1C	Panel specified for message text is not a message panel.
8	20	Command exceeds length of variable data area.

The BLGESC06 TSP contains an example of this user exit.

BLGESPRI–Priority Update

Increments the value associated with an input prefix by the priority adjust amount in the escalation control block. The record must be in update mode.

Input TSCAIPTR containing a pointer to a valid ESCB.

The **Literal/Test data** field in the Data Field Specification panel of the USEREXIT control line to be escalated (the **Current Priority** field as shipped for TSP BLGESC06).

Output
The specified field in the current record is incremented by the specified amount.

Environment
TSP only.

Possible return and reason codes are listed in Table 26.

Table 26. BLGESPRI Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Priority was adjusted successfully.
0	4	Priority adjustment amount equals 0.
0	8	Priority reached maximum (01 if the priority is being decremented, 99 if the priority is being incremented); priority is not adjusted.
4	4	Current priority field (the prefix specified in the **Literal/Test data** field) is not in the problem record; priority is not adjusted.
4	8	Rules record contains a priority adjustment amount that is not valid; priority is not adjusted.
4	C	The prefix in the **Literal/Test data** field is not valid, or the data in the problem record for that prefix is greater than two characters.
4	10	The data in the problem record associated with the prefix in the **Literal/Test data** field is nonnumeric.
8	4	ESCB pointer is not valid.

Table 26. BLGESPRI Return and Reason Codes (continued)

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
8	8	Internal error was detected.

The BLGESC06 TSP contains an example of this user exit.

BLGESPUT–Put TSCA Data in Control Block

Gets data from the TSCA and stores it in the escalation control block. This data can be retrieved by user-exit BLGESGET or a user-written exit routine.

Input TSCAIPTR containing a pointer to a valid ESCB.

TSCASDF containing data (collected by a FINDSDATA control line) to be moved into the ESCB.

The **New data** field in the Data Field Specification panel of the USEREXIT control line containing the 1- to 7-byte hex offset within the ESCB where the data is to be moved.

Output
ESCB updated with the data from the TSCA at the specified offset. Possible return and reason codes are listed in Table 64 on page 508.

Environment
TSP only.

Table 27. BLGESPUT Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful move.
4	4	The length specified in TSCASDFL is not valid.
4	8	The offset specified in the **New data** field is not valid.
8	4	ESCB pointer is not valid.

The following TSPs contain examples of this user exit:
BLGESC03
BLGNOTAE
BLGNOTAU
BLGNOTCE
BLGNOTCU

BLGESPUV–Put Variable Data

Gets data from the variable data area or literal data and stores it in the ESCB.

To move variable data:

Input TSCAIPTR containing a pointer to a valid ESCB.

Get variable data=YES in the Flag Field Specification panel of the USEREXIT control line.

TSCAVDAL containing the length of data to be moved; the TSCA variable data area containing the data to be moved.

The **New data** field in the Data Field Specification panel of the USEREXIT control line containing the 1- to 7-byte hex offset where the data is moved into the ESCB.

Note: **Get variable data=YES** and **Literal/Test data** are mutually exclusive. If both are specified, data is moved from the variable data area, and **Literal/Test data** is ignored.

Output
ESCB is updated with the contents of the variable data area.

Environment
TSP only.

To move literal data:

Input TSCAIPTR containing a pointer to a valid ESCB.

Literal/Test data field in the Data Field Specification panel of the USEREXIT control line calling this exit. This field contains the data to be moved into the ESCB.

New data field in the Data Field Specification panel of the USEREXIT control line containing the 1- to 7-byte hex offset where the data is moved into the ESCB.

Note: **Get variable data= YES** and **Literal/Test data** are mutually exclusive. If both are specified, data is moved from the variable data area, and **Literal/Test data** is ignored.

Output
ESCB is updated with the input value. Possible return and reason codes are listed in Table 65 on page 509.

Environment
TSP only.

Table 28. BLGESPUV Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
4	4	The length specified in TSCAVDAL or in the **Literal/Test data** field is not valid.
4	8	Offset specified in the **New data** field is not valid.
4	C	Neither variable nor literal data was specified for move (**Get variable data** = NO or blank, and the **Literal/Test data** field is blank).
8	4	ESCB pointer is not valid.

The BLGESC04 and BLGESC06 TSPs contain examples of this user exit.

BLGESSCT–Store Criteria

Scans the current rules record for an occurrence of escalation criteria (s-word index 0121) and stores its associated prefix in the next available spot in the ESCB.

Input TSCAIPTR containing a pointer to a valid ESCB.

TSCAUFLD containing a 1- to 7-byte index of the next available slot.

TSCARPD containing the prefix of the escalation criteria (collected from running a FINDSDATA control line) to be moved.

Output
The next available slot in the ESCB is filled. Possible return and reason codes are listed in Table 66 on page 510.

Environment
TSP only.

Table 29. BLGESSCT Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
4	4	The p-word contained in TSCARPD is not valid.
4	8	The p-word length contained in TSCARPDL is not valid.
4	C	Limit of criteria fields (30) has been reached.
8	4	ESCB pointer is not valid.

The BLGESC01 TSP contains an example of this user exit.

BLGESSEA–Get Escalation Criteria

Scans the current rules record for an occurrence of data for each escalation criterion, and adds any criteria found to the variable data area as part of a search argument. The search argument locates problem records that meet the escalation criteria in the rules record.

Input TSCAIPTR containing a pointer to a valid ESCB. The control block is valid if the return code is 0.

TSCAUFLD containing the number of criteria in the ESCB.

The TSCA variable data area containing `6,1,SE+` (the base argument for finding problem records).

Output
TSCAVDAL containing the length of data in the variable data area.

The TSCA variable data area containing the complete search argument for problem records that match the criteria in this rules record.

Possible return and reason codes are listed in Table 67 on page 511.

Environment
TSP only.

Table 30. BLGESSEA Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
4	4	TSCAUFLD did not contain the number of criteria in the ESCB.
4	8	Variable data area is full.
8	4	ESCB pointer is not valid.
8	8	Internal error was detected.

The BLGESC01 TSP contains an example of this user exit.

BLGUSERS–Extract Mail Address from USERS Record

Extracts the mail addresses from the USERS record that match an input name or mail alias.

Input Call BLGTSX ’USEREXIT’,’BLGUSERS’,*name,databaseid,*’SAVE’ or ’NOSAVE’

- *name* - The assignee name or mail alias that is defined in the USERS record. The name is required and must consist of 1 to 60 characters. If more than 60 characters are specified, the name is truncated to 60 characters.
- *databaseid* - An optional single number identifying the database from which to read the USERS record. If more than 1 character is specified only the first character is used. The default is the number 5.
- SAVE or NOSAVE - Saves the USERS record in storage, or does not save the record in storage. This parameter is required.

Output
Addresses are written to REXX stem BLG_ADDRESS. BLG_ADDRESS.0 contains the number of addresses returned. The maximum number of addresses that can be returned is 200.

Environment
TSX only.

Possible return and reason codes are listed in Table 70 on page 514.

Processing Notes:

You can specify that the USERS record be maintained in storage and not re-read on each call, unless another user updates it. If the user exit is called with the NOSAVE option the record is not maintained in storage and is re-read each time BLGUSERS is invoked. The NOSAVE option directs BLGUSERS to unconditionally read the USERS record and not maintain it in storage. By using BLGUSERS, your TSX does not have to explicitly display the USERS record in order to extract mail addresses from the list of names and addresses. BLGUSERS:

- Assumes the USERS record name/alias and address lists are sorted by name or alias. All entries with the same name must be together in the list.

- The name or alias length must be less than or equal to 60 characters.
- The address list in the USERS record must be defined as either string data or a maximum response of one word.
- Each name or alias in the USERS record must have a corresponding address, and vice versa.

If desired, TSXs can also access the USERS record data through normal TSP processing (by displaying the record and using FINDSDATA and/or GETLIST to search for and retrieve data).

Table 31. BLGUSERS Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Processing successful, at least one address is returned.
0	4	At least one name in the list did not have a corresponding address. At least one address is returned.
0	8	The number of addresses found exceeds the maximum allowed. The maximum number of addresses is returned.
4	4	The USERS record was not found. No addresses are returned.
4	8	The input name was not located in the USERS record or no address was defined for the first name occurrence. No addresses are returned.
8	4	Out of storage. No addresses are returned.
8	8	BLGUSERS was called from a TSP. Only a TSX can call it.
8	12	Logic error. The internal control blocks could not be located. No addresses are returned.
12	4	No parameters were specified or the assignee name or mail alias was not specified. Name and SAVE/NOSAVE are required. No addresses are returned.
12	8	The action parameter must be SAVE or NOSAVE. No addresses are returned.
12	12	First character of the name parameter was a blank. Remove leading blanks. No addresses are returned.
16	4	An error occurred while reading the USERS record. Other messages may be issued that indicate the specific error that occurred. No addresses are returned.
16	8	The specified or default database could not be accessed. No addresses are returned.

The BLGTNCEM TSX contains an example of this user exit.

6

Defining User Line Commands

Using PMF, you can define line commands that are unique to your installation and that can be used on various search results lists. You can also define commands that process when a user issues the RUN command. Both cases of user-defined commands are dependent upon the running of TSPs or TSXs. You can make your user's job easier by creating alternate names to identify these TSPs and TSXs. For example, you could create a TSP called TSPCLSRD or a TSX called TSXCLSRD to close a record. Most users probably find it easier to remember something simple, like "close", than to memorize 8-character TSP (or TSX) names. You can relate the simple name "close" to your TSP or TSX by using the Tivoli Information Management for z/OS ALIAS record.

This chapter tells you how to define a user line command for use on a search results list, create an ALIAS record, add an entry to the ALIAS record, and write a TSP or TSX to process a user line command. You must understand Terminal Simulator Panels (TSPs), Terminal Simulator EXECs (TSXs), and the Panel Modification Facility (PMF) to perform the TSP- and TSX-related tasks described in this chapter. For information about TSPs and TSXs, refer to the *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference* . For information about PMF, refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide*.

User line commands can also be defined for use on list processor panels. Refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide* for all information regarding list processor user line commands.

Defining User Line Commands for Search Results Lists

You can define unique line commands to use on search results lists. For example, you can define a line command to automatically close a problem record. Currently, a user must select the record for update from a search results list, change the **Status** field to **closed**, and then file the record. You can automate this procedure by defining a *user line command*.

When a user enters a valid user line command on a search results list, Tivoli Information Management for z/OS takes the data on the table display and puts it into a linked list with any other data collected by previous user line commands. User line commands are not immediately processed. The delay in running the commands allows for multiple records to be used for a single purpose. For example, the user can select a list of features to add to a component record, or the user can close a group of records.

Once the user line command data is collected, either the user or the installation can determine when to process it. Two commands, RUN and DROP, work with the user line commands. Refer to the *Tivoli Information Management for z/OS User's Guide* for complete information about these commands.

Defining User Line Commands and Where They Are Valid

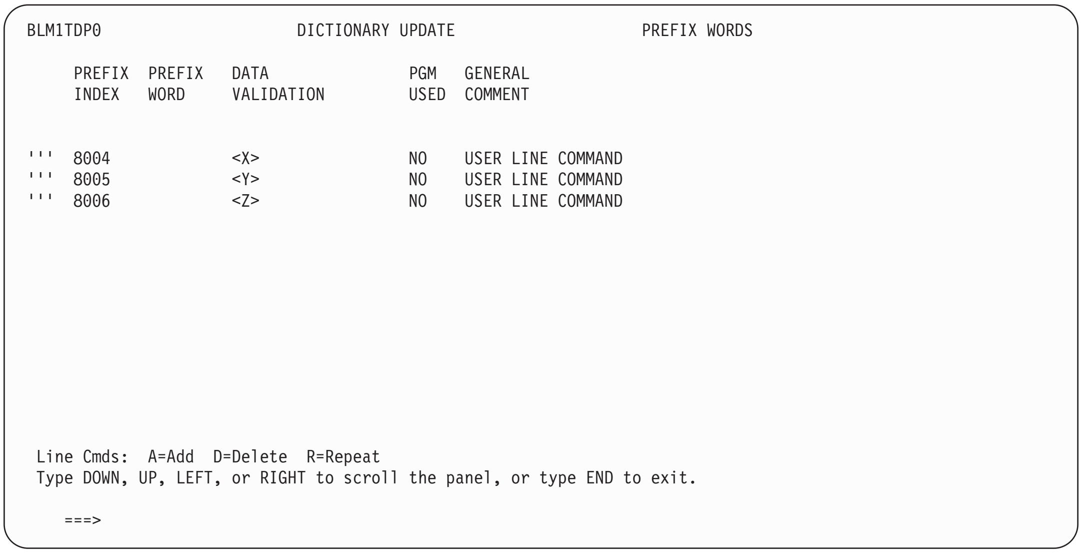
The first thing that you must determine is what line commands you want to define, and on which search results list panels you want to use the new line commands. You must also determine the names of any TSPs or TSXs that you want to run when a user enters one or more of these new commands. Once you have this information, define the new commands as follows:

1. Create a p-word entry in the dictionary for each new user line command. Do not specify a prefix, but do specify the data validation as the line command. For example, if you want to define three user line commands (X, Y, and Z), the data validation for each is as follows:

Command	**Data Validation**
X	<X>
Y	<Y>
Z	<Z>

Note: The line command must be entered in uppercase.

For this example, the Dictionary Display panel appears like this.



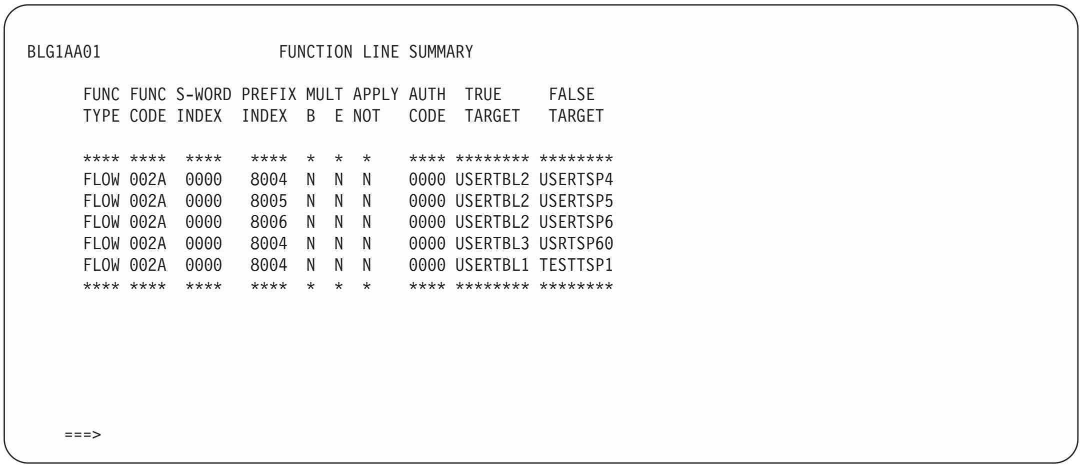
```
BLM1TDP0                  DICTIONARY UPDATE                 PREFIX WORDS

     PREFIX  PREFIX   DATA              PGM   GENERAL
     INDEX   WORD     VALIDATION        USED  COMMENT


'''  8004             <X>               NO    USER LINE COMMAND
'''  8005             <Y>               NO    USER LINE COMMAND
'''  8006             <Z>               NO    USER LINE COMMAND















 Line Cmds:  A=Add  D=Delete  R=Repeat
 Type DOWN, UP, LEFT, or RIGHT to scroll the panel, or type END to exit.

    ===>
```

2. Create a control panel called BLG1AA01. Add one line to this control panel for each line command and search results list combination. If you have two search results list panels that each allow three user line commands, you must create six control lines in the BLG1AA01 control panel. Use a FLOW line with a function code index of 002A, a true target of the search results list panel name, and a p-word index of one of the line commands for each combination.

   If you want to specify a TSP or a TSX to process this command when the user leaves the search results list, make the false target of this control line the name of the TSP or the TSX.

Note: To *leave* the search results list means that the user enters **END, CANCEL, QUIT, INITIALIZE,** or **RESUME**. A search results list is still in effect if the user can return to it at a later point in time. If a user does a **SUSPEND**, the search results list is still active because the user can later issue a **RESUME** to return to the original list. The user can also use the **DROP PANEL** command to drop pending line commands from a previous suspension level. This can only be done if the user has not issued user line commands in the current suspension level. If new line commands were entered in the current suspension level, only those user line commands are affected by the first **DROP PANEL** command. A second **DROP PANEL** can be issued to drop the line commands that exist on previous suspension levels instead of using the **DROP ALL** command. **DROP ALL** *always* deletes all data currently collected. When the user leaves a search results list, the BLGTENDS and BLGTENDL TSPs are run. If you do not need any processing when you leave a search results list, you can delete one or both TSPs from your base panel data set. BLGTENDL runs the DROP PANEL command to free the SLB and all SRB control blocks that were collected to run the user line commands for the current search results list table.

This is an example of the BLG1AA01 control panel.



```
BLG1AA01                    FUNCTION LINE SUMMARY

      FUNC FUNC S-WORD PREFIX MULT APPLY AUTH  TRUE     FALSE
      TYPE CODE INDEX  INDEX  B  E NOT   CODE  TARGET   TARGET

      **** ****  ****   ****  *  *  *    **** ******** ********
      FLOW 002A  0000   8004  N  N  N    0000 USERTBL2 USERTSP4
      FLOW 002A  0000   8005  N  N  N    0000 USERTBL2 USERTSP5
      FLOW 002A  0000   8006  N  N  N    0000 USERTBL2 USERTSP6
      FLOW 002A  0000   8004  N  N  N    0000 USERTBL3 USRTSP60
      FLOW 002A  0000   8004  N  N  N    0000 USERTBL1 TESTTSP1
      **** ****  ****   ****  *  *  *    **** ******** ********





    ===>
```

Note the following about the control panel:

- The prefix index listed in the control panel is the same as the prefix index in the dictionary entry for each user line command.
- The **True target** field identifies each search results list panel that the user line command is valid on.

If a user line command is valid on 3 different search results list panels, you need 3 FLOW control lines for each user line command and search results list panel combination. Tivoli Information Management for z/OS uses the information specified in BLG1AA01 to insure that only user line commands associated with a particular search results list panel are valid on that panel. An optional TSP or TSX, listed in the **False target** field, can also be associated with each user line command and search results list panel combination.

In this example, if a user issues user line command Y on a search results list, the second FLOW control line (for prefix index 8005) is referenced. If the user issues the line command from the panel listed in the **True target** field (USERTBL2), TSP USERTSP5 runs when the user leaves the search results list or issues the RUN command. If the user does not issue the line command from the panel listed in the **True target** field, Tivoli Information Management for z/OS scans the rest of the control panel for a match. If there is no match (either the search results list panel was not associated with the user line command or the user line command itself was not listed in the control panel), an error message is issued. If a match is found, but there is no TSP or TSX specified, the BLGTDFLT TSP runs.

3. For additional control lines that cannot fit on this one control panel, you can continue by creating control panel BLG1AA02, then BLG1AA03, and so on up to control panel BLG1AA09. If you skip a panel in the numerical sequence, any panels that follow the omitted number are ignored.

Using TSP and TSX Aliases

When the user line command is processed can be determined by either the end user or the installation, but *how* the data is processed must be determined by the program administrator.

If your end users prefer to have user line commands process before the search results list ends, they can use the RUN command to start the line commands that have collected so far. Rather than require all your end users to remember a TSP name or a TSX name, you might want to give a more meaningful name to the data to be processed. For example, if you have a user line command to close a record, the user can remember that the command to process that record is RUN CLOSE, rather than RUN TSPCLSRD or RUN TSXCLSRD. This simple name for your TSP or TSX is called an *alias*. You can have more than one alias for each TSP or TSX.

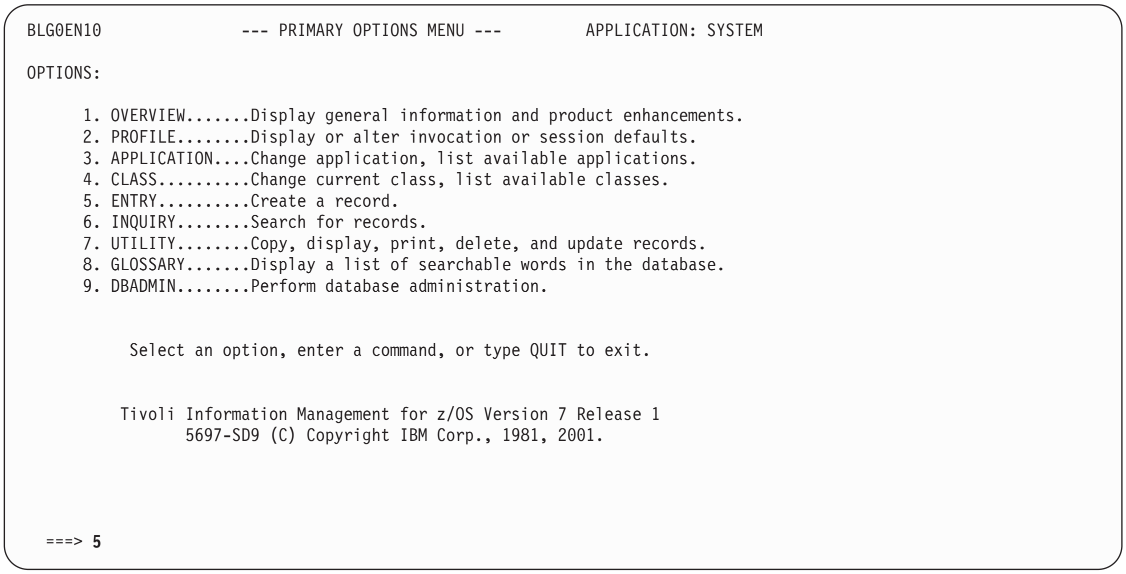
If you want to use an alias, you must add an entry to the ALIAS record. This record correlates the name of your alias to its associated TSP or TSX. Also, you can have an entry in the ALIAS record for a TSP or TSX that has no correlation to user line commands. This type of alias can also be called by the RUN command (for example, RUN CLOSE).

For performance reasons, you should consider creating an alias for each TSX even if the alias name is the same as the TSX name. Refer to the *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference* for more information on invoking TSXs.

The next section tells you how to create the ALIAS record and add entries to it.

Creating the ALIAS Record

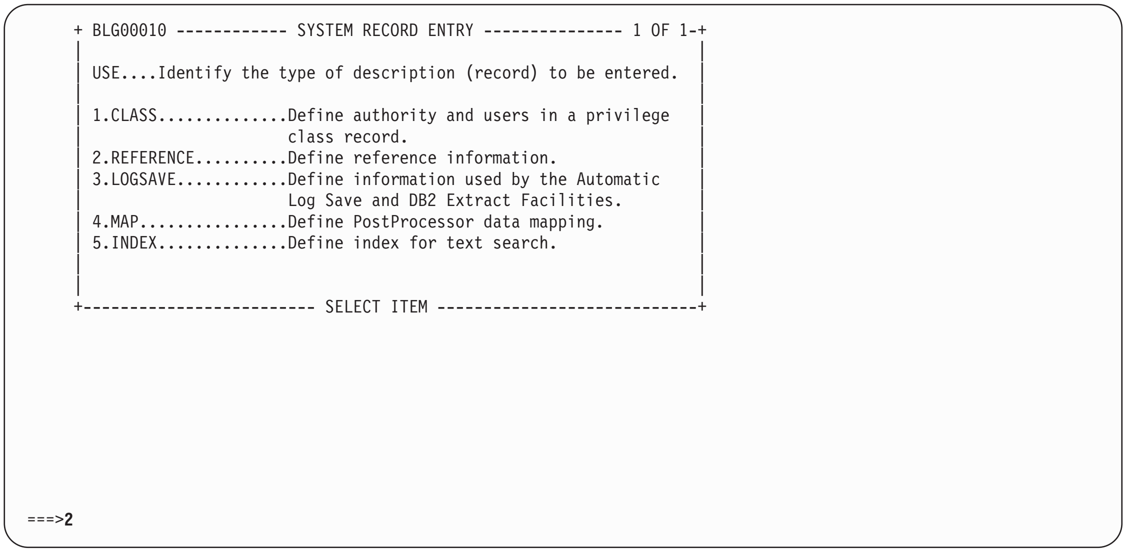
To create the ALIAS record, select **Entry** from the Primary Options Menu for the System application. Type **5** and press **Enter**.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.




  ===> 5
```

On BLG00010, the System Record Entry panel, select the kind of record you want to create.

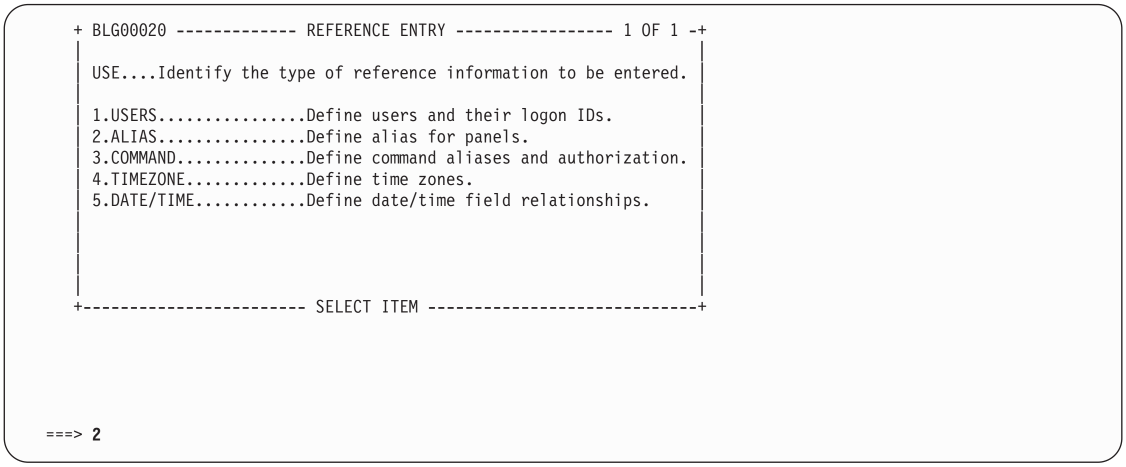
For this exercise, select **Reference**. Type **2** and press **Enter**.



```
       + BLG00010 ------------ SYSTEM RECORD ENTRY --------------- 1 OF 1-+
       |                                                                   |
       | USE....Identify the type of description (record) to be entered.   |
       |                                                                   |
       | 1.CLASS.............Define authority and users in a privilege     |
       |                     class record.                                 |
       | 2.REFERENCE.........Define reference information.                 |
       | 3.LOGSAVE...........Define information used by the Automatic      |
       |                     Log Save and DB2 Extract Facilities.          |
       | 4.MAP...............Define PostProcessor data mapping.            |
       | 5.INDEX.............Define index for text search.                 |
       |                                                                   |
       |                                                                   |
       +------------------------ SELECT ITEM ----------------------------+







===>2
```

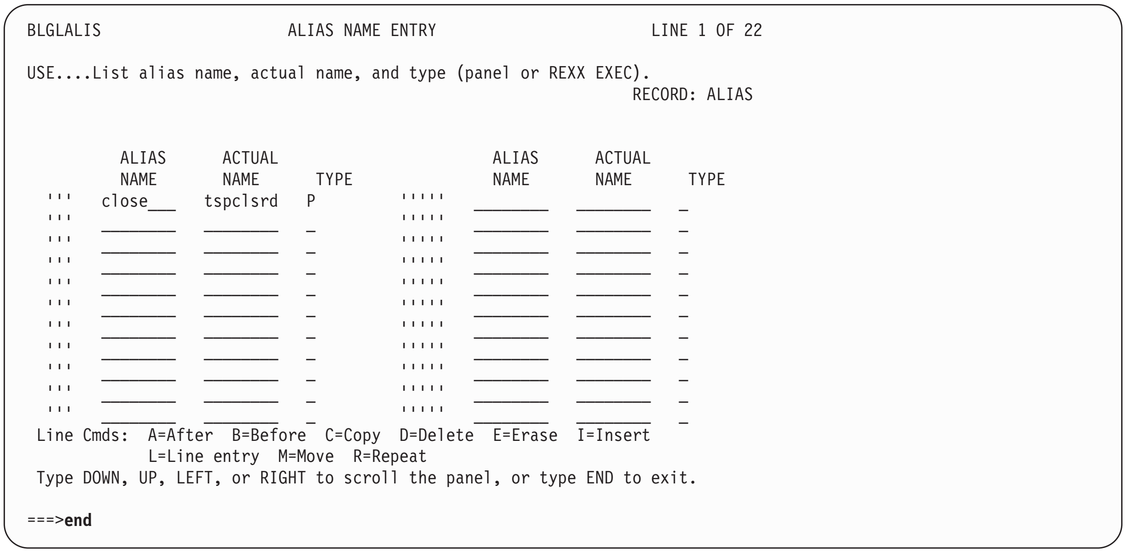
BLG00020, the Reference Entry panel, offers you a choice of creating different types of reference records. Choose **Alias**. Type **2** and press **Enter**.

If you already have the ALIAS record in your system, this selection brings your existing record into update mode so you can add or change data.



```
+ BLG00020 ------------- REFERENCE ENTRY ----------------- 1 OF 1 -+
|                                                                  |
| USE....Identify the type of reference information to be entered. |
|                                                                  |
| 1.USERS................Define users and their logon IDs.         |
| 2.ALIAS................Define alias for panels.                  |
| 3.COMMAND..............Define command aliases and authorization. |
| 4.TIMEZONE.............Define time zones.                        |
| 5.DATE/TIME............Define date/time field relationships.     |
|                                                                  |
|                                                                  |
|                                                                  |
|                                                                  |
+------------------------ SELECT ITEM -----------------------------+




===> 2
```

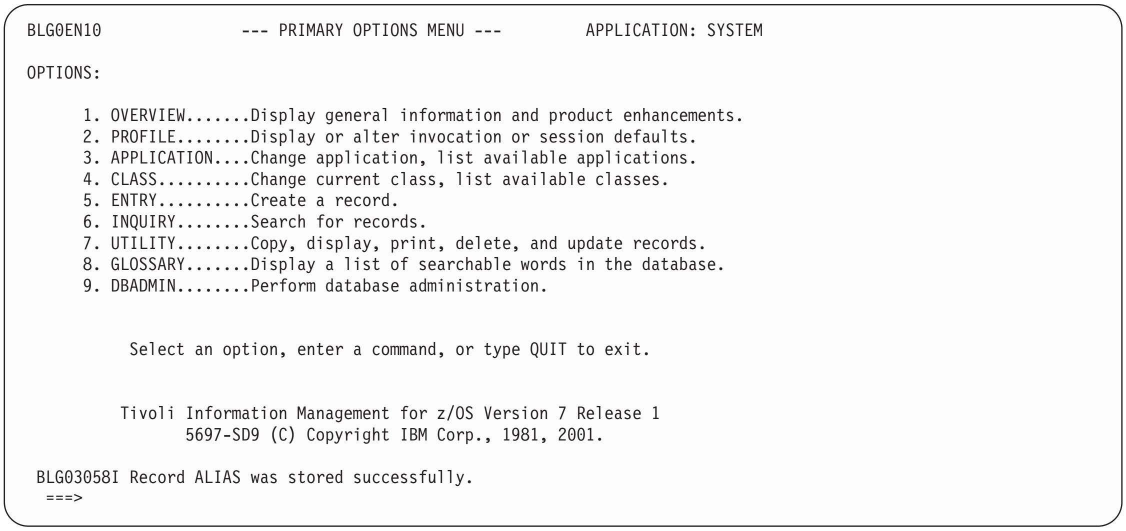
If you are creating the ALIAS record for the first time, an empty Alias Name Entry panel appears. On this panel, list the alias names for your TSPs and TSXs. For example, to identify CLOSE as an alias for TSPCLSRD or TSXCLSRD, enter these values into the record. In the **TYPE** column, enter a **P** if this is for a **TSP** or an **X** if this is for a **TSX**. After you enter the data, type **end** and press **Enter** to save your entries.



```
BLGLALIS                    ALIAS NAME ENTRY                      LINE 1 OF 22

USE....List alias name, actual name, and type (panel or REXX EXEC).
                                                                RECORD: ALIAS


        ALIAS      ACTUAL                            ALIAS      ACTUAL
        NAME       NAME      TYPE                    NAME       NAME      TYPE
  '''   close___   tspclsrd   P        '''''   ________   ________   _
  '''   ________   ________   _        '''''   ________   ________   _
  '''   ________   ________   _        '''''   ________   ________   _
  '''   ________   ________   _        '''''   ________   ________   _
  '''   ________   ________   _        '''''   ________   ________   _
  '''   ________   ________   _        '''''   ________   ________   _
  '''   ________   ________   _        '''''   ________   ________   _
  '''   ________   ________   _        '''''   ________   ________   _
  '''   ________   ________   _        '''''   ________   ________   _
  '''   ________   ________   _        '''''   ________   ________   _
  '''   ________   ________   _        '''''   ________   ________   _
 Line Cmds:  A=After  B=Before  C=Copy  D=Delete  E=Erase  I=Insert
             L=Line entry  M=Move  R=Repeat
 Type DOWN, UP, LEFT, or RIGHT to scroll the panel, or type END to exit.

===>end
```

A valid alias name is a string of 1 to 8 SBCS alphanumeric characters. You can enter as many alias names as you need. If you run out of lines on the screen, you can use the **R** or **I** line commands to add lines to the record.

BLG0EN10, the Primary Options Menu for the System application, appears with a message that the ALIAS record was stored successfully.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---         APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.

 BLG03058I Record ALIAS was stored successfully.
  ===>
```

Processing User Line Commands

After you complete the setup identified in the previous sections of this chapter, you must write the TSPs or TSXs and user exits to process the user line commands that users enter. If you define many user line commands, but only write TSPs or TSXs that account for a subset of those commands, then you can only use that subset, or your TSP or TSX must be general enough to process all the valid line commands it receives. Each user line command must be processed by a TSP or by a TSX.

To write the TSP or TSX and user exit, you must know the data that the user line commands collect. Three control blocks are involved in the data collection:

- **Search User Block (SUB)**. An SUB is created the first time the user issues a user line command on a search results list. It is used to find your way through the second type of control block, the Search List Buffer.

  The TSCASUBP field in the Terminal Simulator Communications Area (TSCA) points to the SUB control block.

  ```
  SUB      DSECT             /*Search user buffer
           DS   0F
  SUBACRON DS   CL4          /*'SUB'
  SUBFSLB  DS   A            /*Pointer to the first SLB
  SUBLSLB  DS   A            /*Pointer to the last SLB
  SUBIBMF  DS   XL1          /*Flag field for Tivoli use
  SUBUSRF  DS   XL1          /*Flag field for user-defined data
  SUBSHRF  DS   XL2          /*Flag field for Tivoli and user data
  ```

- **Search List Buffer (SLB)**. Each time the user issues the SEARCH command, there is one occurrence of the SLB where user line commands were issued. The SLB gives you information about the search results list panel on which the user line commands were issued, and points you to the third type of control block, the Search Row Buffer.

```
SLB      DSECT             /*Search list buffer
         DS   0F
SLBACRON DS   CL4          /*'SLB'
SLBFWDP  DS   A            /*Pointer to the next SLB
SLBPRVP  DS   A            /*Pointer to the previous SLB
SLBFSRB  DS   A            /*Pointer to the first SRB
SLBLSRB  DS   A            /*Pointer to the last SRB
SLBTABL  DS   CL8          /*Name of table panel collecting line commands
SLBTUBP  DA   A            /*Point to TUB
SLBIBMF  DS   XL1          /*Flag field for Tivoli use
SLBUSRF  DS   XL1          /*Flag field for user-defined data
SLBSHRF  DS   XL2          /*Flag field for shared data
SLBACTIV EQU  X'80'        /*This SLB is in use
         DS   A            /*Tivoli reserved area
SLBSNUM  DS   XL2          /*Number of s-word for table panel
SLBSDATA DS   0XL12        /*S-Word data
SLBITEM  DSECT             /*One s-word entry
SLBSLEN  DS   XL2          /*Length of s-word
SLBSWORD DS   CL10         /*Actual s-word
```

- **Search Row Buffer (SRB)**. An SRB is created for each row on which a user line command is entered. Each time a user enters a user line command on a search results list, Tivoli Information Management for z/OS takes all the data that appears on the table display and puts it into an SRB. It then links that SRB to any others that were created by previously entered user line commands on this same table panel. The data in the SRB includes the entire length of the data specified as the maximum for each field in the table panel up to a maximum of 64 characters. This means you get all the data, not just that which appears on the table display.

  The SRB also contains the name of the user line command. This is so you can write a user exit that can determine how to use the data based on the line command that the user entered.

```
    SRB       DSECT             /*Search row buffer
              DS   0F
    SRBACRON  DS   CL4          /*'SRB'
    SRBFWDP   DS   A            /*Pointer to the next SRB
    SRBPRVP   DS   A            /*Pointer to the previous SRB
    SRBLCOL   DS   A            /*Pointer to the last column of data
    SRBRNID   DS   CL8          /*Record ID of selected row
    SRBCPRC   DS   A            /*Pointer to last row processed
    SRBLCMD   DS   CL1          /*Line command to be processed
              DS   CL3          /*Reserved field
    SRBTSPIN  DS   F            /*Index of TSP name in TUB
    SRBLNUM   DS   F            /*Line number on search results list
    SRBIBMF   DS   XL1          /*Flag field for Tivoli use
    SRBUSRF   DS   XL1          /*Flag field for user data
    SRBSHRF   DS   XL2          /*Flag field for shared data
    SRBRECF   EQU  X'80'        /*Indicates record has been modified
    SRBSDATA  DS   0CL68        /*Column data for this row
/*******************************************************************/
/*  There is one SRBITEM per column of data on a table panel.  The   */
/*  first column is the line command column, followed by the next    */
/*  column of data you defined.  For example, on panel BLG1TSRL,     */
/*  the first SRBITEM is the line command column.  The second        */
/*  SRBITEM is the Record ID column, and the last SRBITEM is the     */
/*  Description Abstract column.                                     */
/*******************************************************************/
    SRBITEM   DSECT             /*A column of data
      SRBDLEN DS   XL2          /* Used length of this column
      SRBFLEN DS   XL2          /* Entire length of this column
      SRBDATA DS   CL64         /* Actual data for this column
```

Note: Neither SRBDLEN nor SRBFLEN will ever exceed 64 characters, even if the table panel field length is defined as greater than 64 characters.

The assembler mappings for these control blocks are shipped with Tivoli Information Management for z/OS in the SBLMMACS data set and are named:

Search User Block
BLGUSUB

Search List Buffer
BLGUSLB

Search Row Buffer
BLGUSRB.

Example User Exit

The sample library (SBLMSAMP) contains an example of an assembler user exit that returns the RNID of a record. This user exit is called BLGSRNID. This example contains a conversion from hexadecimal to printable format. You can use this example in conjunction with the TSCA variable data area.

TSP and TSX Considerations

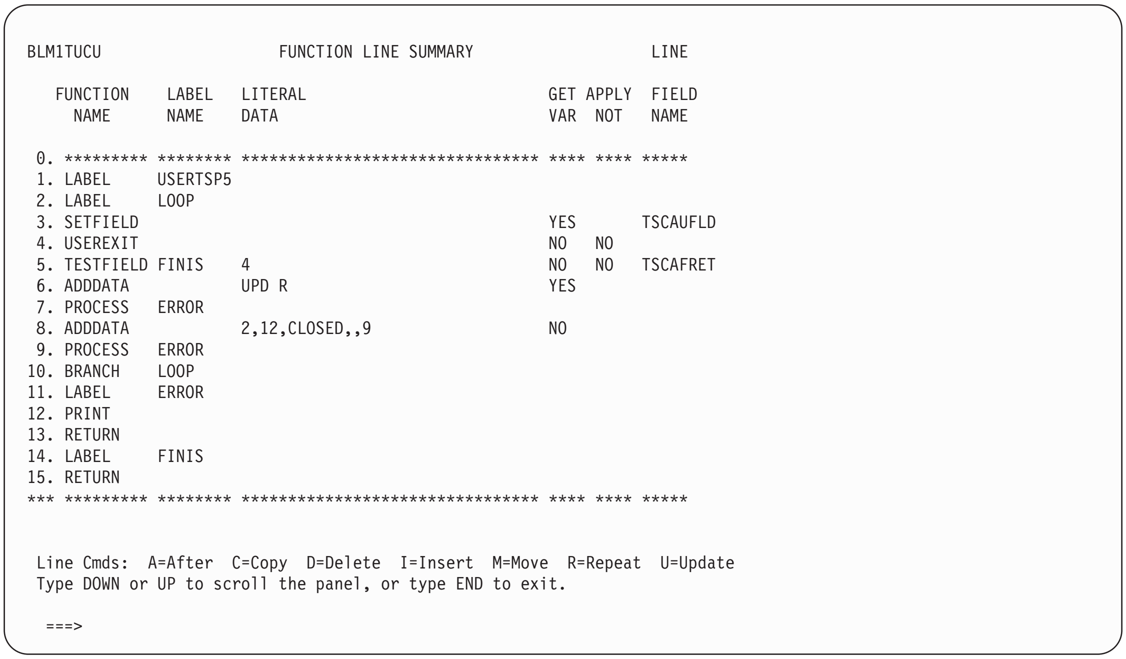
You have access to the SUB whenever you are running a TSP or TSX because the TSCA points to it. The TSP or TSX you write to handle the user line commands must be coded so they can be called in any of these ways:

- The user can issue the RUN command. If you allow your users to select multiple items from the search results list, you must remember that they might issue the RUN command after each one of them. You do not want your TSP or TSX to remove the user from the search results list in that instance.
- The collected data remains on the system until a DROP command is issued. You can have a TSP or TSX process the data later in your session, or your TSP or TSX itself can issue the DROP command if you no longer need the data.
- When the user leaves the search results list, a TSP or TSX runs. This TSP or TSX is the one identified by the control panel containing the appropriate table panel and user line command combination, if specified.

  Note: If you do not assign a TSP or TSX for this particular combination, a default TSP (BLGTDFLT) runs. The BLGTDFLT that Tivoli ships contains only a RETURN. You must modify BLGTDFLT to work the way you want it to.
- A TSP runs before all other line command TSPs run. This TSP (BLGTENDS) that Tivoli ships contains only a RETURN. You must modify it if you need special processing after the other line commands are finished. For example, if a previous TSP issued a SUSPEND command, then performed the actual search and issued line commands, you could have TSP BLGTENDS issue a RESUME command.
- A TSP runs after all other line command TSPs are finished. This TSP, BLGTENDL, issues a DROP PANEL command.

  Note: Tivoli-defined line commands take precedence over user-defined line commands. For example, if you create a U line command for a panel where U is already a valid line command, your user-defined value is not recognized.

This example TSP closes a problem record. This TSP can be used with the user exit described in “Example User Exit” on page 145.



```
BLM1TUCU                  FUNCTION LINE SUMMARY                   LINE

   FUNCTION    LABEL   LITERAL                          GET APPLY  FIELD
     NAME      NAME    DATA                             VAR  NOT   NAME

 0. ********* ******** ******************************** **** **** *****
 1. LABEL     USERTSP5
 2. LABEL     LOOP
 3. SETFIELD                                            YES       TSCAUFLD
 4. USEREXIT                                            NO   NO
 5. TESTFIELD FINIS    4                                NO   NO   TSCAFRET
 6. ADDDATA            UPD R                            YES
 7. PROCESS   ERROR
 8. ADDDATA            2,12,CLOSED,,9                   NO
 9. PROCESS   ERROR
10. BRANCH    LOOP
11. LABEL     ERROR
12. PRINT
13. RETURN
14. LABEL     FINIS
15. RETURN
*** ********* ******** ******************************** **** **** *****


 Line Cmds:  A=After  C=Copy  D=Delete  I=Insert  M=Move  R=Repeat  U=Update
 Type DOWN or UP to scroll the panel, or type END to exit.

  ===>
```

To further explain the code lines of the TSP in the previous panel:

Line Action

2 Starts processing and provides the label to loop back to.

3 Sets the **TSCAUFLD** field to **Y**. The GETSRNID user exit, which is run in the next step, uses this value to determine whether to use or skip the SRB created for a particular line command. In this case, it only uses an SRB if it was created because the user specified the **Y** user line command.

4 Starts the GETSRNID user exit. (This is the example user exit that is in the sample library and referenced in “Example User Exit” on page 145.) It scans through the SRBs for the first one associated with the Y user line command that has not yet been processed and stores the corresponding record ID in the TSCA variable data area.

5 This line checks the return code set in the user exit. If the return code is 4, the TSP branches to label FINIS and processes a return indicating that all processing is finished.

6 This line issues an UPDATE command. It also performs a GET VARIABLE DATA operation to add the RNID of the record (that the line command was issued for on the search results list) to the command. The RNID is put into the variable data area of the TSCA by the user exit.

7 Processes line 6.

8 and 9
Closes the record and files it.

10 This line branches back to label LOOP and starts the process over again if more line commands need to be processed.

7

Defining Command Aliases and Authorization

If your end users have a need to enter commands that are meaningful to them (for instance, STARTOVER may be easier to remember than INITIALIZE), you can set up a *command alias* to customize Tivoli Information Management for z/OS commands and improve user productivity. A command alias is set up by the program administrator through a reference record called the COMMAND record. A command alias can be defined for Tivoli Information Management for z/OS commands or for immediate response chains (IRCs) that begin with a Tivoli Information Management for z/OS command (for example, %PDFEDIT could be set up as an alias for PROFILE,1,51,PDF,,9). Command aliases are invoked the same way as standard Tivoli Information Management for z/OS commands. Anywhere a Tivoli Information Management for z/OS command can be invoked, a command alias can be invoked.

Additionally, you can define an authorization code for each command alias to restrict its usage by privilege class. Command alias and authorization can also be used to set up authorization for the Tivoli Information Management for z/OS commands. To set up authorization for a Tivoli Information Management for z/OS command, define a command alias identical to the Tivoli Information Management for z/OS command and set an authorization code for it.

You can also optionally specify a help panel to be shown when the user enters HELP for the command alias. If you do not specify a help panel, Tivoli Information Management for z/OS displays the help panel for the first command in the actual command string that the command alias represents.

Command alias names can be up to 32 characters long. The actual command string represented by the command alias can be up to 80 characters long. Command alias names must begin with A - Z, $ (X'5B'), # (X'7B'), @ (X'7C'), or % (X'6C'). Other characters in the name can consist of A-Z, 0-9, $, #, or @. The actual command string must begin with a standard Tivoli Information Management for z/OS command.

Future commands or aliases may be added to the Tivoli Information Management for z/OS product which may be identical to a command alias you have defined for your users. To avoid this conflict, start your command alias names with a **$, #, @, or % character.** Tivoli Information Management for z/OS will not use these characters as the first character of a command.

If you define a command alias to match a Tivoli Information Management for z/OS command exactly, the command alias will be processed in all cases *except* when it is the *first* response in the command string for another command alias. The Tivoli Information

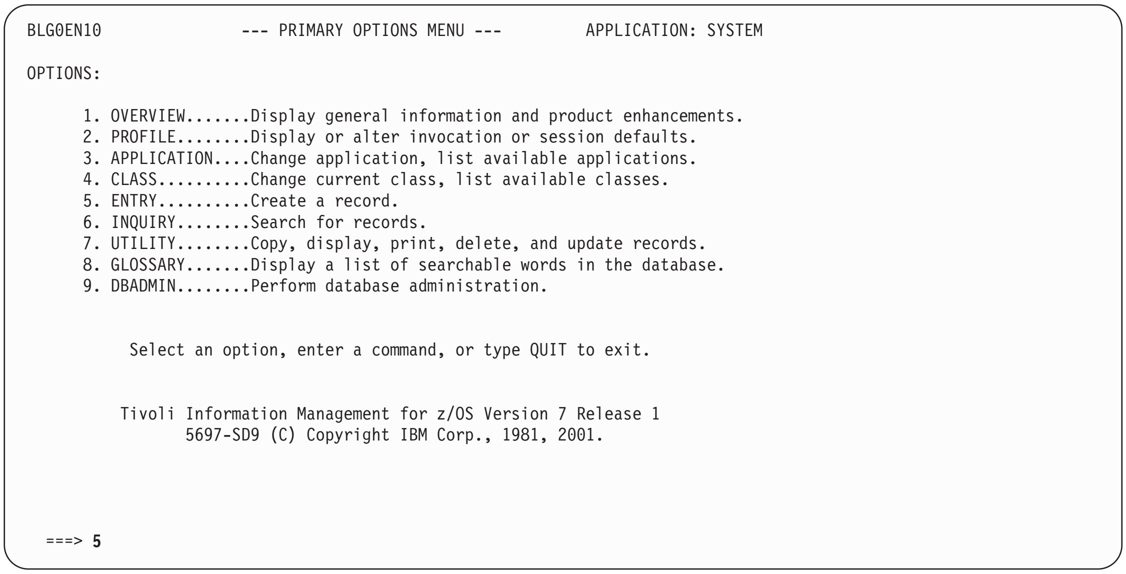
Management for z/OS command will always be used as the first response in the actual command string. For example, if a command alias of SHOW is defined with an actual command string of DISPLAY R, the Tivoli Information Management for z/OS DISPLAY command is used even if there is a command alias with a name of DISPLAY defined. This exception enables you to define command aliases which redefine existing Tivoli Information Management for z/OS commands to include additional arguments, or simply to authorize the command aliases. You could, for instance, define a command alias named VIEW for the Tivoli Information Management for z/OS VIEW command, and set it up with a valid authorization code to restrict use of the VIEW command in your user community.

To invoke a command alias, users simply enter the command alias name as they would any other command. Tivoli Information Management for z/OS processes the actual command string for that command alias or displays a message if the user is not authorized.

The next section shows you how to create a COMMAND record, and set up authorization codes for command aliases.

Creating the COMMAND Record

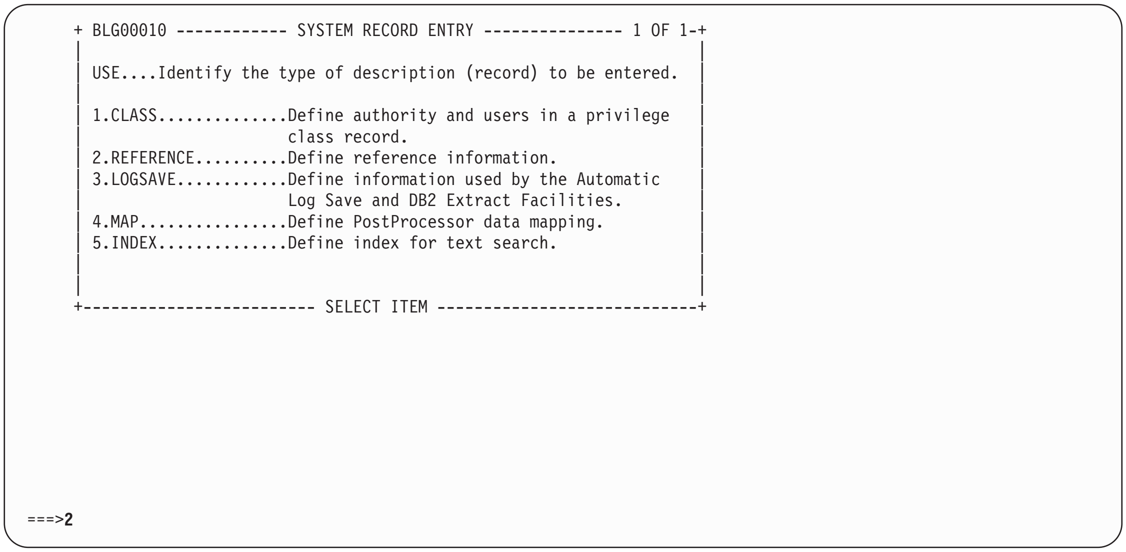
To create the COMMAND record, select **Entry** from the Primary Options Menu for the System application. Type **5** and press **Enter**.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.




  ===> 5
```

On BLG00010, the System Record Entry panel, select the kind of record you want to create.

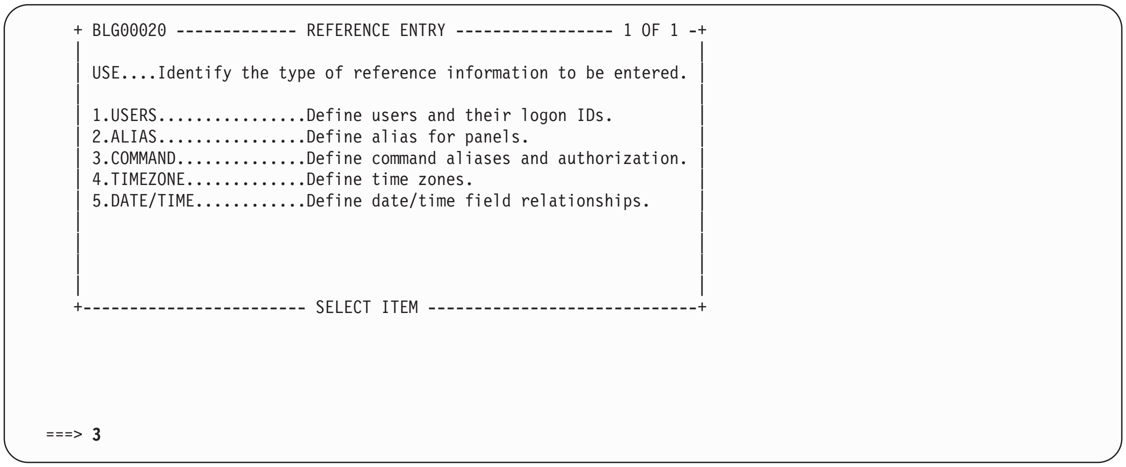
For this exercise, select **Reference** by typing **2** and pressing **Enter**.



```
       + BLG00010 ------------ SYSTEM RECORD ENTRY --------------- 1 OF 1-+
       |                                                                  |
       | USE....Identify the type of description (record) to be entered.  |
       |                                                                  |
       | 1.CLASS.............Define authority and users in a privilege    |
       |                     class record.                                |
       | 2.REFERENCE.........Define reference information.                |
       | 3.LOGSAVE...........Define information used by the Automatic     |
       |                     Log Save and DB2 Extract Facilities.         |
       | 4.MAP...............Define PostProcessor data mapping.           |
       | 5.INDEX.............Define index for text search.                |
       |                                                                  |
       |                                                                  |
       +------------------------- SELECT ITEM ----------------------------+









===>2
```

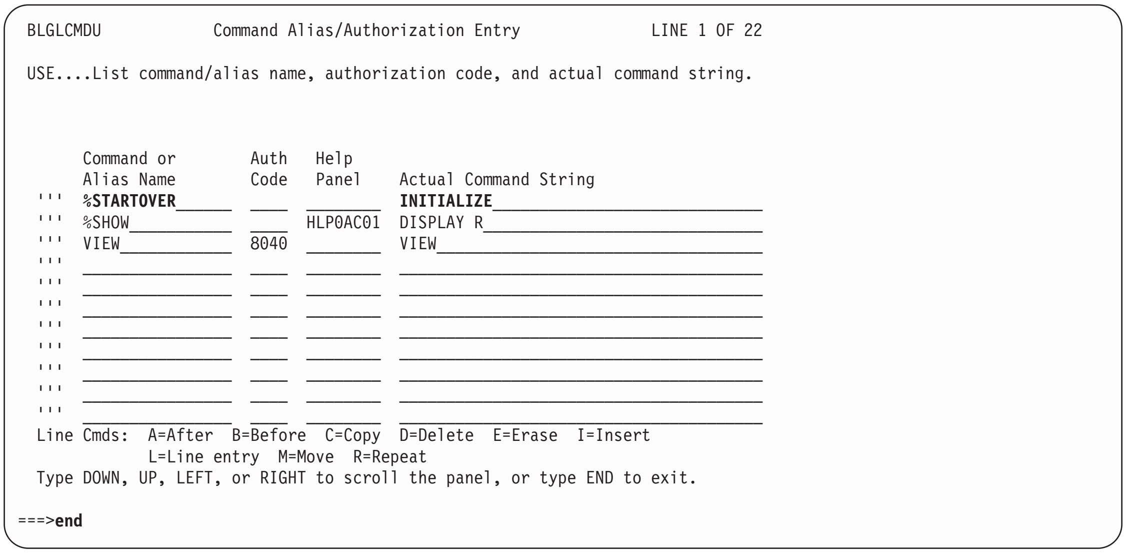
BLG00020, the Reference Entry panel, offers you a choice of creating three types of reference records. Choose **Command** by typing **3** and pressing **Enter**.

If you already have the COMMAND record in your system, this selection brings your existing record into update mode so you can add or change data.



```
+ BLG00020 ------------- REFERENCE ENTRY ----------------- 1 OF 1 -+
|                                                                  |
| USE....Identify the type of reference information to be entered. |
|                                                                  |
| 1.USERS...............Define users and their logon IDs.          |
| 2.ALIAS...............Define alias for panels.                   |
| 3.COMMAND.............Define command aliases and authorization.  |
| 4.TIMEZONE............Define time zones.                         |
| 5.DATE/TIME...........Define date/time field relationships.      |
|                                                                  |
|                                                                  |
|                                                                  |
|                                                                  |
+------------------------ SELECT ITEM -----------------------------+





===> 3
```

If you are creating the COMMAND record for the first time, an empty Command Alias/Authorization Entry panel appears. On this panel, list the **Command or Alias Name** for the commands you want to set up with aliases and specify the actual command string the alias represents. In this example, STARTOVER is identified as an alias name for the INITIALIZE command. **Auth Code** or **Help Panel** entries are optional. After you enter the data, type **end** on the command line and press **Enter** to save your entries.



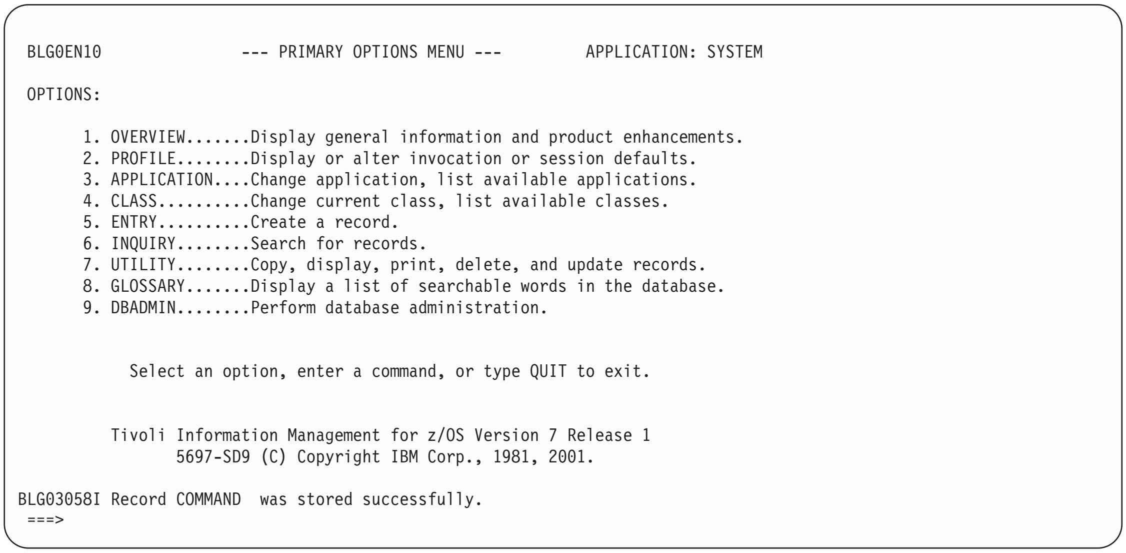
```
BLGLCMDU            Command Alias/Authorization Entry              LINE 1 OF 22

USE....List command/alias name, authorization code, and actual command string.



      Command or        Auth   Help
      Alias Name        Code   Panel     Actual Command String
 '''  %STARTOVER______  ____  ________  INITIALIZE_____________________________
 '''  %SHOW___________  ____  HLP0AC01  DISPLAY R______________________________
 '''  VIEW____________  8040  ________  VIEW___________________________________
 '''  ________________  ____  ________  _______________________________________
 '''  ________________  ____  ________  _______________________________________
 '''  ________________  ____  ________  _______________________________________
 '''  ________________  ____  ________  _______________________________________
 '''  ________________  ____  ________  _______________________________________
 '''  ________________  ____  ________  _______________________________________
 '''  ________________  ____  ________  _______________________________________
 '''  ________________  ____  ________  _______________________________________
 Line Cmds:  A=After  B=Before  C=Copy  D=Delete  E=Erase  I=Insert
             L=Line entry  M=Move  R=Repeat
 Type DOWN, UP, LEFT, or RIGHT to scroll the panel, or type END to exit.

===>end
```

Use **Auth Code** to associate an authorization code with the command alias. The authorization code consists of up to 4 digits. (For more information on how to create authorization codes, refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide*.) Use **Help Panel** to associate a help panel with the command alias. The help panel identifier consists of 8 alphanumeric characters.

Note: Only part of the **Command or Alias Name** and **Actual Command String** are shown on this panel. The **Command or Alias Name** can be 32 characters long. The **Actual Command String** can be 80 characters long so only part of the fields are actually shown on this panel. You can use the **RIGHT** and **LEFT** commands to scroll the panel. You can enter data which is longer by typing **L1** (for alias name) or **L4** (for actual command string) in the line command area to the left of the row you want to update. Then press **Enter** and type the field value on the assisted-entry panel. (You cannot, however, enter commas on the assisted-entry panel using this method.)

You can enter as many alias names as you need. If you run out of lines on the screen, you can use the I or R line commands to add lines to the record.

The Primary Options Menu panel appears with a message indicating the COMMAND record was stored successfully.



```
 BLG0EN10               --- PRIMARY OPTIONS MENU ---         APPLICATION: SYSTEM

 OPTIONS:

       1. OVERVIEW.......Display general information and product enhancements.
       2. PROFILE........Display or alter invocation or session defaults.
       3. APPLICATION....Change application, list available applications.
       4. CLASS..........Change current class, list available classes.
       5. ENTRY..........Create a record.
       6. INQUIRY........Search for records.
       7. UTILITY........Copy, display, print, delete, and update records.
       8. GLOSSARY.......Display a list of searchable words in the database.
       9. DBADMIN........Perform database administration.


            Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.

BLG03058I Record COMMAND  was stored successfully.
 ===>
```

To change the COMMAND record once it is saved, simply enter **5,2,3** from the System Primary Options menu (BLG0EN10) or use any supported method of record update (for example, the UPDATE command).

Usage Notes

- Users should enter command aliases the same way they enter normal commands. Any operands the user supplies are appended to the end of the actual command string associated with the alias, separated by a blank. For example, if SHOW is an alias for DISPLAY R, then SHOW 1234 will be processed as DISPLAY R 1234.
- Changes to the COMMAND record become effective the next time the user session interacts with the BLX-SP.
- If you set up a command alias with an actual command string that does not begin with a unique valid Tivoli Information Management for z/OS command, users will receive a message indicating the command is not valid when they invoke the alias name.
- Users and TSPs can specify a command using the minimum number of characters necessary to uniquely identify the command (for example, **se** can be used for SEARCH or **q** can be used for QUIT). If a command alias begins with the same characters as a Tivoli Information Management for z/OS command or another command alias, it could change the minimum number of characters needed to uniquely identify the command, and require that the user or TSP use a longer abbreviation of the command. To avoid having your command alias names conflict with existing commands, refer to the *Tivoli Information Management for z/OS User's Guide* for a list of Tivoli Information Management for z/OS commands and their acceptable truncations.

  Future commands or aliases may be added to the Tivoli Information Management for z/OS product which may be identical to a command alias you have defined for your users. This situation could cause the minimum abbreviation requirements to change. To avoid this possibility, start the command alias names with a **$, #, @,** or **%** character. Tivoli Information Management for z/OS will not use these characters as the first character of a command. Additionally, you should use full command names in a TSP, TSX, or PF key definition to avoid the need to change them should the minimum abbreviation become longer in the future.
- You can use the PRINT command to print the COMMAND record.

8

Setting Up Stored Response Chains

You can establish a set of stored response chains (SRCs) that enable users to perform repetitive tasks quickly, if you are the program administrator or other authorized user. Also, you can establish SRCs that enter certain data in formats that are appropriate for your organization.

Because an SRC follows the same prompting sequence that a user follows when entering data, you can build an SRC for any prompting sequence that a user follows repeatedly. In addition to being faster, creating SRCs ensures that users follow a particular prompting sequence and enter certain data values. An SRC prompting sequence encourages a consistent method of data entry that helps ensure users can retrieve data as they need it.

Assessing Needs for SRCs

Before setting up SRCs for your organization, assess the needs of each user group. Base your decision on such factors as user experience, need for flexibility and speed, and your guidelines for data collection. If, for instance, each user deals with different types of problems or changes, organization-wide SRCs cannot provide the flexibility your site requires. For greater flexibility and usability, encourage users to design their own SRCs.

It might be useful to establish meaningful naming conventions and good descriptions for SRCs so users can recall the name of the SRC they want to use, or can recognize it in a list of SRCs. For instance, all SRCs for creating records might start with C, followed by the user's initials, followed by a specific identifier.

Planning SRCs

If you have authority, you can create SRCs for all your Tivoli Information Management for z/OS users. An SRC record consists of two parts: descriptive information and SRC responses. For each SRC that you create, you must decide which responses you want entered by the SRC and which response you want entered by the user.

Before creating an SRC, ask yourself:

- Is there a related SRC that I could use as a guide for creating my new SRC?
- From which panel should the SRC start? Keep in mind that SRCs starting on the Primary Options Menu can also be processed when Tivoli Information Management for z/OS is started, thus bypassing the display of that panel.
  - Should the SRC be able to start from any application? You can start an SRC with ;;SUSPEND or ;;INITIALIZE and subsequently select a specific application so that the SRC can be used with any application. Or, if you use only one application, the **Invocation application** field in the profile can remain set to that application. If you

start the SRC with the SUSPEND command, use the RESUME command as either the last command in the SRC (input as ;;RESUME), or in the final response (input as RESUME).

  - Should the SRC be able to start from any panel? By beginning an SRC with certain commands (REPORT, for example), any user can, from any panel, use SUSPEND to temporarily leave the current dialog. Then the user can perform searches, print reports, or change the output destination.

- Should you change the responses of the SRC? You can display and modify only responses within a dialog that begins within the SRC with the CHANGE command. You cannot display or modify any responses before the first collected dialog with the CHANGE command if an SRC begins in the middle of a dialog.
- Will the SRC run differently based on a user's profile setting? If so, start the SRC with the PROFILE command and change the settings that affect the processing of the SRC. Temporarily save the profile and end the SRC with **PROFILE,8** to restore the user's profile to the usual settings.
- Will a search argument in the SRC yield different results based on the length of the SDIDS key? If an SRC contains a fully qualified search argument written for an SDIDS with an 18-byte key, the search results may be different with an SDIDS with a 34-byte key. To get the same results with SDIDSs with both the shorter and longer keys, add a period to the end of the fully qualified search argument written for an SDIDS with a 34-byte key. Tivoli Information Management for z/OS will ignore the characters following the period if any exist in the data record. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for more information about the lengths of SDIDS keys.
- What data should the SRC provide? Can any common information be entered by default without limiting this SRC to only a few users?
- What data should the user provide? At what points in the prompting sequence should the SRC stop and let the user enter data that is unique to the particular task, such as the user's name or an ABEND code?
- What panel should be displayed to the user when the SRC finishes processing? This panel can be a search results list or a panel that requests additional data.
- Is this SRC flexible enough for a number of users and yet specific enough to be of value?
- Which users can use the SRC? The privilege class of the user creating the SRC is automatically entered as the only class that can use the SRC. If you want other users to use the SRC, either clear the privilege class field to authorize all users, or add each privilege class on the SRC Execute Classes panel. Master privilege class automatically has the authority to use any SRC.

For information on how to create SRCs, refer to the *Tivoli Information Management for z/OS User's Guide*.

Modifying and Controlling SRCs

To meet more specific or changing needs, you can modify the SRCs created at your organization. To make sure that you can modify the SRCs you create, document each one the same way:

- Use the description of the SRC that you are modifying as a guide to its contents.

- Note such specifics as the name, the starting panel, and the default data.
- Decide which responses in the SRC you want to change.

To modify an SRC, you need SRC update authority. Refer to the *Tivoli Information Management for z/OS User's Guide* for information on how to update an SRC. After you make all your changes, file the SRC.

Rules for Modifying SRCs

When you modify SRCs, you cannot do the following:
- Add new panels to the prompting sequence
- Change freeform text
- Modify information within a dialog that began before running the SRC

When you modify SRCs, you can do the following:

- Change data in the existing prompting sequence in the SRC and data in the final responses at the end of the SRC.
- Delete commands and the ;;commands collected, not change them. If you delete the ;;PROFILE or ;;REPORT command, the system deletes all responses associated with the command.

If the changed item does not alter the subsequent panel flow, the data you specify replaces the previous data, and the Changed Collected Data panel reappears. If the changed item alters the subsequent panel flow, a warning panel appears asking you to verify the change in the panel flow.

Deleting SRCs

Instead of modifying an existing SRC, you can delete it and create a new one. SRCs, like other records, are deleted with the DELETE command or D (delete) line command on a search results list. Refer to the *Tivoli Information Management for z/OS User's Guide* for information on working with records and deleting them. Only users with SRC delete authority can delete SRCs.

Controlling SRCs

You can control who creates, updates, and deletes SRCs through privilege class authorities and record ownership. Only users with the appropriate authority can perform these functions. This control is useful in limiting the number of SRCs created and in identifying who is responsible for maintaining them.

Once created, SRCs are available for use by authorized users or by all users if no SRC-execute classes are specified. A user does not need authority to display or print SRCs. A user can display descriptive information about the SRC (such as the SRC description, the start panel name, or the application) and the actual responses in the SRC. The user can also find an appropriate SRC by using information from the description field in a search argument. If a user specifies the EXECUTE command with no operands, a list is displayed of SRCs that the user is authorized to run.

9

Using OS/390 Text Search for Improved Problem Resolution or Freeform Text Searches

As an administrator for the Tivoli Information Management for z/OS database, you can take advantage of the freeform text stored in records in the database to create a knowledge base of information that can have great potential value to your enterprise. For example, if you use Tivoli Information Management for z/OS to support help desk operations, you can use the text stored in your existing problem records to create solution records that analysts and users can search to help solve new problems. By maintaining problem and solution information in this way, your help desk staff can operate more efficiently and your customers can benefit from shorter problem durations.

The indexing and searching of freeform text can be done through use of OS/390 Text Search, which is available with OS/390 Release 2.8 or later. A component of OS/390 Text Search, the Text Search Engine, can be used by Tivoli Information Management for z/OS to index and search the freeform text data associated with Tivoli Information Management for z/OS records. The Text Search Engine is available as part of OS/390 UNIX System Services.

You can also create solution records and search freeform text from the Desktop. Information on how so can perform these functions is described in the *Tivoli Information Management for z/OS Desktop User's Guide*. If you use the Desktop, you must still perform the setup tasks described in "Getting Started" on page 161.

Whether or not you decide to create a knowledge base of problem solution information, your users can still take advantage of the Text Search Engine to search data stored in freeform text portions of records in your database.

To take advantage of the Text Search Engine, you must first perform some setup and customization tasks as described in this chapter. This chapter does not explain everything there is to know about the OS/390 Text Search or its Text Search Engine; rather, it describes the setup and administrative tasks associated with using the Text Search Engine with Tivoli Information Management for z/OS. For more information about OS/390 Text Search, you can review these publications:

- *OS/390 Text Search: Installation and Administration for the Text Search Engine*
- *OS/390 Text Search: Programming the Text Search Engine*

Note: It is assumed that you are familiar with these publications and that the Text Search Engine is already installed. If the Text Search Engine is not installed, it is recommended that your systems programmer or installer perform a customizable installation as described in *OS/390 Text Search: Installation and Administration for the Text Search Engine*. The follow-on Text Search Engine customization steps required for Tivoli Information Management for z/OS will add additional files used by the Text Search Engine to communicate with Tivoli Information Management for z/OS.

Also, depending on how job responsibilities are divided at your company, you may need to know how to issue Text Search Engine commands natively to perform index and document administration or to diagnose problems related to the use of the Text Search Engine. Tivoli Information Management for z/OS provides administrative support for index creation through the Text Search Engine. However, there may be circumstances where you will need to perform a task natively in the Text Search Engine environment because the Tivoli Information Management for z/OS panel interface does not enable you to perform the same function. Therefore, it is generally a good idea to be somewhat familiar with the Text Search Engine documentation and administrative procedures.

Depending on how your organization is structured, most of the administrative tasks associated with implementing a knowledge base or using the Text Search Engine can be performed by a single systems administrator. However, your organization may also have a ″solutions″ administrator who is involved primarily with the maintenance of a knowledge base. Solution administrator tasks can include creating knowledge base records on a regular basis and updating those records when necessary. Both types of administrative tasks are included in this chapter. This chapter describes how to perform the following tasks:

- Customize the Text Search Engine.
- Start the Text Search Request Processor.
- Create an index to be used by the Text Search Engine.
- Index the text data.
- Perform maintenance on the index.
- Create and maintain the records providing solutions to problems.

Knowledge Base and Freeform Text Search Concepts

Tivoli Information Management for z/OS provides a way for you to create solution records from problem record data, and to search the freeform text data in the solutions or in other types of records (such as problem or change records). Your help desk analysts and end users can use a panel interface to search for text data in the records. The results are displayed on a search results table panel.

Solution records can contain the following data:

- SCIM data (system, component, item, module)

  SCIM data represents a hierarchy of four levels of components at your company. For example:

  ```
  System:    INTRANET
  Component: WEBSITE
  Item:      PRODINFO
  Module:    HTML
  ```

The exact definition of a system, component, item, or module depends on the decisions made by your company, but Tivoli Information Management for z/OS interprets these fields as follows in problem records:

```
System          System name
Component       Device name
Item            Key item affected
Module          Program name
```

- One-line abstract description (Tivoli Information Management for z/OS interprets this field as the *Description* data in a problem record.)

  For example: `Link on PRODINFO Web site not working`

- Problem description (Tivoli Information Management for z/OS interprets this field as the *Description freeform text* data in a problem record.)

  For example:

  ```
  Customer cannot seem to access the new product info
  for the ABC widget announced last month. When they
  click on the link for ABC they get a File Not Found
  error. Problem is reproducible. Who moved the files?
  This was working last week.
  ```

- Resolution information (Tivoli Information Management for z/OS interprets this field as the *Resolution freeform text* data in a problem record.)

  For example:

  ```
  File Not Found error was corrected by restoring
  the HTML file to its proper location on the web
  server. ABC widget file information was unintentionally
  erased by Marketing. Employees responsible for updating
  files should be instructed to follow the procedures
  available on prodinfo/howtoupd.htm to update product
  files.
  ```

The combination of all this type of information represents the Tivoli Information Management for z/OS *knowledge base*. Although Tivoli Information Management for z/OS provides support for setting up a knowledge base of problem resolution data generated through solution records, you can also create search indexes for other types of records you may have in your Tivoli Information Management for z/OS database. As provided, Tivoli Information Management for z/OS enables you to create the following types of indexes for freeform text searches:

- Solution index – To search description and resolution freeform text in solution records
- Problem index – To search description freeform text in problem records
- Change index – To search description and resolution freeform text in change records
- User-defined indexes – To search freeform text in records of your choice

Tivoli Information Management for z/OS provides data view and data attribute records that enable you to index description freeform text for problem records and description and resolution freeform text for change and solution records. If you wish, you can modify these data model records or create your own data model records to index additional types of freeform text.

After the data is indexed, you can exploit the power of the OS/390 Text Search Engine to search the freeform text associated with those records.

For example, suppose you had the following freeform text associated with a problem record in your database. In this example, the data is being viewed through the Tivoli Information Management for z/OS Info editor:

```
BLG1TDES                        DESCRIPTION TEXT                     LINE 1 OF 14

''''   04/26/2000 User cannot print properly to new printer installed in
''''              building 600. Printer ID is 600B13A. Output comes out on
''''              all blue separator paper and pagination is wrong.
''''              IP address of printer is 111.22.33.444. User mentioned
''''              problem last week to Lance Harris but is still
''''              having problems.
''''
''''
''''
''''
''''
''''
''''
''''
****   ********** *** BOTTOM OF DATA ***
```

By creating a problem index that will be used by the Text Search Engine, you can enable your users to search on any of the freeform text data shown in the paragraph. This could be useful to help desk analysts who need to ″drill down″ into existing problem records to find information in the database. For example, a help desk analyst could search on the specific IP address mentioned, the word ″blue,″ or perhaps a specific employee name if mentioned in the problem description.

Likewise, a change index could be used to search on the description and resolution freeform text associated with change records.

A solution index can be created to search on freeform text associated with problem resolution data normally stored in problem records. However, the solution index does not search problem records directly. You must first create the solution records as a separate type of record in your database. A solution administrator can create a solution record by using data entry panels or automatically when filing a problem record. See “Creating a Solution Record Using Data Entry Panels” on page 182 for information on how to create a solution record using the data entry method. For information on how to create a solution record when filing a problem record, refer to “Creating a Solution Record When Filing a Problem Record” on page 185.

The freeform text you can search on includes paragraph-type descriptive text as well as the one-line description abstract of the solution.

General users and help desk analysts can enter text search arguments through a Tivoli Information Management for z/OS panel interface to search on freeform text data. For information about entering searches for Tivoli Information Management for z/OS freeform text data through use of the Text Search Engine, users can refer to the *Tivoli Information Management for z/OS User’s Guide*. A summary of how to do freeform text searching is also provided in this chapter for your benefit.

Searches of freeform text data can also be performed through the Tivoli Information Management for z/OS application program interfaces (APIs). Information on searching through the APIs is provided in *Tivoli Information Management for z/OS Application Program Interface Guide*.

Getting Started

Before you can use the Text Search Engine to find freeform text data associated with Tivoli Information Management for z/OS records in your database, you must perform some customization and setup tasks for your environment. An illustration of a typical environment is shown in Figure 5.

The OS/390 Text Search component of OS/390 is required.

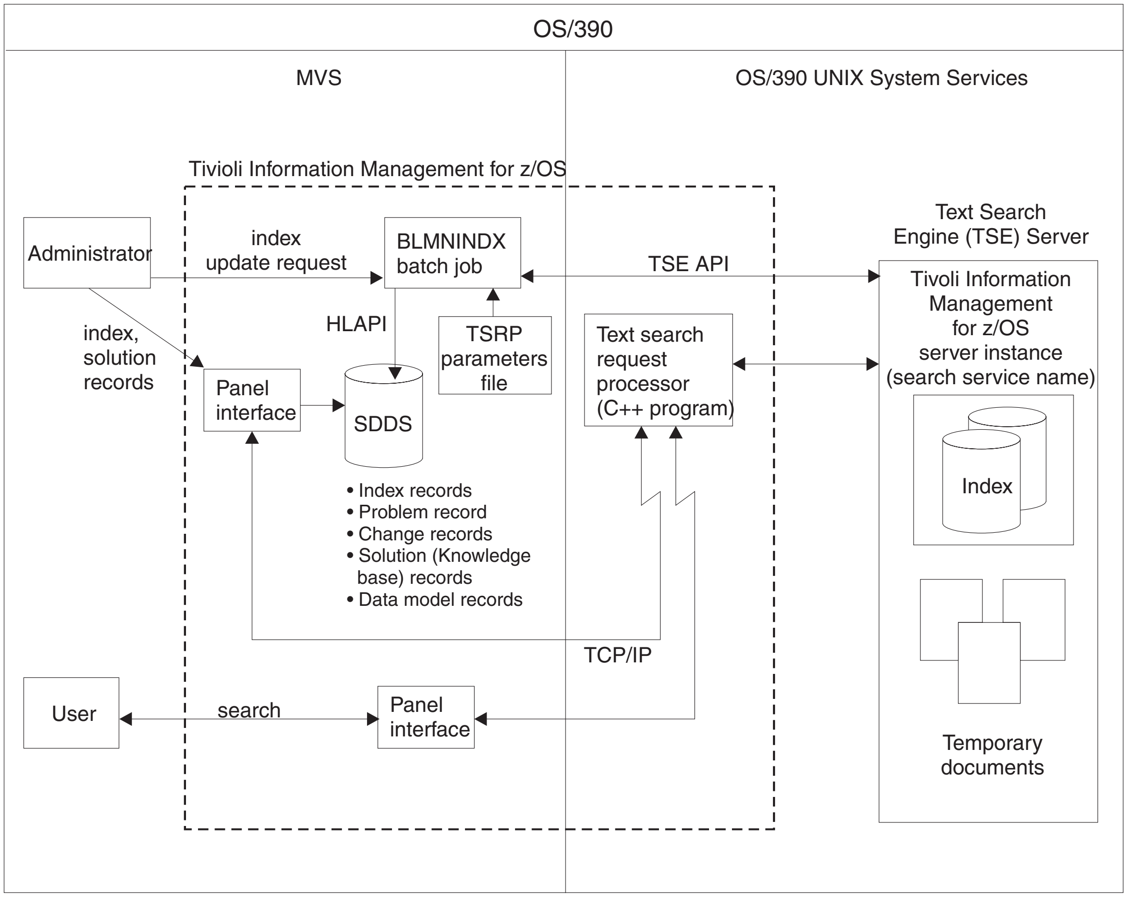


Figure 5. Using the Text Search Engine with Tivoli Information Management for z/OS. The BLMNINDX batch job uses the Tivoli Information Management for z/OS high-level API and the Text Search Engine API to create a text search index in OS/390 UNIX® System Services.

As illustrated in the diagram, a Tivoli Information Management for z/OS batch job (BLMNINDX) is run to load the freeform text extracted from the Tivoli Information Management for z/OS database into the the text search index that ultimately resides in an OS/390 UNIX directory. A Tivoli Information Management for z/OS text search request processor program residing in OS/390 UNIX acts as an interface between Tivoli Information Management for z/OS and the Text Search Engine. A parameters file governs how the Text Search Engine session with Tivoli Information Management for z/OS should be controlled.

The customization and setup tasks include the following:

Customization

- Update the Text Search parameters file (TSRP parameters member BLMTSRPP).
- Complete customization of the Text Search Engine (sample BLMTSRCJ).

- Create the cataloged procedure to start the text search request processor (sample BLMTSRP).
- Start the search request processor.

These activities are described in “Customizing the Text Search Engine (Sample BLMTSRCJ)” on page 165.

Setup for Tivoli Information Management for z/OS

- Update privilege classes.
- Load the appropriate data model records.
- Perform any necessary tailoring.
- Create the text indexes.

These activities are described in “Setting up Tivoli Information Management for z/OS” on page 167.

Customizing the Search Environment

Before you can use the Text Search Engine to create a knowledge base or perform freeform text searches, you must customize the Text Search Engine environment for use with Tivoli Information Management for z/OS.

Updating the Text Search Request Processor Parameters (Sample BLMTSRPP)

Before you can index Tivoli Information Management for z/OS records for use with the Text Search Engine, you must customize the parameters file used by the text search request processor. The parameters file is an OS/390 environment file that governs how the text search request processor should operate and includes parameters for controlling the Text Search Engine session and the Text Search Engine API sessions used by the BLMNINDX batch job when building or updating an index. You specify these parameters in a text search request processor parameters member. A sample parameters member, BLMTSRPP, is provided in the SBLMSAMP library. You can tailor this sample or create your own.

The text search request processor parameters member is installed by default in a partitioned data set in MVS. You can, if you wish, install it as an HFS file in OS/390 UNIX System Services.

Note: Some text search request processor parameters specify the file location. Because the BLMNINDX batch job does not create directories if they do not already exist, you should ensure that the directory locations specified in the parameters exist before running the job.

Before modifying the BLMTSRPP parameters, note the following:

- The data set must be either an MVS variable record length data set, an MVS fixed block data set, or an HFS file. If it is an HFS file, the text search request processor must have read access to the file.
- You can define only one parameter per line.
- Any data after the equal sign (=) is treated as a value for the parameter. Do not include sequence numbers or spaces.

- To add a comment, place the comment on a line by itself. Each comment must begin with a pound sign (#) in the first column.

Controlling the Text Search Engine Session

The following BLMTSRPP parameters control the Text Search Engine session. Tailor these parameters to your environment before customizing the Text Search Engine environment. The directory locations specified in these parameters refer to directory paths in OS/390 UNIX System Services.

IMOCONFIGSRV=*server path*
: Specify the location of the Text Search Engine server configuration file. The default location is /etc/TextTools/TextSearch/server.

IMOCONFIGCL=*client path*
: Specify the location of the Text Search Engine client configuration file. The default location is /etc/TextTools/TextSearch/client.

IMOLANG=*lang*
: Specify the languange identifier for the indexes. The default is 6011, U.S. English. The valid values for IMOLANG are defined in /usr/lpp/TextTools/include/imolang.h.

IMOCCSID=*ccsid*
: Specify the coded character set identifier for the indexes. The default value is 500, International Latin - 1. The valid values for IMOCCSID are defined in /usr/lpp/TextTools/include/imoccsid.h.

SEARCH_SERVICE_NAME=*service name*
: Specify the name (up to 8 characters) of the Text Search Engine search service to use. The default value for Tivoli Information Management for z/OS is IMSRCH. (The Text Search Engine command **imocrcl** also creates a client profile or search service name.)

SERVER_INSTANCE_NAME=*service instance*
: Specify the name (up to 8 characters) of the Text Search Engine search server instance. A server instance defines a particular set of indexes that users can work with. The default value for Tivoli Information Management for z/OS is INFOMAN. (The Text Search Engine command **imocrins** also creates a server instance.)

RESOURCE _PATH=*resource path*
: Specify the location of the Text Search Engine dictionary files. The default location is /usr/lpp/TextTools/dict.

INDEX_DATA_PATH=*index data path*
: Specify the location of the index data files. The default location for Tivoli Information Management for z/OS indexes is /etc/InfoMan/TextSearch/indexes. The files that are created to build a text search index are placed in this location.

TEXT_PATH=*directory name*
: Specify the location of the Tivoli Information Management for z/OS *xxxxxxxx*.txt files. This is a work directory for the BLMNINDX batch job. Text files are written to this directory while a text index is updated. When the index update is complete, the files are removed from this directory. The default location is /etc/InfoMan/TextSearch/txt.

NLSPATH=*directory name*
: Specify the location of the Tivoli Information Management for z/OS message

catalog. The default location is /usr/lpp/InfoMan/TextSearch/nls/msg/%N:/usr/lpp/TextTools/nls/En_US/%N.

PATH=*directory name*
Specify the location of the Text Search Engine directory for executables. The default location is /usr/lpp/InfoMan/TextSearch/bin:/usr/lpp/TextTools/bin.

LIBPATH=*directory name*
Specify the location of the Text Search Engine directory for libraries. The default location is /usr/lpp/InfoMan/TextSearch/lib:/usr/lpp/TextTools/lib.

Controlling the TCP/IP Session

The following BLMTSRPP parameters define the TCP/IP session users will use for freeform text searching. Tailor these parameters to your environment.

MAXCONNECT=*10*
Specify the maximum number of client conversations that the text search request processor can run concurrently. Valid values are from 1 to 100. The default value is 10.

MAXQUEUED=*1*
Specify the maximum number of client conversations that are waiting to connect to the text search request processor. Valid values are from 1 to 10. The default value is 1.

COLUMNS=*132*
Specify the column width of the TSRP log file. The default is 132.

Controlling the BLMNINDX API Sessions

The following BLMTSRPP parameters control the Text Search Engine API sessions used by the BLMNINDX batch job. Tailor these parameters to your environment.

PRIVILEGE_CLASS=*privilege class*
Specify the privilege class name that must exist in the Tivoli Information Management for z/OS database. The class must have display and update authority for the records being processed and database administration authority to read the index records. This parameter is required; no default is provided. The PRIVILEGE_CLASS parameter sets the PICACLSN field for the API transactions that are performed.

The privilege class specified here is the privilege class associated with the user ID specified in the APPLICATION_ID parameter.

The PRIVILEGE_CLASS, SESSION_MEMBER, and APPLICATION_ID parameters contain values needed to initialize a Tivoli Information Management for z/OS HLAPI session. They have no relationship to the actual records that are indexed.

SESSION_MEMBER=*member name*
Specify the session-parameters member name that the BLMNINDX job should use to access the Tivoli Information Management for z/OS database (for example, BLGSES00). The database must be accessed as database 5, the read/write database.

APPLICATION_ID=*user ID*
Specify the application identifier of the BLMNINDX job. For simplicity, it is recommended that this user ID be the same as the user ID specified in the job card used to start the text search request processor (see the description of the BLMTSRCJ

JCL in “Customizing the Text Search Engine (Sample BLMTSRCJ)”). This user ID must exist in the privilege class defined by the PRIVILEGE_CLASS parameter. The APPLICATION_ID parameter is required; no default is provided.

DEBUG={NO|YES}
Specify whether the Tivoli Information Management for z/OS writes out API messages. When DEBUG=YES is specified, LLAPI messages are written to the data set specified by the APIPRINT DD statement in the BLMNINDX batch job. The HLAPI messages are written to the data set specified by the HLAPILOG DD statement. If you are analyzing problems, DEBUG=YES can facilitate problem determination.

SEARCH_ARGS=*search criteria for records to index*
Specify freeform search arguments used to identify the records to be indexed. This field is a blank delimited field. Specify arguments on a single line and include the actual p-words. P-words must be included because alias table processing is not used. For example:

```
SEARCH_ARGS=DATE/2001/02/20 - DATE/2001/03/20 STAC/CLOSED
```

Each argument is taken to perform a combined search. This example retrieves records for indexing that were created between February 20, 2001 and March 20, 2001.

This is an optional parameter. No default value is provided. If this parameter is omitted, all new or updated records are retrieved for indexing.

NUMBER_OF_HITS=*maximum number of records to process*
Define the number of records to be processed for each ndex. The value specified must be an 8-digit number between 00000001 and 99999999. This parameter can be used alone or with the SEARCH_ARGS parameter to limit the number of records processed.

This is an optional parameter. If no value is specified, all of the new and updated records are processed.

INDEX=*index record ID*
Specify the name of the text index to be updated. The name is the record ID that was set up for a particular index record in Tivoli Information Management for z/OS. The INDEX parameter is optional. If no value is specified, all of the Tivoli Information Management for z/OS text indexes are updated. If a value is specified (for example, MYINDEX), only that index is updated.

RECORDS={UPDATED|ALL}
Specify which records should be indexed. You can specify that ALL records associated with the specified index should be indexed, or only those records that have been added or updated since the last time BLMNINDX was run (UPDATED). This parameter is optional; the default value is UPDATED. When RECORDS=ALL is specified, the INDEX parameter is required. ALL is used to rebuild the entire contents of the index and should be used only if the Text Search index must be rebuilt.

Customizing the Text Search Engine (Sample BLMTSRCJ)

Before using the Text Search Engine or starting the text search request processor, you should ensure that the Text Search Engine environment is customized for Tivoli Information Management for z/OS. You can run the BLMTSRCJ sample JCL provided in the Tivoli

Information Management for z/OS SBLMSAMP sample library to complete this customization. The job runs a REXX program, BLMTSRCH, which does the following:

- Creates a Text Search Engine server instance
- Creates a search service name (client profile) for Tivoli Information Management for z/OS. The client profile provides information required by the Text Search Engine to interface with Tivoli Information Management for z/OS, such as the identification of the communication method that will be used to connect to the Text Search Engine server.

This is an example of BLMTSRCJ.

```
//TSRC1    JOB (account),REGION=6M
//*                                                                   *
//*                                                                   *
//* This is sample JCL for running the REXX exec that completes setup *
//* of the Intelligent Miner for Text (IM4T) Text Search Engine (TSE) *
//* for Tivoli Information Management for z/OS.                       *
//*                                                                   *
//* You must update the parm field as appropriate and also change the *
//* path names for DD names STDOUT, STDERR, and STDENV.               *
//*                                                                   *
//*                                                                   *
//*-------------------------------------------------------------------*
//STEP1  EXEC PGM=BPXBATCH,
//            PARM='PGM /usr/lpp/InfoMan/TextSearch/bin/blmtsrch'
//*
//STDOUT   DD PATH='/tmp/InfoMan/blmtsrch.out',
//            PATHMODE=SIRWXU,PATHOPTS=(OWRONLY,OCREAT,OTRUNC)
//*
//STDERR   DD PATH='/tmp/InfoMan/blmtsrch.err',
//            PATHMODE=SIRWXU,PATHOPTS=(OWRONLY,OCREAT,OTRUNC)
//*
//STDENV   DD DSN=BLM.SBLMSAMP(BLMTSRPP),DISP=SHR
//
```

Starting or Stopping the Text Search Request Processor (Sample BLMTSRP)

The text search request processor (TSRP) is a started task that maps requests from Tivoli Information Management for z/OS into Text Search Engine API calls and returns the results to Tivoli Information Management for z/OS. It is used to create and update the text indexes and to process search requests.

The TSRP started task must run with the user ID assigned to the Text Search Engine. You must set up RACF® or a similar security product to associate the cataloged procedure for the TSRP with the appropriate user ID. Use the same user ID that ws created by the installer of OS/390 Text Search. The default Text Search Engine user ID is WEBADM.

A sample procedure to start the text search request processor is provided in the SBLMSAMP sample library (BLMTSRP). You must update the PARM field as appropriate and change the data set names for DD names STEPLIB, VARS, SYSPRINT, SYSOUT to correspond to the data set names at your installation.

Note: You must store the procedure to start the text search request processor as a cataloged procedure in SYS1.PROCLIB.

```
//*                                                                   *
//*  This sample catalogued procedure starts the text search request
//*  processor (TSRP) address space using the sample parameters data
//*  member BLMTSRPP.
//*
//*-------------------------------------------------------------------*
```

```
//*  Update the following with uppercase text:
//*
//*      FROM                     TO
//* ==================================================================
//* BLM.SBLMMOD1       - The actual name of the Tivoli Information
//*                      Management for z/OS data set used at your
//*                      installation.
//* BLM.SBLMSAMP       - The name of the TSRP parameters data set used
//*                      at your installation.
//* CEE.SCEERUN        - The name of the LE run time library
//*                      at your installation.
//* CBC.SCBCCMP        - The name of the C run time library
//*                      at your installation.
//* BLMTSRPP           - The name of your TSRP parameters member.
//*
//*  Store the updated procedure as a member in your system's procedure
//*  library, such as SYS1.PROCLIB.
//*
//*********************************************************************
//BLMTSRP PROC
//BLMTSRP EXEC PGM=BLMTSRP,REGION=32M,
// PARM=('ENVAR(_CEE_ENVFILE=DD:VARS),HEAPPOOLS(ON),RPTSTG(ON)/')
//STEPLIB  DD   DISP=SHR,DSN=BLM.SBLMMOD1
//         DD   DISP=SHR,DSN=CEE.SCEERUN
//         DD   DISP=SHR,DSN=CBC.SCBCCMP
//VARS     DD   DISP=SHR,DSN=BLM.SBLMSAMP(BLMTSRPP)
//OUTDSC   OUTPUT DEST=HOLD
//CEEDUMP  DD   SYSOUT=*,OUTPUT=(*.OUTDSC)
//PARM     DD   SYSOUT=*,OUTPUT=(*.OUTDSC)
//SYSPRINT DD   SYSOUT=*,OUTPUT=(*.OUTDSC)
//SYSOUT   DD   SYSOUT=*,OUTPUT=(*.OUTDSC)
//SYSIN    DD   DUMMY
//          PEND
//
```

To start the text search request processor, enter the MVS system operator command START for the cataloged procedure BLMTSRP. For example:

```
S BLMTSRP
```

The text search request processor must be started for search activity to take place or before you can create or update a Tivoli Information Management for z/OS text search index in OS/390 UNIX System Services.

To stop the text search request processor, enter the MVS system operator command STOP. For example:

```
P BLMTSRP
```

If you are shutting down the system before an IPL, or if you want to prevent users from accessing text indexes for some reason (perhaps if you are working on a problem), you may want to shut down the text search request processor.

Setting up Tivoli Information Management for z/OS

To set up Tivoli Information Management for z/OS for use with the Text Search Engine (in order to create a knowledge base or to do freeform text searching), follow these procedures:

- Update privilege classes. If you plan to use solution records, add solution record authority to the appropriate privilege classes of users who will maintain those records. Ensure that users have the appropriate privilege classes and authorities to work with the records they will be searching on. Users who have display authority for problem records can search solution records.

- If they have not been loaded already, load the data model records required to support use of OS/390 Text Search with Tivoli Information Management for z/OS data. You can use the BLHRCDSL TSX to do this.

  Instructions on how to load data model records with the BLHRCDSL TSX are provided in the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference*. The required records are provided in the base product support list and can be loaded if BLHLRALL is specified to load all data model records.

- Create additional data model records as necessary if you have customized panels. Data view records and data attributes are provided with Tivoli Information Management for z/OS to support freeform text searches, but these records are based on certain s-word values. If you use different s-word values, you will need to update the records:

  - For problem records: Data view record BLHRPRBT retrieves the description abstract (s-word index 0E0F) and description freeform text (s-word index 0E01) from problem records for indexing. If you use different s-word indexes for these fields, you must create data attribute records for the fields. Then, update BLHRPRBT to use the new data attribute records.
  - For change records: Data view record BLHRCHGT retrieves the description abstract (s-word index 0E0F), description freeform text (s-word index 0E01), and resolution freeform text (s-word 0E03) from change records for indexing. If you use different s-word indexes for these fields, you must create a data attribute record for the fields and update BLHRCHGT to use the new data attribute records.

- Control panel BLH1S013 maps the fields from the problem record to the solution record when an existing problem record is used to create the new solution record. If you intend to use solution records and if you use different s-words in your problem records or want to map different fields to the solution record, you must update this control panel.

- When you create a solution record from a problem record by selecting option **10. Create solution and file record** on BLG0BU00, the Problem Record Summary panel, or by using the data entry panels and specifying a problem record ID in the Problem record field, the following data is checked:

  - The status of the problem record is CLOSED.
  - The problem record contains description freeform text (S0E01).
  - The problem record contains resolution freeform text (S0E03).

  If your problem management application does not use STAC/ or 'CLOSED' for the status, or does not use description or resolution freeform text, or uses different s-words for the freeform text data, you must change TSXs BLHTXSOL and BLGTXSPR to match your application. TSXs provided with Tivoli Information Management for z/OS are available in the SBLMTSX data set.

- When you create a Text Search Engine index for problem records, change records, or solution records through the panels provided with Tivoli Information Management for z/OS, the following corresponding index records are created in Tivoli Information Management for z/OS database 5:

Text Index Name	Record ID	Description
INDXPROB	INDXPROB	Problem description freeform text
INDXCHNG	INDXCHNG	Change description freeform text
INDXSOLN	INDXSOLN	Solution description freeform text

These record IDs are predefined. If you already have records in your database with these record IDs and need to retain what you have, you will need to specify other record IDs for Tivoli Information Management for z/OS to use for the indexes. Follow these procedures:

- Use the Panel Modification Facility (PMF) to update the corresponding terminal simulator panel (TSP). Change the existing record ID to the new value you want the indexing process to use:

TSP	Existing Record ID	Description
BLGRFPRO	INDXPROB	For indexing problem freeform text
BLGRFCHG	INDXCHNG	For indexing change freeform text
BLGRFSOL	INDXSOLN	For indexing solution freeform text

- Add the new RNID/ p-word values to the dictionary.
- Update panel BLG6URN5 with the new p-word values.
- Copy the updated TSPs and panel to the read panel data set.
- Update the BLHTXNDX TSX to change the corresponding When statement to the new index name. The statement for the problem index, for example, looks like this before you change it:

```
When tscacrid = 'INDXPROB' Then    /* Creating the problem index    */
```

Understanding Access Control

If you are installing and using the OS/390 Text Search Engine for the first time, this section can help you to understand how access is controlled so that you can perform your customization and setup tasks more efficiently.

General users

Tivoli Information Management for z/OS users who need to do freeform text searching require a Tivoli Information Management for z/OS signon (a valid MVS/TSO user ID), including a privilege class assignment and authority for the record type searched. The TSO user ID should have an OMVS segment to enable the user to access OS/390 UNIX System Services.

Administrators

The Tivoli Information Management for z/OS administrator or solution administrator requires an OMVS segment in addition to a Tivoli Information Management for z/OS signon to create indexes, run the BLMNINDX batch job, and perform index maintenance from the Tivoli Information Management for z/OS environment.

If the administrator is responsible for starting and stopping the Tivoli Information Management for z/OS text search request processor, MVS operator authority is required to run the TSRP started task. The administrator must also have DBADMIN authority (in order to create index records) and must also have authority to work with solution records and authority to work with problem records. An OMVS segment and TSO signon are not required to start and stop the text search request processor because Tivoli Information Management for z/OS handles starting the processor.

Installer
The person who installs the OS/390 Text Search Engine must have an MVS/TSO user ID, an OMVS segment to use OS/390 UNIX System Services, and must create an administrative Text Search Engine user ID as part of installing OS/390 Text Search.

When the Text Search Engine is installed, a user ID (UID) and group ID (GID) are created. These values by default are WEBADM and IMWEB respectively, although other names can be defined. The Text Search Engine user ID WEBADM is an OS/390 UNIX System Services user only; it cannot be used for logon purposes.

Text Search Engine user ID
This user ID is created during installation of Text Search Engine and additional Text Search Engine setup tasks and is WEBADM by default.

TSRP Started Task
This allows Tivoli Information Management for z/OS to submit requests to the TEXT Search Engine.The TSRP must run with the Text Search Engine user ID.

File Owner
The owner of Text Search Engine should have write access to the same directories as listed in the Tivoli Information Management for z/OS text search parameters file (the INDEX_DATA_PATH and TEXT_PATH location). The owner may or may not be a Tivoli Information Management for z/OS administrator.

Authorization codes
The authorization codes used to enter, update, or delete solution records can be found in the *Tivoli Information Management for z/OS Panel Modification Facility Guide*.

Creating an Index

Before you can search the freeform text in Tivoli Information Management for z/OS records, you must create a text index for each type of record that will be searched using the Text Search Engine. The following is a basic outline of the steps involved in creating an index.

1. Specify the index that you want to create.
2. Index the records:
   a. Retrieve the records that should be indexed and create index documents.
   b. Schedule the documents for indexing.
   c. Index the scheduled documents.

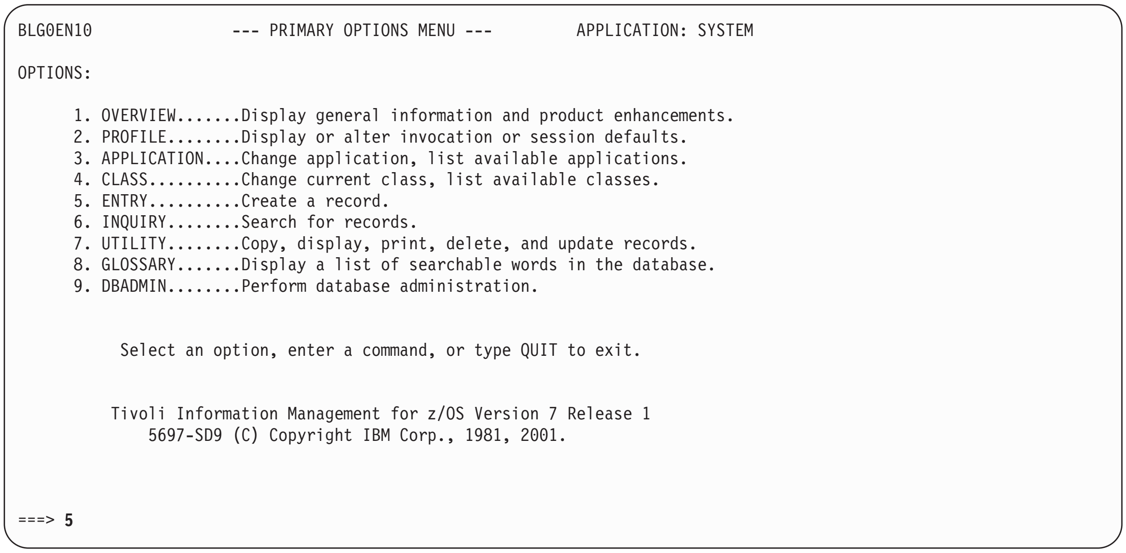
After the index is complete, it can be used to search the documents.

A batch job, BLMNINDX, is provided with Tivoli Information Management for z/OS in the SBLMSAMP library to build and update the indexes.

Specifying the Index

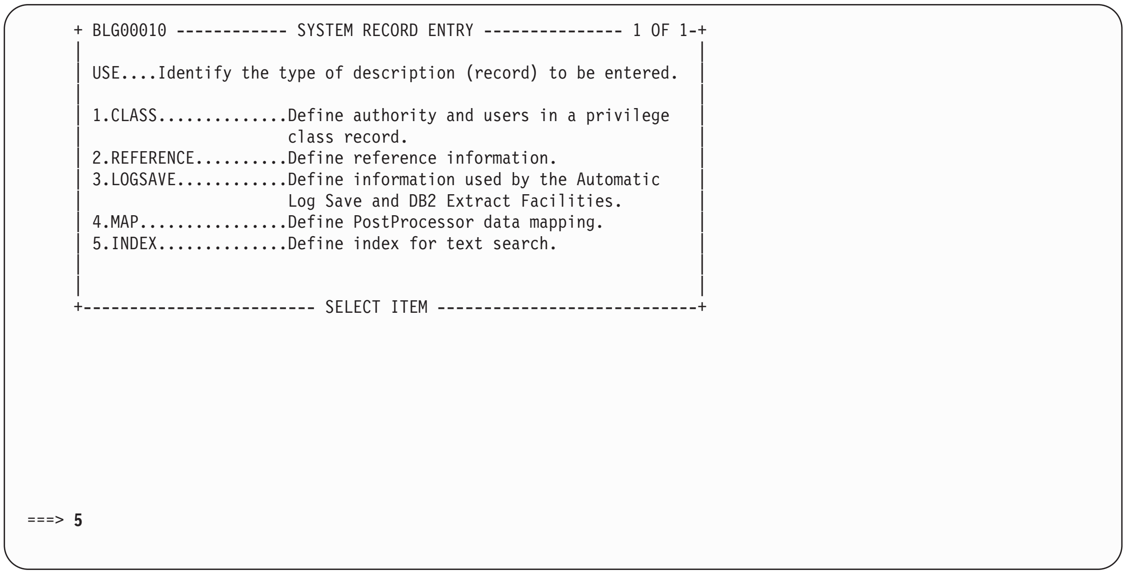
To specify an index that you want to create, you must be logged on to the SYSTEM application of Tivoli Information Management for z/OS. The Text Search Request Processor (TSRP) started task must be running in order to create an index.

On BLG0EN10, the Primary Options Menu panel, select option **5. Entry** to create a record.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
              5697-SD9 (C) Copyright IBM Corp., 1981, 2001.



===> 5
```

On BLG00010, System Record Entry, type **5** for INDEX.



```
      + BLG00010 ------------ SYSTEM RECORD ENTRY --------------- 1 OF 1-+
      |                                                                   |
      | USE....Identify the type of description (record) to be entered.   |
      |                                                                   |
      | 1.CLASS.............Define authority and users in a privilege      |
      |                     class record.                                 |
      | 2.REFERENCE.........Define reference information.                  |
      | 3.LOGSAVE...........Define information used by the Automatic      |
      |                     Log Save and DB2 Extract Facilities.          |
      | 4.MAP...............Define PostProcessor data mapping.            |
      | 5.INDEX.............Define index for text search.                 |
      |                                                                   |
      |                                                                   |
      +------------------------ SELECT ITEM -----------------------------+









 ===> 5
```

On BLH0N000, the Index Entry panel, select the specific index to create from the list and press Enter.

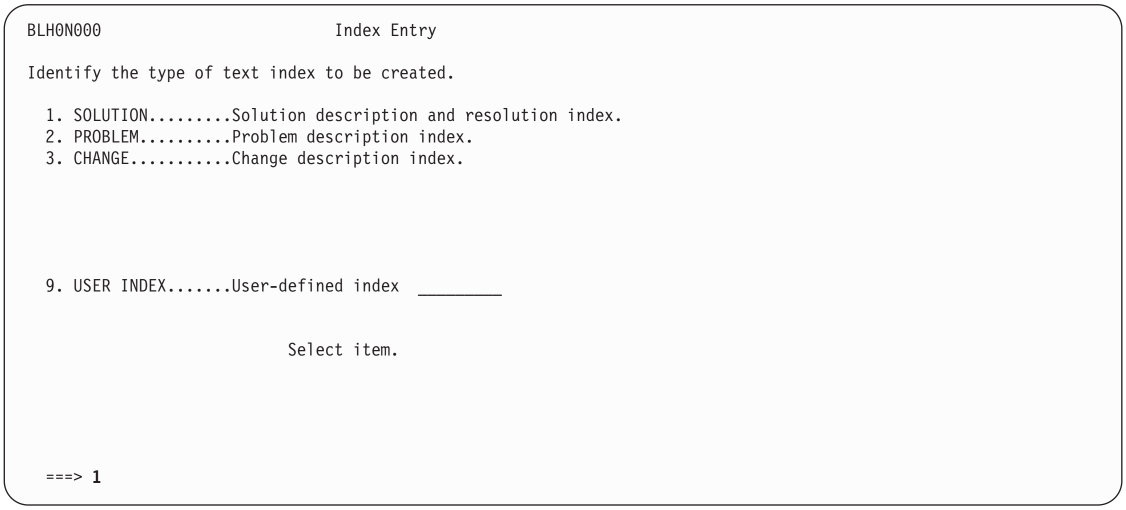
SOLUTION
Index for solution record description and resolution freeform text.

PROBLEM
Index for problem record description freeform text.

CHANGE
Index for change record description and resolution freeform text.

If you are creating an index for a record type not listed here (such as a user-defined record type), follow the procedures described in “Defining Your Own Index” on page 172.

In this example, we will request the solution index. Type **1** and press Enter.



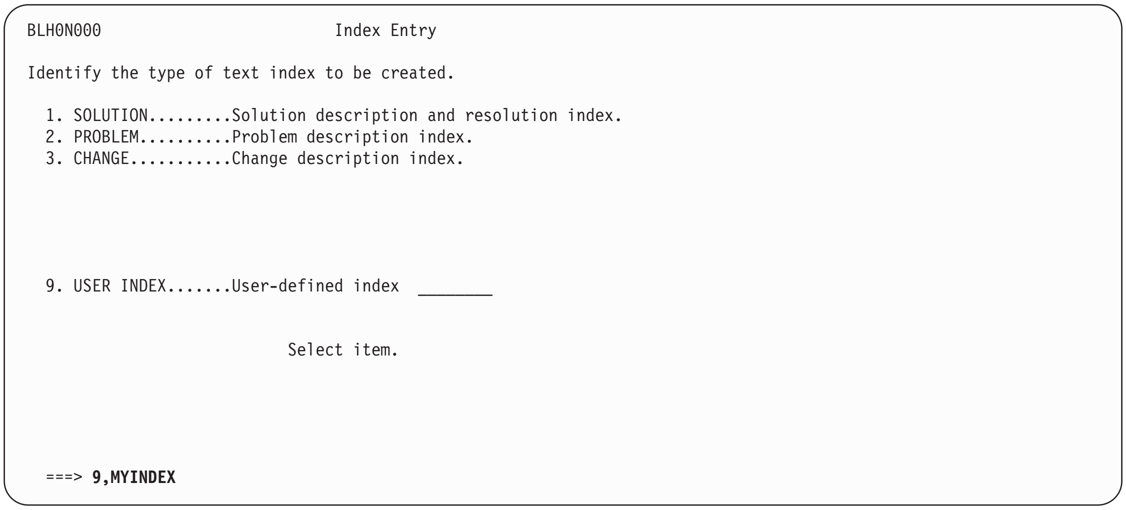
```
BLH0N000                         Index Entry

Identify the type of text index to be created.

  1. SOLUTION.........Solution description and resolution index.
  2. PROBLEM..........Problem description index.
  3. CHANGE...........Change description index.





  9. USER INDEX.......User-defined index  _________


                            Select item.





  ===> 1
```

After entering a selection, a request is sent to the Text Search Engine to create the specified index. After the index is created successfully, a corresponding index record is created in the Tivoli Information Management for z/OS database and a message is displayed to indicate the record was created.

If the text search request processor is not started or is unavailable, a warning message is displayed when you try to create the record.

Defining Your Own Index

If you want to define your own index rather than use one of the predefined index offerings provided with Tivoli Information Management for z/OS, type **9** on BLH0N000, the Index Entry panel, and specify a 1- to 8-character name for your index. The name will be used as the record ID for the index record in the Tivoli Information Management for z/OS database. Press Enter to display BLH0N001, the Index Record Entry panel.

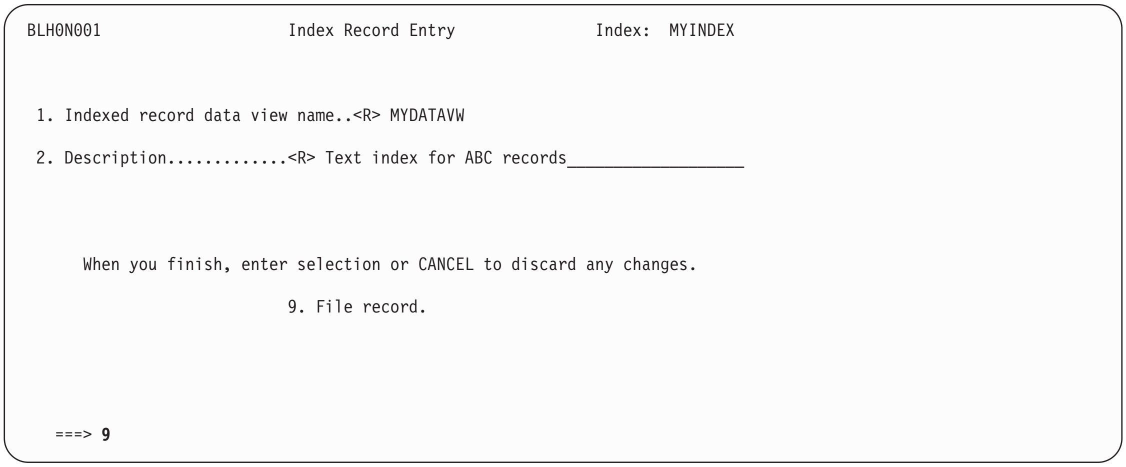


```
BLH0N000                         Index Entry

Identify the type of text index to be created.

  1. SOLUTION.........Solution description and resolution index.
  2. PROBLEM..........Problem description index.
  3. CHANGE...........Change description index.





  9. USER INDEX.......User-defined index  ________


                            Select item.





  ===> 9,MYINDEX
```

On BLH0N001, the Index Record Entry panel, enter the name of the data view record; in that data view you identify which freeform text you want to be able to index. The name can be 1 to 8 characters but the first character must be alphabetic.

Note: It is not necessary for the data view record to exist when you file the record on this panel; however, you will receive an error later when you actually start the indexing process if the data view record does not exist in the database. Therefore, it is best to set up a data view record for your index before entering a user-defined index record.

Enter a description of the index you are creating (up to 45 characters). Type **9** and press Enter to request the creation of the index by the Text Search Engine and to file a corresponding index record in the Tivoli Information Management for z/OS database.



```
BLH0N001                  Index Record Entry             Index:  MYINDEX


 1. Indexed record data view name..<R> MYDATAVW

 2. Description............<R> Text index for ABC records___________________



      When you finish, enter selection or CANCEL to discard any changes.

                          9. File record.




   ===> 9
```

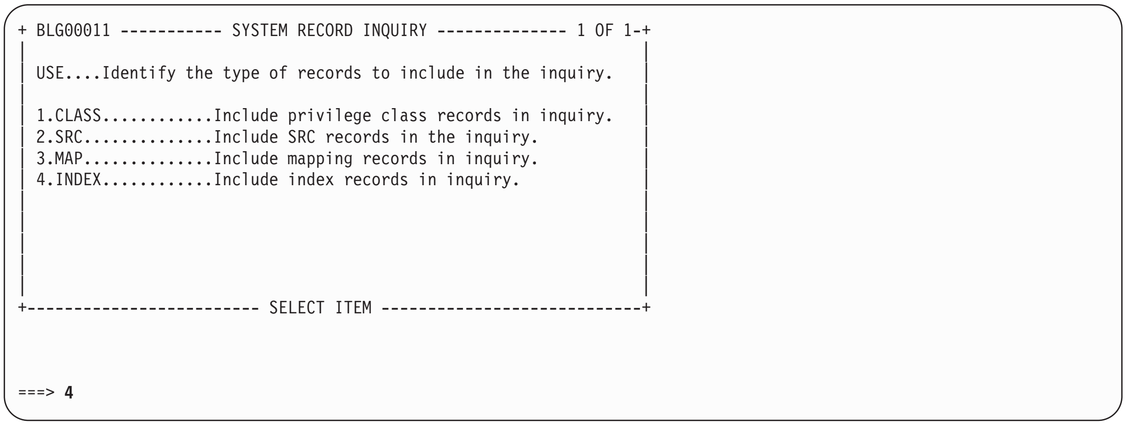
If the text search request processor is not started or is unavailable, a warning message is displayed after you try to file the record.

Setting Up Data View Records

The data view records provided with Tivoli Information Management for z/OS to support the use of OS/390 Text Search are listed in Table 33 on page 191. If you have different types of records and would like to be able to perform freeform text searches against those records, you can create your own data view records (and data attribute records) and indexes. That is, the data view record identifies the types of freeform text that will be indexed. If you want to search other types of freeform text or search freeform text in other types of records, you may need to modify the data model records that are provided or else create your own. If you do create your own data view records, you also need to create the index. Use the data view and data attribute records provided as a model to construct your own data model records.

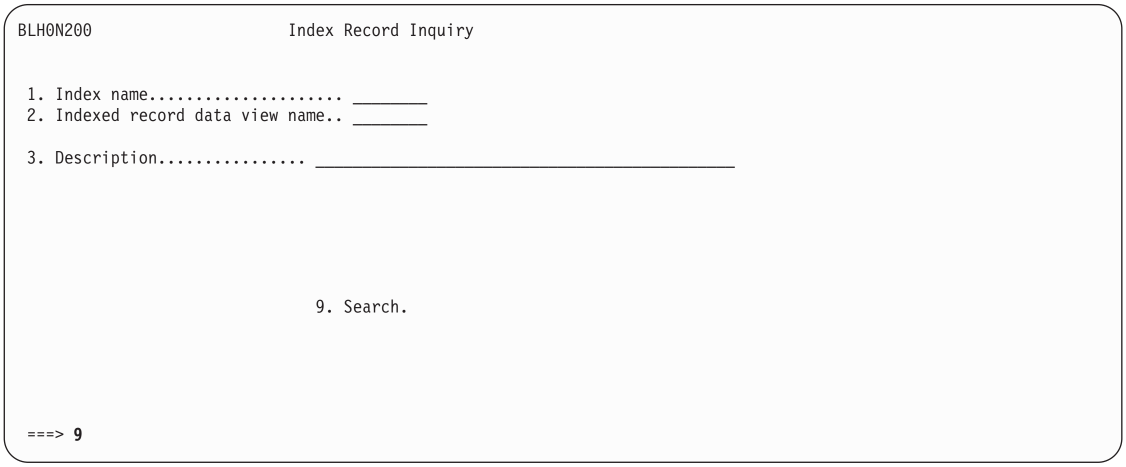
Searching for Index Records

To search for index records in the Tivoli Information Management for z/OS database, select option 6 on BLG0EN10, the Primary Options Menu for the System application. Then, select **4. INDEX** on BLG00011, the System Record Inquiry panel.



```
+ BLG00011 ----------- SYSTEM RECORD INQUIRY -------------- 1 OF 1-+
|                                                                  |
| USE....Identify the type of records to include in the inquiry.   |
|                                                                  |
| 1.CLASS............Include privilege class records in inquiry.   |
| 2.SRC..............Include SRC records in the inquiry.           |
| 3.MAP..............Include mapping records in inquiry.           |
| 4.INDEX............Include index records in inquiry.             |
|                                                                  |
|                                                                  |
|                                                                  |
|                                                                  |
|                                                                  |
+------------------------- SELECT ITEM ----------------------------+



===> 4
```

On BLH0N200, provide search data, type **9**, and press Enter to begin the search. You can, for example, search for the index record by specifying its data view record name or by the index record description. The search performs a database search using the search criteria that you provided.



```
BLH0N200                    Index Record Inquiry


 1. Index name.................... _______
 2. Indexed record data view name.. _______

 3. Description............... ________________________________________








                               9. Search.







 ===> 9
```

Displaying Index Records

To display an existing index record, use the DISPLAY command as you would any record in the Tivoli Information Management for z/OS database (for example, DISPLAY R MYINDEX). The record ID is the index name.

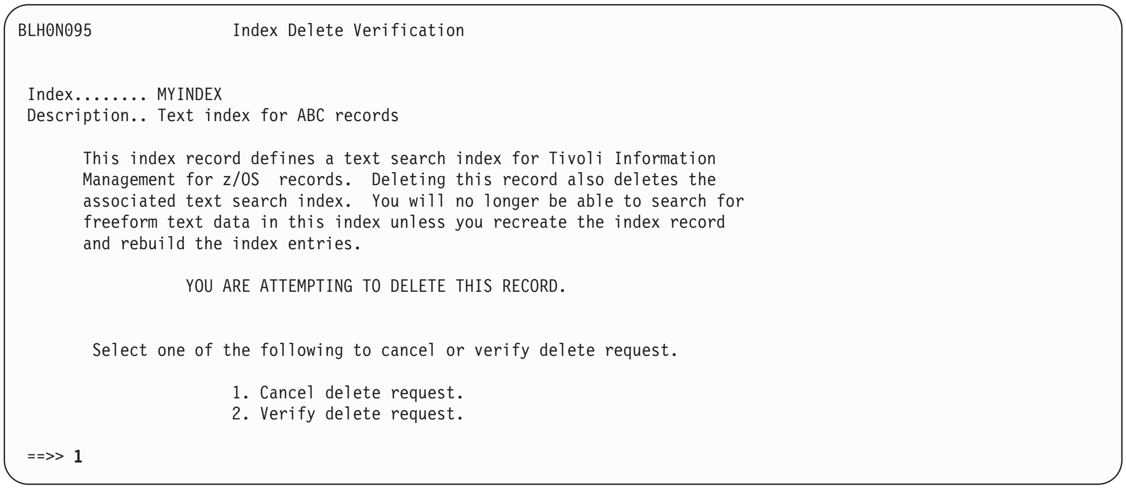
Deleting Index Records

You can use the DELETE command to delete an index record in the Tivoli Information Management for z/OS database. A delete verification panel is displayed to confirm your deletion request. If you delete the index record from the Tivoli Information Management for z/OS database, you will also delete the corresponding text search index from its directory location on OS/390 UNIX System Services.

Note: Deleting an index does not remove the Index flag (INDX/) from the associated Tivoli Information Management for z/OS records.

After an index record is deleted, users will no longer be able to search for freeform text data in the index.

Type **1** and press Enter to delete the index record, or select **2** to cancel the request.



```
BLH0N095              Index Delete Verification


 Index........ MYINDEX
 Description.. Text index for ABC records

       This index record defines a text search index for Tivoli Information
       Management for z/OS  records.  Deleting this record also deletes the
       associated text search index.  You will no longer be able to search for
       freeform text data in this index unless you recreate the index record
       and rebuild the index entries.

                  YOU ARE ATTEMPTING TO DELETE THIS RECORD.


        Select one of the following to cancel or verify delete request.

                      1. Cancel delete request.
                      2. Verify delete request.

 ==>> 1
```

You can delete only one index record at a time.

Indexing the Records

To extract the freeform text data from the Tivoli Information Management for z/OS database and create index documents for the OS/390 Text Search index, run the BLMNINDX batch job provided with Tivoli Information Management for z/OS (see member BLMNINDJ in the SBLMSAMP library). You can create or update all Tivoli Information Management for z/OS indexes or a specific index. The creation of indexes is done through Tivoli Information Management for z/OS panels, as previously described. The BLMNINDX job by default updates all indexes having index records in the Tivoli Information Management for z/OS database.

You should run the BLMNINDX batch job on a regular schedule according to your needs. For example, you may find it sufficient to run the indexing job once daily during off-peak hours. If you find that users require more current updates during the day, you should run the batch job more frequently. You can refresh an index at any time. For optimum performance, however, it is best to avoid periods of peak activity. During a refresh of the index, the current version is always available for searching by users. When the update is complete, the index is automatically committed for searching.

The BLMNINDX job breaks down the index process into three steps:

1. It searches for and retrieves the records from the Tivoli Information Management for z/OS database that need to be indexed.

   In this step, the records associated with each index record are searched and retrieved. By default, records that have not already been indexed (or records that have been updated and need to be reindexed) are retrieved. As each record is retrieved from the Tivoli Information Management for z/OS database, the freeform text for that record is written to a temporary freeform text file in an OS/390 UNIX System Services directory. (For example, for a change record, both the description and resolution freeform text is written to the text file.) The record is flagged as having been indexed. The following field is added to a record when its freeform text is indexed:

   Text index name
       Prefix = INDX/
       s-word index = S12E3
       Data attribute record = BLH&INDX

Note: If the freeform text associated with an indexed record is updated in the Tivoli Information Management for z/OS at a later point, the index flag is removed from that record so that the updated information can be retrieved when the BLMNINDX job runs again. In order to properly control what is updated, you should note the parameters that are available in "Controlling the BLMNINDX API Sessions" on page 164.

The location of the temporary freeform text file in OS/390 UNIX System Services is /*path*/*index*/*xxxxxxxx*.txt where:

- *xxxxxxxx* is the VSAM key of the record.
- *index* is the text index name associated with the record (for example, INDXPROB, INDXCHNG, INDXSOLN).
- *path* is the TEXT_PATH parameter value defined in the BLMTSRPP parameters member.

The BLMNINDX job also produces two other files for each index:

- records.list – This file contains a list of the *xxxxxxxx*.txt files that will be scheduled for indexing by the Text Search Engine. The location of this file is /path/index/records.list.
- rnid.list – This file contains a list of the record IDs that were retrieved from the Tivoli Information Management for z/OS database. This file supports the RESTART option so the BLMNINDX job can identify which records need to have the index flag reset. The location of this file is /path/index/rnid.list.

2. It schedules the documents for indexing.

   A "document" is a file in the OS/390 UNIX System Services directory which contains the freeform text from a single Tivoli Information Management for z/OS record.

3. It indexes the documents.

   This step also causes all the temporary freeform text files generated in the first step to be deleted.

You can run all three steps, or specify which step to start if you do not want to run all three steps. For example, if a problem occurs, you can re-run BLMNINDX to perform only the last step or only the last two steps by specifying the appropriate parameter. If you are updating the index, you should run the job with all three steps.

You can specify the following options when running BLMNINDX:

START
Process all three steps to update the index; that is, if you specify START, you will also trigger the SCHEDULE and INDEX steps. The START option is the default.

SCHEDULE
The Tivoli Information Management for z/OS database records have already been retrieved. Schedule the documents for indexing and update the index; that is, if you specify SCHEDULE, you are also triggering the INDEX step.

INDEX
Documents have been scheduled for indexing. Update the index.

RESTART
Start from the beginning. Change the index flag to indicate records are not indexed (even if records were retrieved), erase the previously written files, and proceed from the beginning with starting, scheduling, and indexing the documents. That is, if you specify RESTART, you are also triggering the START, SCHEDULE, and INDEX steps.

This sample is provided as member BLMNINDJ in the SBLMSAMP sample library.

```
//INDEX    JOB (account)
//*
//* This is a sample jobstream for running BLMNINDX to index
//* the freeform text data for each index.
//*
//* You must update the parm field as appropriate and also change the
//* data set names for DD names STEPLIB, VARS, SYSPRINT, APIPRINT,
//* HLAPILOG to correspond to the data set names at your
//* installation.
//*
//* For STEPLIB DD you must include the BLX-SP server and session-
//* parameters member data sets.
//*
//* PARM field valid settings are:
//*   /START     Process all indexing steps, this is the default.
//*
//*   The following settings are used when a failure occurs during
//*   the previous run of this job. See the messages issued to
//*   determine which steps did not complete successfully.
//*
//*   /SCHEDULE  Begin job with schedule step.
//*   /INDEX     Begin job with index step.
//*   /RESTART   Redo entire job from the beginning.
//*
//*-----------------------------------------------------------------
//* INDEX THE FREEFORM TEXT DATA FOR EACH INDEX IN THE DATABASE
//*-----------------------------------------------------------------
//INDXPGM   EXEC PGM=BLMNINDX,
//          PARM='ENVAR(_CEE_ENVFILE=DD:VARS)/START'
//STEPLIB   DD  DISP=SHR,DSNAME=BLM.SBLMMOD1
//          DD  DISP=SHR,DSN=CEE.SCEERUN
//          DD  DISP=SHR,DSN=CBC.SCBCCMP
//VARS      DD  DISP=SHR,DSN=BLM.SBLMSAMP(BLMTSRPP)
//BLGTSX    DD  DISP=SHR,DSN=BLM.SBLMTSX
//SYSPRINT  DD SYSOUT=*
//APIPRINT  DD SYSOUT=*
//HLAPILOG  DD SYSOUT=*
//SYSIN     DD DUMMY
//***************************************************
```

`_CEE_ENVFILE=DD:VARS` indicates that the DD statement VARS is used to point to the text search request processor parameters file. This must be either a fixed block PDS or fixed block sequential file. If the text search request processor parameters (in effect, the environment variables) are in a PDS, you can override the parameter values by specifying the following:

```
//        DD   *
```

After this line, list the parameters you wish to override and include the desired values. If you use an OS/390 UNIX System Services file to specify parameter values, you cannot override parameters in this manner. Although you can specify an OS/390 UNIX System Services file or a variable blocked PDS file on the VARS DD statement, anything you specify after `DD *` is ignored.

The DD statements required to run BLMNINDX are as follows. Use the correct data set names for your installation:

STEPLIB
Identifies the BLX-SP server and session-parameters member data sets.

VARS Specifies the location of the environment variables file.

SYSPRINT
A sequential data set (RECFM=VBA) containing summary and error messages that you can write to a system output device or data set.

APIPRINT
A sequential data set (RECFM=VBA) containing messages about the Tivoli Information Management for z/OS API subtask that you can write to a system output device or data set.

HLAPILOG
A sequential data set (RECFM=VBA) containing transaction activity messages created by the Tivoli Information Management for z/OS high-level API that you can write to a system output device or data set.

Scenario – Running All Three Index Steps

In this example, the BLMNINDX job is set up to begin with the START option (see the first line) to run all three index steps:

```
//INDXPGM  EXEC PGM=BLMNINDX,PARM='ENVAR(_CEE_ENVFILE=DD:VARS)/START'
//STEPLIB  DD   DISP=SHR,DSN=BLM.SBLMMOD1
//         DD   DISP=SHR,DSN=CEE.SCEERUN
//         DD   DISP=SHR,DSN=CBC.SCBCCMP
//VARS     DD   DISP=SHR,DSN=BLM.SBLMSAMP(BLMTSRPP)
//         DD   *
TEXT_PATH=/u/jsmith/inline/test
//BLGTSX   DD   DISP=SHR,DSN=BLM.SBLMTSX
//SYSPRINT DD SYSOUT=*
//APIPRINT DD SYSOUT=*
//HLAPILOG DD SYSOUT=*
//SYSIN    DD DUMMY
```

The path specified in `TEXT_PATH` overrides the text path value set in the text search request processor parameters file. In this example, the temporary freeform text files will be written to the `/u/jsmith/inline/test` directory in OS/390 UNIX System Services.

Scenario – Scheduling the Index Update without Record Retrieval

In this example, records have already been retrieved. The BLMNINDX job will schedule the documents for indexing. To perform the INDEX step, you must run BLMNINDX again and specify the INDEX option to update the index. As was described on page 176, specifying INDEX causes only the index to be updated.

```
//INDXPGM  EXEC PGM=BLMNINDX,PARM='ENVAR(_CEE_ENVFILE=DD:VARS)/SCHEDULE'
//STEPLIB  DD   DISP=SHR,DSN=BLM.SBLMMOD1
//         DD   DISP=SHR,DSN=CEE.SCEERUN
//         DD   DISP=SHR,DSN=CBC.SCBCCMP
//VARS     DD   DISP=SHR,DSN=BLM.SBLMSAMP(BLMTSRPP)
//         DD   *
TEXT_PATH=/u/jsmith/inline/test
//BLGTSX   DD   DISP=SHR,DSN=BLM.SBLMTSX
//SYSPRINT DD SYSOUT=*
//APIPRINT DD SYSOUT=*
//HLAPILOG DD SYSOUT=*
//SYSIN    DD DUMMY
```

Maintaining the Index

As administrator for Tivoli Information Management for z/OS, you may need to perform index maintenance tasks periodically. To keep your index current, you need to carry out index updates regularly by running the BLMNINDX batch job. How often you do this depends on how quickly documents are added to and updated in your Tivoli Information Management for z/OS database.

After several index updates, it is recommended that you reorganize the index to optimize the space it uses and to improve the search time for queries against it. A reorganization can also remove obsolete information from the index. To reorganize an index, the Text Search Engine administrator can use a Text Search Index command (IMOREOIX) as described in *OS/390 Text Search: Installation and Administration for the Text Search Engine*. The Text Search Engine will also automatically reorganize an index based on the value set for UPDATETHRESHOLD in the imosrv.ini file.

For example:

```
imoreoix -s IMSRCH -x INDXSOLN
```

In this example, IMSRCH is the search service name. This is the default value set by Tivoli Information Management for z/OS. INDXSOLN is the name of the index that will be reorganized.

Additionally, you may need to delete an index if it no longer serves a useful purpose. When you delete an index, the following are also removed:

- the Text Search index in OS/390 UNIX System Services
- the index record in Tivoli Information Management for z/OS

The index flag in associated Tivoli Information Management for z/OS records (prefix word INDX/) is *not* removed from these records when an index is deleted.

You should *not* use the Text Search Engine IMODELIX command natively to delete a Tivoli Information Management for z/OS index. If you do, the Tivoli Information Management for z/OS index record and Text Search Engine index will not be synchronized. Instead, you should use the Tivoli Information Management for z/OS panel interface to delete text search indexes.

For instructions on how to delete an index using Tivoli Information Management for z/OS panels, see “Deleting Index Records” on page 174.

The BLMNMNT Utility

BLMNMNT provides a way to perform Text Search Engine (TSE) maintenance and status functions through a batch job running on MVS. All of the functions performed by BLMNMNT have a corresponding TSE command that can be issued while in OS/390 UNIX System Services and a corresponding TSE API function call.

The purpose of BLMNMNT is to allow you to perform certain Text Search Engine (TSE) functions without having to go into OS/390 UNIX System Services to issue the TSE commands. You can also use BLMNMNT in conjunction with BLMNINDX to obtain function status, index status, display index messages, or any of the other functions before and after BLMNINDX runs. For example, you can run BLMNMNT before you run BLMNINDX in order to see how many documents are indexed and then run it again after you run BLMNINDX to see how many new documents were added. You can also display the number of documents contained in both the primary and secondary indexes after BLMNINDX runs to decide if the index needs to be reorganized. If it is determined that the index should be reorganized, you can then run BLMNMNT again to perform that function.

These are the parameters that can be passed to BLMNMNT:

Input parameter
Purpose

No input parameter
Gives index status, such as

- Number of scheduled documents.
- Number of documents in the primary index.
- Number of documents in the secondary index.
- Number of indexing messages.

CLEAR
Removes all entries from a TSE index. This specification should be used with care.

CLRDOCS
Removes scheduled documents from the TSE index queue.

DELMSGS
Prints TSE indexing messages and then deletes them.

FUNCSTAT
Gives status of each function for an index. Each index has four functions associated with it:

- Search
- Schedule Documents
- Index documents, updates an index
- Merge, reorganizes an index

This will tell if an index is enabled or disabled, and when the last time each function ran, whether it is currently executing, or whether it has been stopped due to an error.

INDXINFO
Give characteristics of each index. The characteristics that are displayed are:

- Type of TSE index (should be NGRAM)
- Coded character set identifier (CCSID) of an index
- Client library service DLL (should be IMOLSCFS)
- Server library service DLL (should be BLMLSSFS)
- Index directory
- Index work directory

INDXMSGS
Prints TSE indexing messages, but does not delete them.

REORG
Reorganizes and merges an index. Merges primary and secondary indexes.

RESUME
Resumes an index after it has been suspended.

SUSPEND
Suspends an index so that no functions can be performed against it.

UPDATE
Updates an index, indexes any documents that are currently scheduled for an index.

The basic processing of BLMNMNT is to first perform a EhwListIndexes transaction to retrieve all of the TSE indexes. The requested action is then performed against each index. By using the INDEX= environment variable processing can be limited to one index; this is especially important if you are performing an action like clearing an index of all of its entries.

This table shows the relationship between the input parameter, the TSE API call, and the TSE line command.

Table 32. Relationship between input parameters, TSE API, and line commands.

Input Parameter	TSE API Call	Line command example
no parm	EhwGetIndexStatus	imostaix -s IMSRCH -x INDXSOLN
CLEAR	EhwClearIndex	imoclrix -s IMSRCH -x INDXSOLN
CLRDOCS	EhwClearScheduledDocuments	imoqueue -s IMSRCH -x INDXSOLN -clear
DELMSGS	EhwGetIndexingMsgs	imomsgix -s IMSRCH -x INDXSOLN -delete
FUNCSTAT	EhwGetIndexFunctionStatus	imostfix -s IMSRCH -x INDXSOLN
INDXINFO	EhwGetIndexInfo	imostaix -s IMSRCH -x INDXSOLN
INDXMSGS	EhwGetIndexingMsgs	imomsgix -s IMSRCH -x INDXSOLN
REORG	EhwReorgIndex	imoreoix -s IMSRCH -x INDXSOLN
RESUME	EhwResumeIndex	imoctrix -s IMSRCH -x INDXSOLN -resume
SUSPEND	EhwSuspendIndex	imoctrix -s IMSRCH -x INDXSOLN -suspend
UPDATE	EhwUpdateIndex	imoupdix -s IMSRCH -x INDXSOLN

The following example shows BLMNMNT running with the FUNCSTAT keyword as the input parameter. The path name preceding the slash (/) tells BLMNMNT where to find its environment variables file. In this case it is using the data set specified on the VARS DD statement. Note that an OS/390 UNIX System Services file could also be used as the input file if desired. This sample is provided as member BLMNMNTJ in the SBLMSAMP sample library.

```
//GO       EXEC PGM=BLMNMNT,PARM='ENVAR(_CEE_ENVFILE=DD:VARS)/FUNCSTAT'
//STEPLIB  DD   DISP=SHR,DSN=BLM.SBLMMOD1
//         DD   DISP=SHR,DSN=CEE.SCEERUN
//         DD   DISP=SHR,DSN=CBC.SCBCCMP
//VARS     DD   DISP=SHR,DSN=BLM.SBLMSAMP(BLMTSRPP)
//SYSPRINT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD DUMMY
//SYSIN    DD DUMMY
```

Creating Solution Records for a Knowledge Base

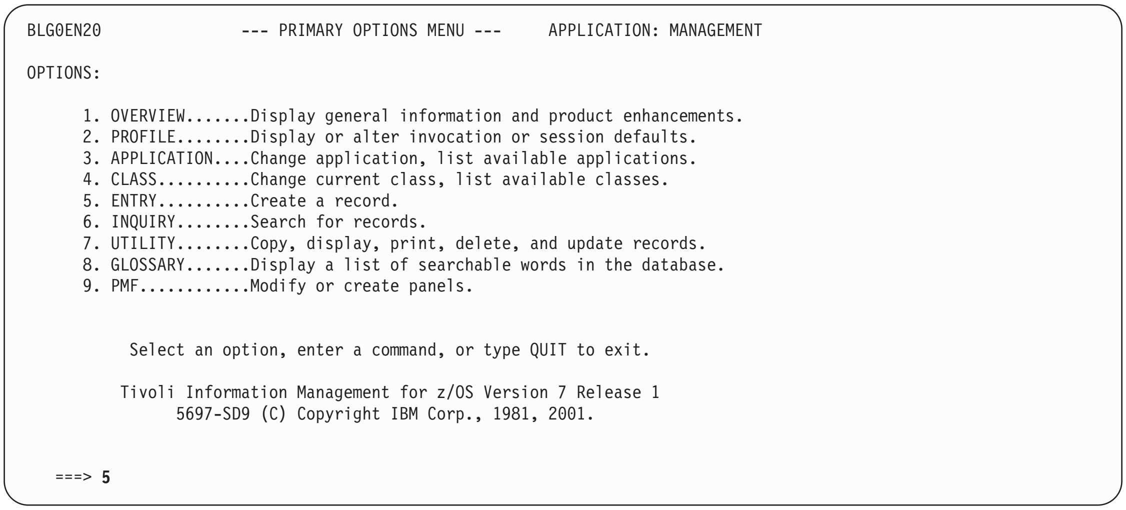
If you want to set up a knowledge base of solution records that your help desk can use to resolve problems, you must create solution records in the Tivoli Information Management for z/OS database and a solution index to enable searches to be performed against the solution data.

To create solution records, you must be in the Tivoli Information Management for z/OS Management application. From the host 3270 interface, you can create solution records either by:

- Entering a solution record directly through the record entry panel.
- Filing an existing problem record (which is in a closed status) and requesting the creation of a solution record.

Creating a Solution Record Using Data Entry Panels

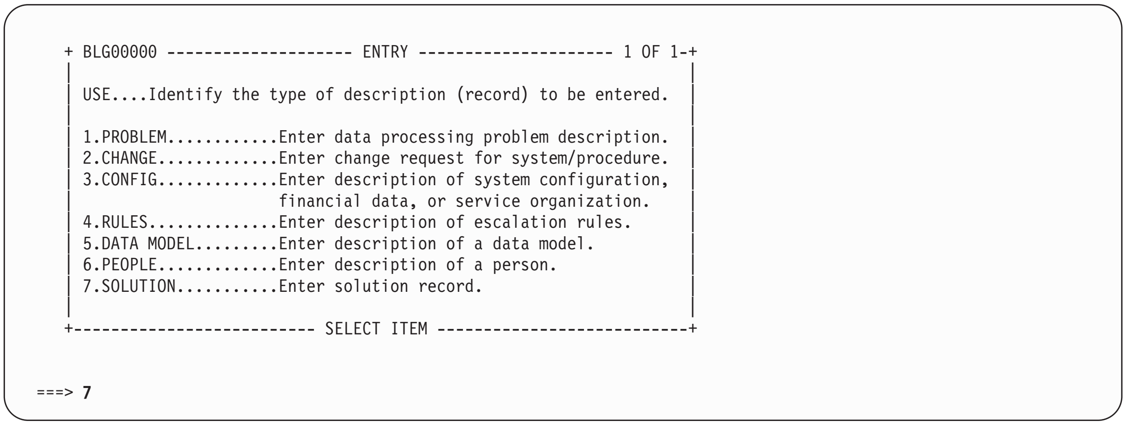
On panel BLG0EN20, the Primary Options Menu, select **5. Entry** and press Enter.



```
BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.


           Select an option, enter a command, or type QUIT to exit.

          Tivoli Information Management for z/OS Version 7 Release 1
                5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


   ===> 5
```

Select **7. Solution** and press Enter.

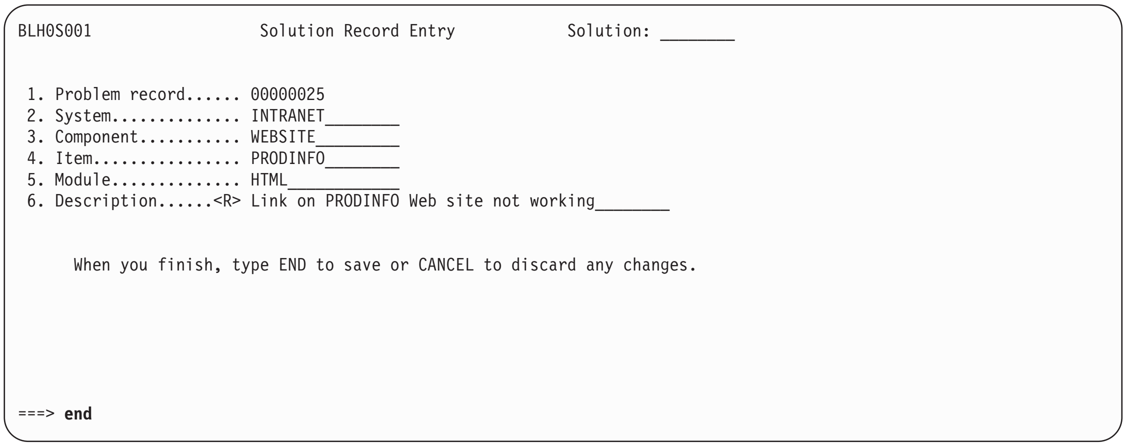
Note: Users must have solution authority to use this option.



```
    + BLG00000 -------------------- ENTRY -------------------- 1 OF 1-+
    |                                                                 |
    | USE....Identify the type of description (record) to be entered. |
    |                                                                 |
    | 1.PROBLEM...........Enter data processing problem description.  |
    | 2.CHANGE............Enter change request for system/procedure.  |
    | 3.CONFIG............Enter description of system configuration,  |
    |                     financial data, or service organization.    |
    | 4.RULES.............Enter description of escalation rules.      |
    | 5.DATA MODEL........Enter description of a data model.          |
    | 6.PEOPLE............Enter description of a person.              |
    | 7.SOLUTION..........Enter solution record.                      |
    |                                                                 |
    +------------------------- SELECT ITEM ---------------------------+


 ===> 7
```

On the Solution Record Entry panel, enter the information for the solution. You can enter the record ID of a closed problem to create a solution from that problem or you can provide your own information. If you enter your own information, do not enter a record ID in the

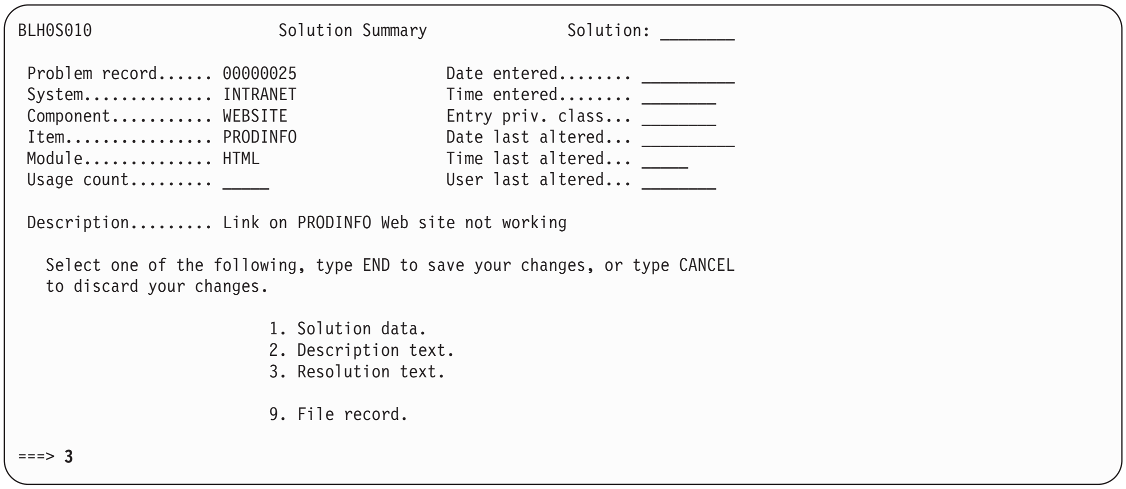
Problem Record field. The following example shows how to create a solution record from a closed problem record. In addition to being CLOSED, the problem must contain description and resolution freeform text; otherwise, the request to create the solution from the problem will be rejected. When you enter a problem record ID in the Problem record field, information is copied from the problem record into the solution. Enter the problem record ID and press Enter. The information displayed for the system, component, item, and module is copied from the problem to the solution. Description and resolution freeform text are also copied to the solution.



```
BLH0S001                 Solution Record Entry           Solution: _______


 1. Problem record...... 00000025
 2. System.............. INTRANET________
 3. Component........... WEBSITE_________
 4. Item................ PRODINFO________
 5. Module.............. HTML____________
 6. Description......<R> Link on PRODINFO Web site not working________


      When you finish, type END to save or CANCEL to discard any changes.







===> end
```

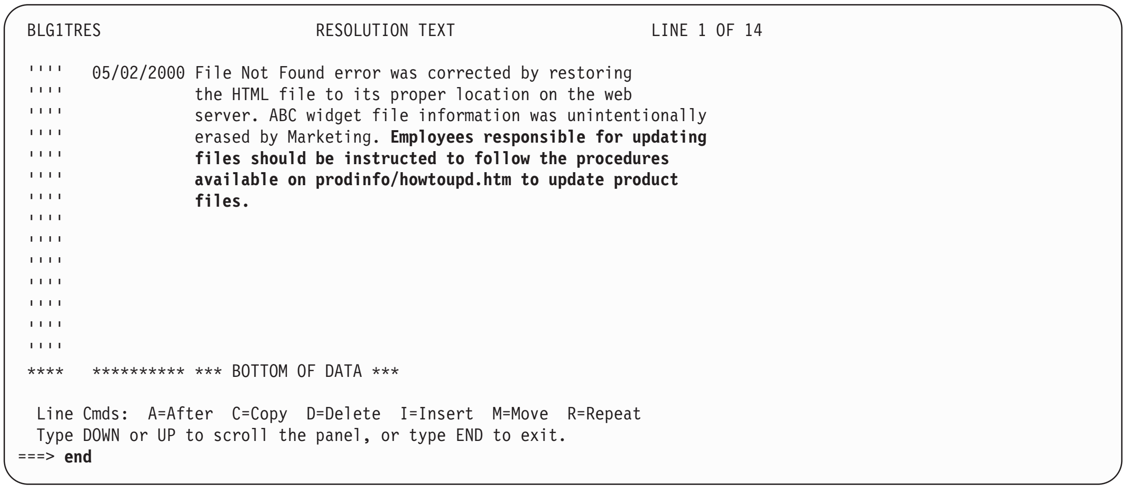
Type **end** and press Enter. The Solution Summary panel, BLH0S010, is displayed. On this panel, you can choose option **1** to change the solution data, option **2** to change the description text, or option **3** to change the resolution text. Option **9** files the record. If you are providing your own information in the solution, choose option **2** to enter your description freeform text and option **3** to enter your resolution freeform text.

For this example, type **3** and press Enter to see the Resolution text that was copied from the problem.



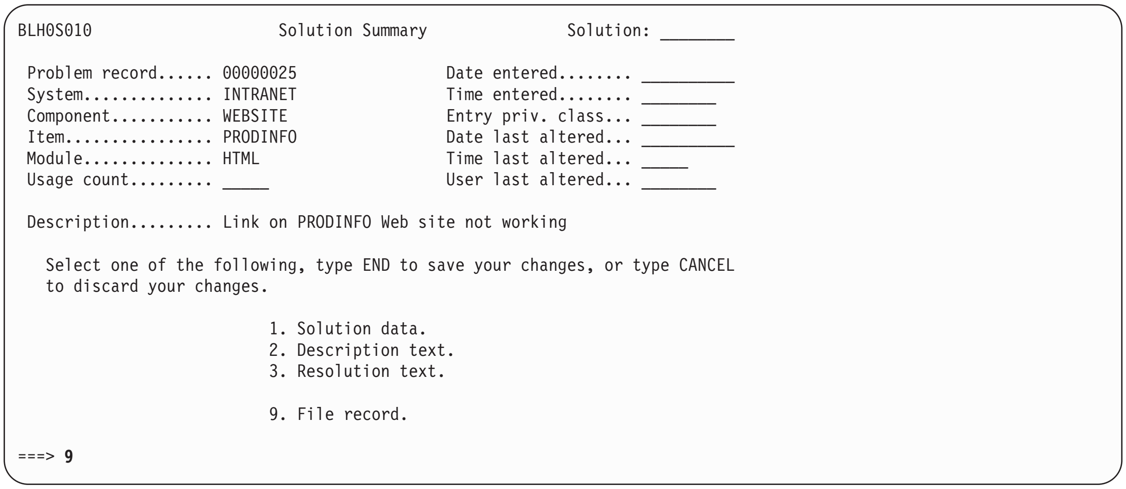
```
BLH0S010                   Solution Summary              Solution: _______

 Problem record...... 00000025             Date entered........ __________
 System.............. INTRANET             Time entered........ ________
 Component........... WEBSITE              Entry priv. class... ________
 Item................ PRODINFO             Date last altered... __________
 Module.............. HTML                 Time last altered... _____
 Usage count......... _____                User last altered... ________

 Description......... Link on PRODINFO Web site not working

   Select one of the following, type END to save your changes, or type CANCEL
   to discard your changes.

                        1. Solution data.
                        2. Description text.
                        3. Resolution text.

                        9. File record.

===> 3
```

You can update the text as needed. Then type **end** and press Enter to return to the Solution Summary panel.



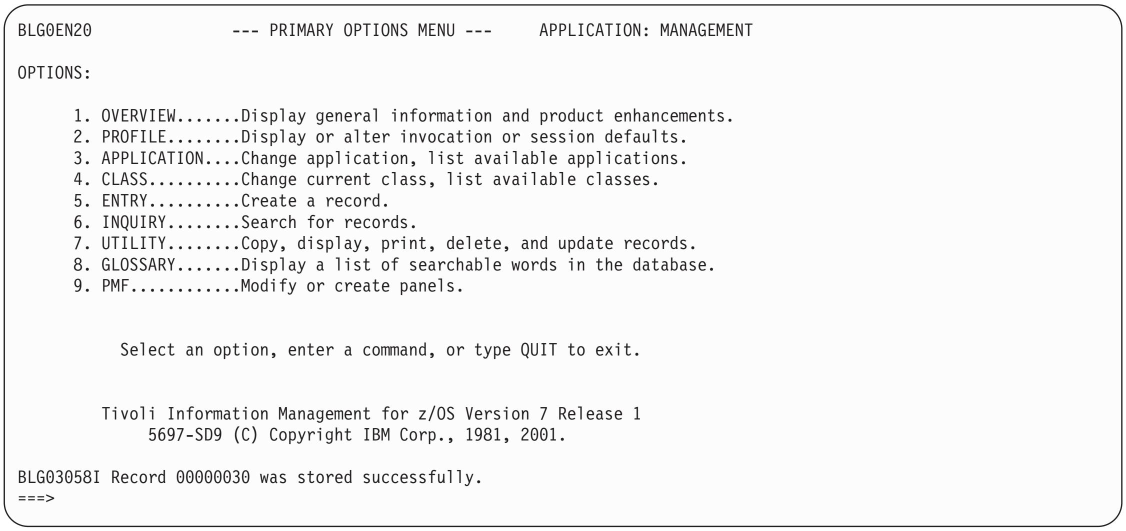
```
 BLG1TRES                      RESOLUTION TEXT                    LINE 1 OF 14

 ''''   05/02/2000 File Not Found error was corrected by restoring
 ''''              the HTML file to its proper location on the web
 ''''              server. ABC widget file information was unintentionally
 ''''              erased by Marketing. Employees responsible for updating
 ''''              files should be instructed to follow the procedures
 ''''              available on prodinfo/howtoupd.htm to update product
 ''''              files.
 ''''
 ''''
 ''''
 ''''
 ''''
 ''''
 ''''
 ****   ********** *** BOTTOM OF DATA ***

  Line Cmds:  A=After  C=Copy  D=Delete  I=Insert  M=Move  R=Repeat
  Type DOWN or UP to scroll the panel, or type END to exit.
===> end
```

Type **9** and press Enter to file the solution record.



```
BLH0S010                  Solution Summary              Solution: ________

 Problem record...... 00000025                Date entered........ __________
 System.............. INTRANET                Time entered........ ________
 Component........... WEBSITE                 Entry priv. class... ________
 Item................ PRODINFO                Date last altered... __________
 Module.............. HTML                    Time last altered... _____
 Usage count......... _____                   User last altered... ________

 Description......... Link on PRODINFO Web site not working

   Select one of the following, type END to save your changes, or type CANCEL
   to discard your changes.

                        1. Solution data.
                        2. Description text.
                        3. Resolution text.

                        9. File record.

===> 9
```

A message confirms that the solution record was stored successfully.

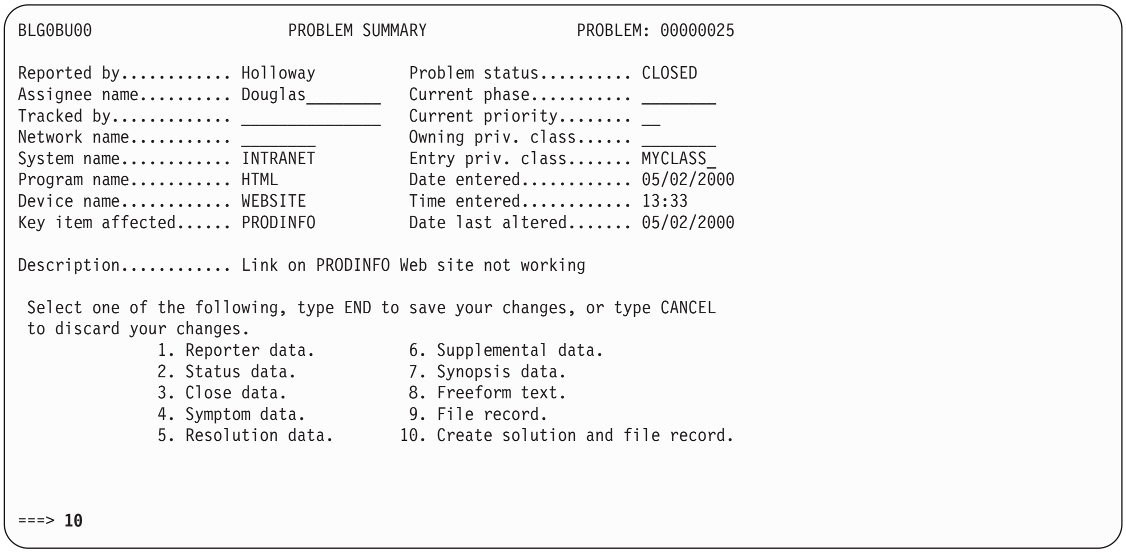


```
BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.


           Select an option, enter a command, or type QUIT to exit.


         Tivoli Information Management for z/OS Version 7 Release 1
              5697-SD9 (C) Copyright IBM Corp., 1981, 2001.

BLG03058I Record 00000030 was stored successfully.
===>
```

Creating a Solution Record When Filing a Problem Record

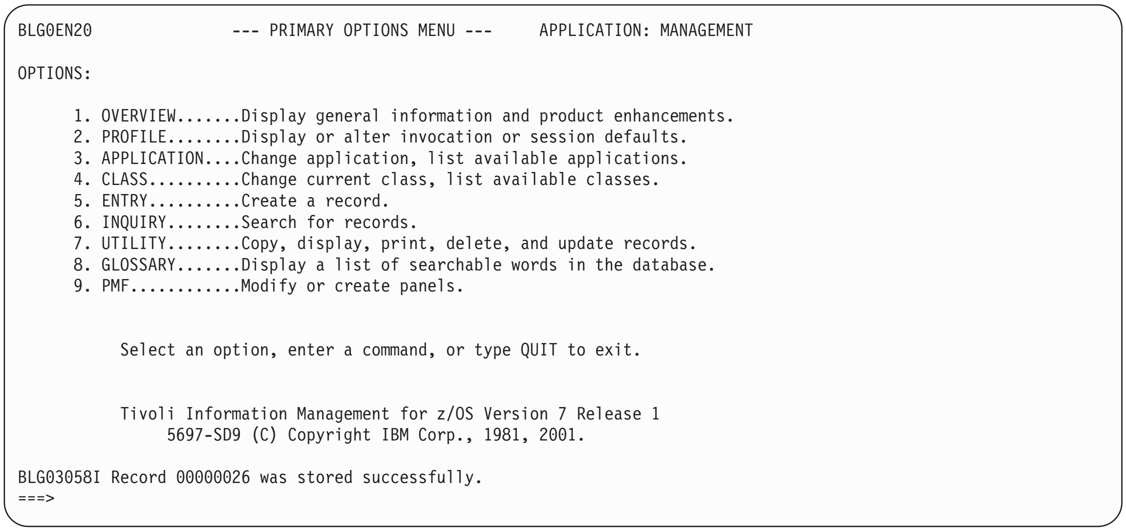
On the Problem Summary panel, BLG0BU00, select **10. Create solution and file record** and press Enter to create the solution record and file the problem record. If the status of the problem is CLOSED and the problem contains both description and resolution freeform text, the solution record is created with data that is copied from the problem record. If the problem record does not contain this freeform text or its status is not CLOSED, the request to create a solution record is rejected.

Note: Users must have solution authority to use this option. Also, the problem record must be closed and must contain both description and resolution freeform text. Otherwise, the request to create a solution record is rejected.



```
BLG0BU00                    PROBLEM SUMMARY              PROBLEM: 00000025

Reported by........... Holloway         Problem status.......... CLOSED
Assignee name......... Douglas________  Current phase........... ________
Tracked by............ _______________  Current priority........ __
Network name.......... ________         Owning priv. class...... ________
System name........... INTRANET         Entry priv. class....... MYCLASS_
Program name.......... HTML             Date entered............ 05/02/2000
Device name........... WEBSITE          Time entered............ 13:33
Key item affected..... PRODINFO         Date last altered....... 05/02/2000

Description........... Link on PRODINFO Web site not working

 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
              1. Reporter data.         6. Supplemental data.
              2. Status data.           7. Synopsis data.
              3. Close data.            8. Freeform text.
              4. Symptom data.          9. File record.
              5. Resolution data.      10. Create solution and file record.



===> 10
```

A message is displayed to indicate successful creation of the solution record. Note that the solution record is associated with a unique record ID.



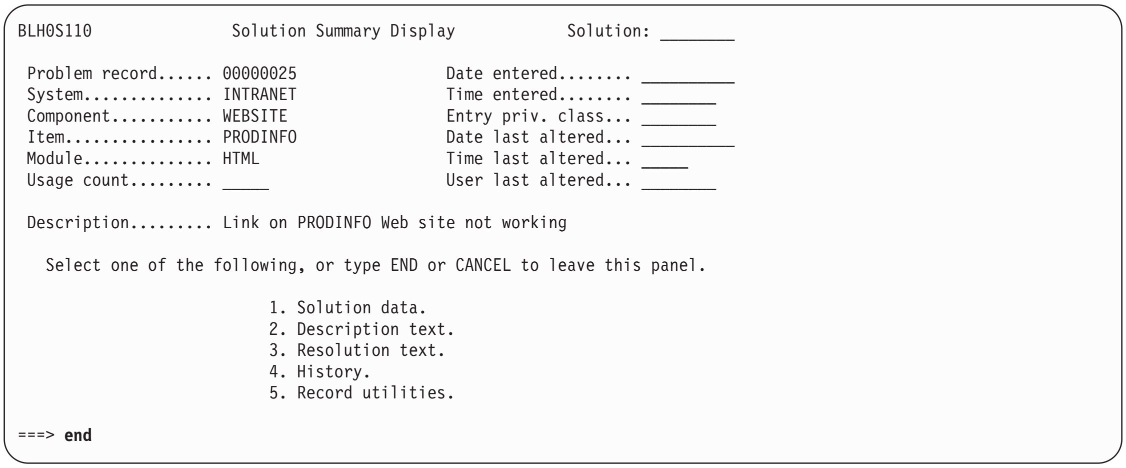
```
BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.


           Select an option, enter a command, or type QUIT to exit.


           Tivoli Information Management for z/OS Version 7 Release 1
                5697-SD9 (C) Copyright IBM Corp., 1981, 2001.

BLG03058I Record 00000026 was stored successfully.
===>
```

Note: After a solution record is created, if you go back and update the original problem record, you must also update the associated solution record.

Working with Solution Records

Like other records in the Tivoli Information Management for z/OS database, you can display, search, print, copy, and update solution records. Remember to specify the record ID of the solution record (not the problem record) when displaying or searching solution records.

The Solution Summary Display panel, BLH0S110, provides some additional functions. For example, select **5. Record utilities** to display a utilities panel that you can use to request a list of record references. The List Record References selection searches for p-word RN**/ with the current solution record ID value. If you have customized your system to add the solution record ID to the problem record when using the solution, then you can view a list of all solution references.



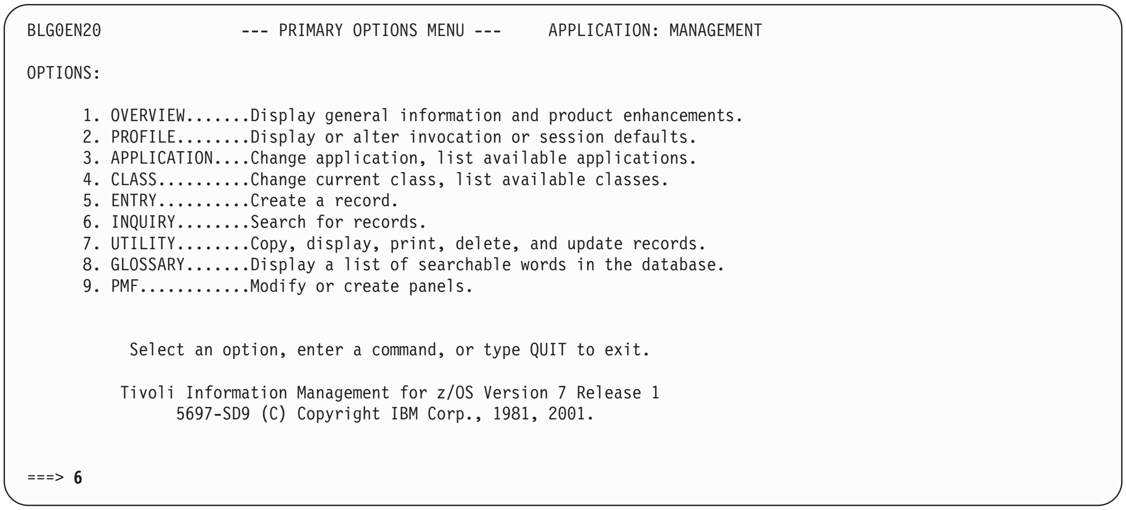
```
BLH0S110              Solution Summary Display           Solution: _______

 Problem record...... 00000025               Date entered........ _________
 System.............. INTRANET               Time entered........ _______
 Component........... WEBSITE                Entry priv. class... _______
 Item................ PRODINFO               Date last altered... _________
 Module.............. HTML                   Time last altered... _____
 Usage count......... _____                  User last altered... _______

 Description......... Link on PRODINFO Web site not working

   Select one of the following, or type END or CANCEL to leave this panel.

                           1. Solution data.
                           2. Description text.
                           3. Resolution text.
                           4. History.
                           5. Record utilities.

===> end
```

Searching for Solutions

You can search for information in solution records by doing an inquiry. Select **6. Inquiry** on the Primary Options menu and select the Solution option.

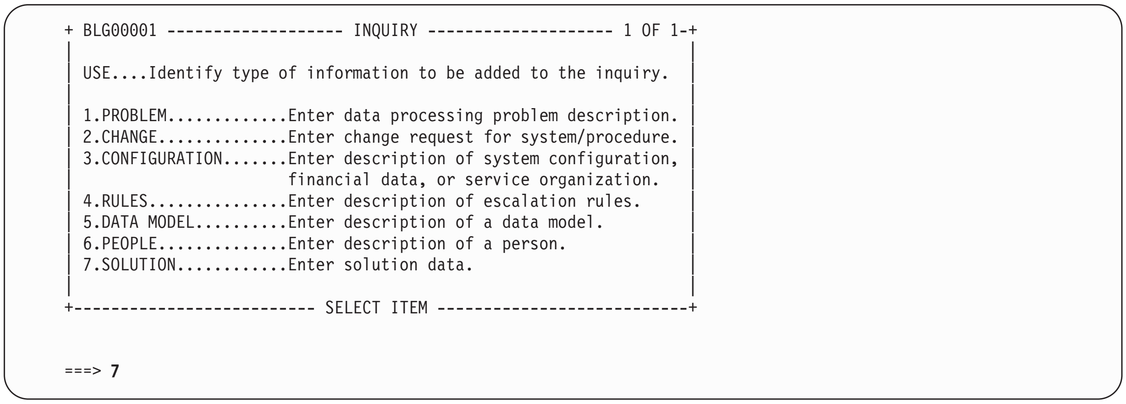
Doing an Inquiry of Solution Records

Select the **Inquiry** option on the Primary Options Menu in the Management application and press Enter.



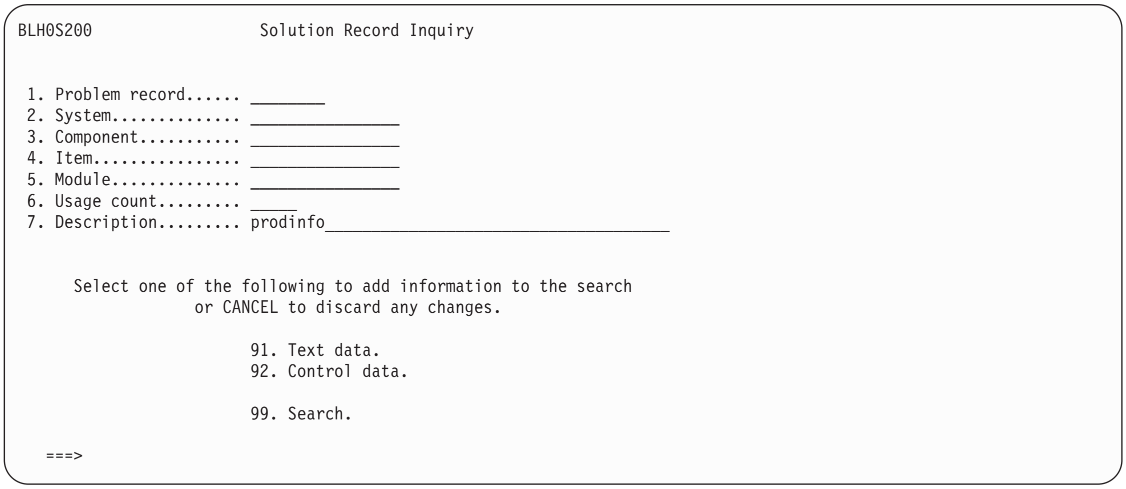
```
 BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

 OPTIONS:

       1. OVERVIEW.......Display general information and product enhancements.
       2. PROFILE........Display or alter invocation or session defaults.
       3. APPLICATION....Change application, list available applications.
       4. CLASS..........Change current class, list available classes.
       5. ENTRY..........Create a record.
       6. INQUIRY........Search for records.
       7. UTILITY........Copy, display, print, delete, and update records.
       8. GLOSSARY.......Display a list of searchable words in the database.
       9. PMF...........Modify or create panels.


            Select an option, enter a command, or type QUIT to exit.

           Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


 ===> 6
```

Select **7. SOLUTION** and press Enter.



```
+ BLG00001 ------------------- INQUIRY -------------------- 1 OF 1-+
|                                                                   |
| USE....Identify type of information to be added to the inquiry.  |
|                                                                   |
| 1.PROBLEM............Enter data processing problem description.  |
| 2.CHANGE.............Enter change request for system/procedure.  |
| 3.CONFIGURATION......Enter description of system configuration,  |
|                      financial data, or service organization.    |
| 4.RULES..............Enter description of escalation rules.      |
| 5.DATA MODEL.........Enter description of a data model.          |
| 6.PEOPLE.............Enter description of a person.              |
| 7.SOLUTION...........Enter solution data.                        |
|                                                                   |
+-------------------------- SELECT ITEM ---------------------------+


===> 7
```

Enter your search arguments on the Solution Record Inquiry panel.



```
BLH0S200                 Solution Record Inquiry


 1. Problem record...... ________
 2. System.............. ________________
 3. Component........... ________________
 4. Item................ ________________
 5. Module.............. ________________
 6. Usage count......... _____
 7. Description......... prodinfo_____________________________________


      Select one of the following to add information to the search
                   or CANCEL to discard any changes.

                         91. Text data.
                         92. Control data.

                         99. Search.

  ===>
```

You have the option to perform a search based on the solution usage count. A usage count is the number of times the solution has been used to solve a problem.

At this point you can decide what kind of search you want to perform. Select the option that suits your needs and press Enter:

- To search on paragraph style freeform text, select option 91. A Text Search Arguments panel (BLG1TTSA) is displayed to collect additional search arguments. Use of the Text Search Arguments panel is described in “Searching Freeform Text” on page 188. Although it can help to narrow down your search results, you do not have to specify a problem record ID or SCIM data to search the freeform text.
- To run a search based on the control data, select option 92. A Solution Control Data Inquiry panel is displayed to collect data for searching. You can search on the privilege class used when the solution record was created, the date or time the record was entered or updated, or the user ID of the person who last altered the record.
- Select option 99 to perform a search based on data provided on this panel, or to initiate the search based on criteria specified through use of the other inquiry options if you returned to this panel.

Any matching records are displayed in a search results list, as shown on page 190. Select the records to be displayed from the search results list using the line commands.

Deleting Solutions

You can use the delete utility or issue the DELETE command to delete solution records (or any other record that is indexed for freeform text search) from the Tivoli Information Management for z/OS database. A delete verification panel is displayed to confirm your deletion request. You can proceed with the request or cancel the deletion. The deletion of a solution record causes the solution data to be scheduled for removal from the solution text index. The text index entries are deleted the next time the BLMNINDX batch job is run to update the index.

Searching Freeform Text

For purposes of searching, freeform text is considered to be the 1-line description abstract and paragraph-style freeform text data associated with a record.

Note: Freeform text search is initiated from quick search panels, so your user profile must be set to `Quick search=yes`.

In Tivoli Information Management for z/OS, you can search freeform text indexed for use by the Text Search Engine for the following records:

Problem records
On panel BLG0E090, the Problem Inquiry Summary panel, select **10. Text data**.

Change records
On panel BLG0F090, the Change Inquiry Summary panel, select **7. Text data**.

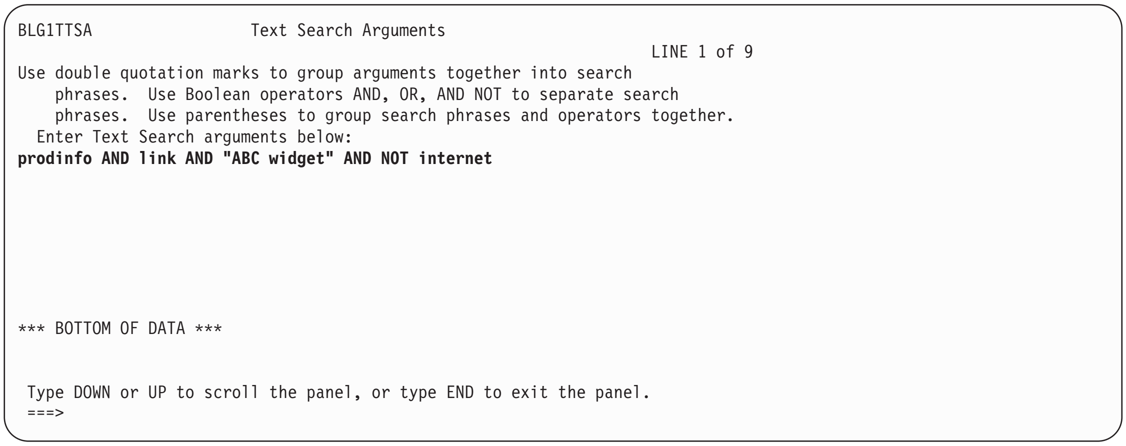
Solution records
On panel BLH0S200, the Solution Record Inquiry Summary panel, select **91. Text data**.

All of these options take you to BLG1TTSA, the Text Search Arguments panel, where you enter your freeform text search arguments. If you have created a user-defined index for your own types of records, you need to add a Text Data selection to the Inquiry Summary panel for each of those record types. Model your new Text data selection after selection **91** on panel BLH0S200. This will require that you update panel BLG1ATSA with the prefix word for your index.

Using the Text Search Arguments Panel

This section describes how to search freeform text using the Text Search Arguments panel BLG1TTSA. Freeform text search arguments must be entered through this panel; they cannot be entered on the command line. However, if you supply the search arguments on the lines in this panel, you can type ;SEARCH on the command line to perform the search rather than go back to the previous inquiry panel to perform the search.

Enter the search arguments for the freeform text.



```
BLG1TTSA                 Text Search Arguments
                                                             LINE 1 of 9
Use double quotation marks to group arguments together into search
    phrases.  Use Boolean operators AND, OR, AND NOT to separate search
    phrases.  Use parentheses to group search phrases and operators together.
  Enter Text Search arguments below:
prodinfo AND link AND "ABC widget" AND NOT internet









*** BOTTOM OF DATA ***



 Type DOWN or UP to scroll the panel, or type END to exit the panel.
 ===>
```

In this example, a search will be performed against the 1-line abstract, description freeform text, and resolution freeform text for the words *prodinfo* and *link* and the term *ABC widget* but not *internet*.

To search on text containing one word or another, use the OR operator (for example, `prodinfo OR link`).

The AND NOT operator will exclude records containing the text in the search (in this case, the argument shown excludes all indexed records containing the word *internet*).

To search for a phrase containing blanks, type the phrase with the blank spaces (for example, `"ABC widget"`). Quotation marks are needed to search on a phrase containing multiple words separated by a blank space.

See "Search Tips" on page 190 for more information about entering freeform text search arguments.

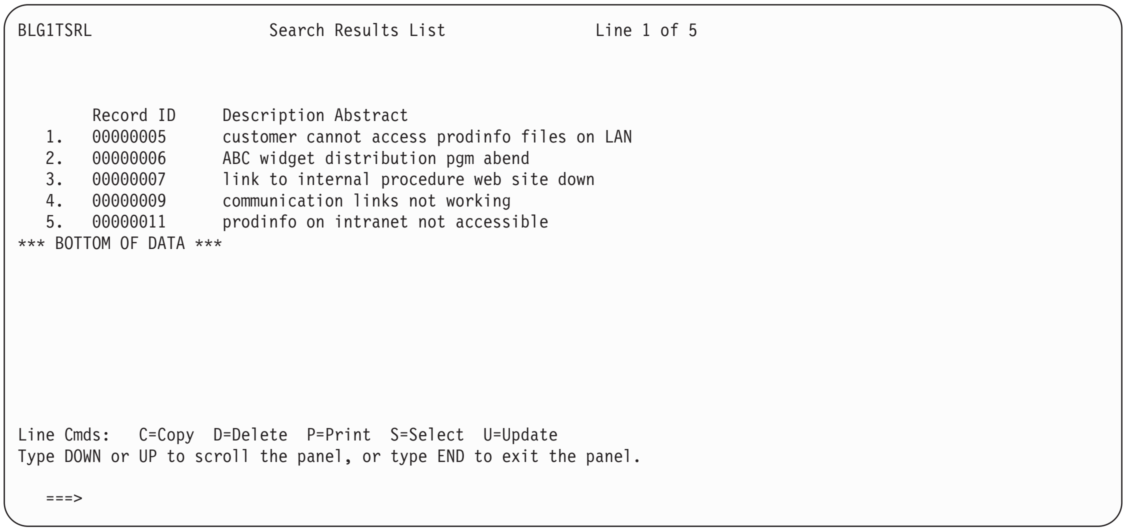
You can enter the argument on one line as shown, or use the space provided to separate sections of the argument. (If you run out of space, scroll forward to get more blank lines.) However, if you need to use more than one line, be sure to ″connect″ the argument appropriately by including a Boolean operator. For example, do this:

```
prodinfo AND
link AND
"ABC widget"
AND NOT internet
```

Do not do this:

```
prodinfo
link
"ABC widget"
NOT internet
```

Type **;SE** to perform the text search or type **end** and select the Search option to run the search from the previous inquiry panel. A list of matching record IDs is displayed on a search results list panel.



```
BLG1TSRL                 Search Results List              Line 1 of 5



       Record ID     Description Abstract
   1.  00000005      customer cannot access prodinfo files on LAN
   2.  00000006      ABC widget distribution pgm abend
   3.  00000007      link to internal procedure web site down
   4.  00000009      communication links not working
   5.  00000011      prodinfo on intranet not accessible
*** BOTTOM OF DATA ***









Line Cmds:   C=Copy  D=Delete  P=Print  S=Select  U=Update
Type DOWN or UP to scroll the panel, or type END to exit the panel.

   ===>
```

To display a record, type **s** to the left of the record ID and press Enter.

Note: To display the record details, you must have display authority for the type of record listed.

You can also combine a freeform text search with a search of other field data in the record by supplying data in the desired fields. When the search is performed, the results of both the Text Engine search and Tivoli Information Management for z/OS search are displayed in the search results list.

Search Tips

When searching freeform text, consider using the following Boolean operators and techniques to help make your search more precise:

- Boolean operators

  The following are Boolean operators: AND, OR, AND NOT, and parenthesis ( ). If you need to include any of these in your search arguments, specify them in ALL CAPS format. Be sure to include spaces around the operator also. For example:

  AND To find documents containing the words *printer*, *server*, and *fonts*, enter `printer AND server AND fonts`.

  OR To find documents containing the word *typestyle* or *fonts*, enter `typestyle OR fonts`.

  AND NOT
  To find documents that do not contain specific text, use the AND NOT operator. For example, to find documents containing the word *hardware* but not the word *printers*, enter `hardware AND NOT printers`.

  ( ) To group parts of your query together for more complicated searching, use parentheses. For example, to find documents containing the word *hardware* and either *printers* or *workstations*, enter `hardware AND (printers OR workstations)`.

- Fuzzy searches

You can find character strings that are similar to the specified search term. For example, a search for *Extender* finds the mistyped word *Extendrrs*. You can also specify a required degree of similarity. If you use fuzzy searches, the first three characters must match.

- Phrases

  Quotation marks are needed to search on a phrase containing multiple words separated by a blank space. For example, to find documents containing the phrase *token ring adapter* enter `"token ring adapter"` with double quotation marks. The Text Search Engine will find the words in the exact order entered.

- Special characters

  There are no special rules to follow. The types of characters you can use are dictated by the index and language specified.

  To mask a single character in a string, use the percent character (%). For example, to find documents containing the word *hardwire* or *hardware*, enter `hardw%re`.

- Wildcards

  Use the asterisk (*) wildcard to search for occurrences of text that may include slight variations. For example, to find documents containing the words *window*, *windows*, or *windowing*, enter `window*`.

- Case of text

  Searches are not case-sensitive. A search for *window*, *Window*, or *WINDOW* is performed the same way.

The indexing and search in the indexes is based on n-grams; that is, limited-length character sequences. The analysis is not based on a dictionary, and there is no linguistic processing involved. This technology enables high-performance indexing, and search using both exact and fuzzy matching. This index not only supports English and European languages (documents in codepage 850), but it is also optimized for double-byte character set languages such as Japanese.

Using Program Exits

The following specific program exits are also available for use if you are writing a program and require data collected from panels related to the use of freeform text search:

- BLG02157 – Freeform Text Search
- BLG02158 – Create Search Index
- BLG02159 – Delete Search Index

For more information about program exits, refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide*.

Data Model Records

Data view records and data attribute records are provided with Tivoli Information Management for z/OS to support text indexing:

Table 33. Data view records provided to support freeform text search

Data View Records	
Name	**Description**

Table 33. Data view records provided to support freeform text search (continued)

Data View Records	
BLHRINDX	Retrieve index record
BLHRCHGT	Retrieve change record text data
BLHRPRBT	Retrieve problem record text data
BLHRSOLT	Retrieve solution record text data

Table 34. Associated data attribute records

Data Attribute Records	
Name	**Description**
BLH&COMP	Component
BLH&COUX	Solution usage count
BLH&DESC	Description abstract
BLH&DVRC	Indexed record data view name
BLH&INDX	Text index name
BLH&ITEM	Item
BLH&ITYP	Associated record s-word index
BLH&MODL	Module
BLH&NNDX	Index record identifier
BLH&RNPD	Original problem record ID
BLH&SYST	System
BLH&TERM	Text search term
BLH&UNDX	User-defined index name

10

Understanding Your Tivoli Information Management for z/OS Database

This chapter provides an overview of the Tivoli Information Management for z/OS database and describes how you can take advantage of the various options provided by Tivoli Information Management for z/OS to set up a database that best suits your organization's needs. In addition, it discusses the methods available within Tivoli Information Management for z/OS to help you back up and recover your database.

The Databases

Users of Tivoli Information Management for z/OS can be interactive users, batch jobs, or applications that call API transactions. As a user proceeds through a set of panels, making selections and entering data, Tivoli Information Management for z/OS collects the structured words (s-words) and prefix words (p-words) that correspond to the fields and data being entered. When the user files the record, the s-word and p-word data used for searching goes into the SDIDS. That data, along with everything else, goes into the SDDS.

These data sets form the actual database for Tivoli Information Management for z/OS. The records created by users are stored in the structured description data set (SDDS) in the form of structured descriptions. The records are located by keywords stored in the structured description index data set (SDIDS).

The structured description log data set (SDLDS) is an optional data set used as a backup for records written to the SDDS.

The pairing of an SDDS and SDIDS constitute a database. Tivoli Information Management for z/OS uses two types of databases: Tivoli Information Management for z/OS format databases and user-defined format databases. Both types of formats are made up of SDDS and SDIDS data sets. Databases 0, 1, 2, and 3 are user-defined format databases; databases 4, 5, 6, 7, 8, and 9 are Tivoli Information Management for z/OS format databases. Database 5 is a read/write database. The other databases are read-only databases. The Tivoli Information Management for z/OS read/write database can also contain an SDLDS. The two formats differ in the way that you enter data into the database. You enter data into a Tivoli Information Management for z/OS format read/write database interactively through Tivoli Information Management for z/OS panels or dialogs or through an API. A user-defined database receives data loaded through a utility program (BLGOZUD). The BLGOZUD utility enables you to load data into the database, which you can search, display or print, but you cannot update the records in a user-defined database. A detailed description of the BLGOZUD utility is provided in the *Tivoli Information Management for z/OS Operation and Maintenance Reference*.

Structured Description Data Set (SDDS)

The Structured Description Data Set (SDDS) is made up of from 1 to 100 VSAM key-sequenced data sets (clusters) that contain the data records. For a Tivoli Information Management for z/OS format database, the SDDS contains the information supplied by the user (either interactively through panels or using API programs) or collected by Tivoli Information Management for z/OS. Each Tivoli Information Management for z/OS SDDS logical record can consist of one or more physical VSAM records. For a user-defined format database, the SDDS contains the text entries that you load from a sequential file.

Tivoli Information Management for z/OS requires that you define an SDDS. The SDDS of a Tivoli Information Management for z/OS format database can consist of from 1 to 100 VSAM clusters. A 100-cluster SDDS enables users to store up to 400GB (GB equals 1 073 741 824 bytes) of information in a database.

The SDDS of a user-defined format database can consist of only 1 VSAM cluster.

Figure 6 on page 195 illustrates the way logical records are filed in multiple data sets, using five VSAM clusters as an example. Several items are important in understanding how records are stored in multiple data sets:

- A *logical record* in Tivoli Information Management for z/OS consists of one or more physical VSAM records.
- Each *physical* VSAM record that is related to the same Tivoli Information Management for z/OS logical record has the same SDDS position number (also called a *root VSAM key* or *VSAM sequence number*).
- The SDDS position number is the second through fourth byte of the physical VSAM key.
- All physical VSAM records with the same SDDS position number are stored in the same VSAM data set.

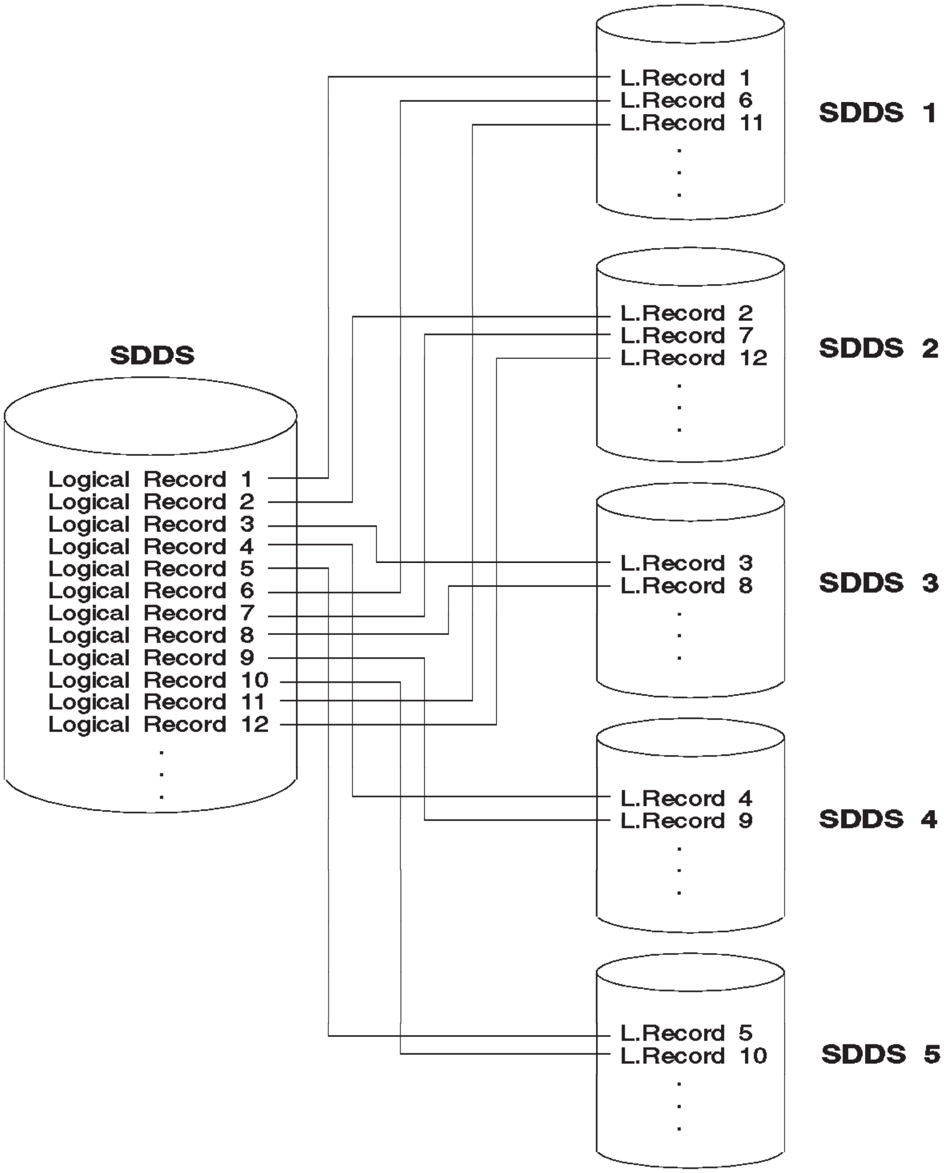


Figure 6. Logical Record Locations in an SDDS

Structured Description Index Data Set (SDIDS)

The Structured Description Index Data Set (SDIDS) is a special high-performance index that accelerates the search for records in the SDDS. It contains the words that you use as search keywords for records in the SDDS. The SDIDS is sometimes called the glossary. Each logical record in the SDIDS contains a searchable word and position markers to indicate which records in the SDDS contain that word. Each logical record can consist of one or more physical VSAM records. This is true for both Tivoli Information Management for z/OS and user-defined format databases.

The SDIDS is made up of from 1 to 100 VSAM key-sequenced data sets (clusters). You define one or more clusters, depending on your needs. The number of SDIDS clusters and SDDS clusters do not have to be the same, because they are independent of one another. The SDIDS data is compressed.

The information in the SDIDS is created and managed by Tivoli Information Management for z/OS and is in addition to the index component that VSAM defines for any key-sequenced data set.

Figure 7 shows the relationship between the words in the SDIDS and the records in the SDDS.

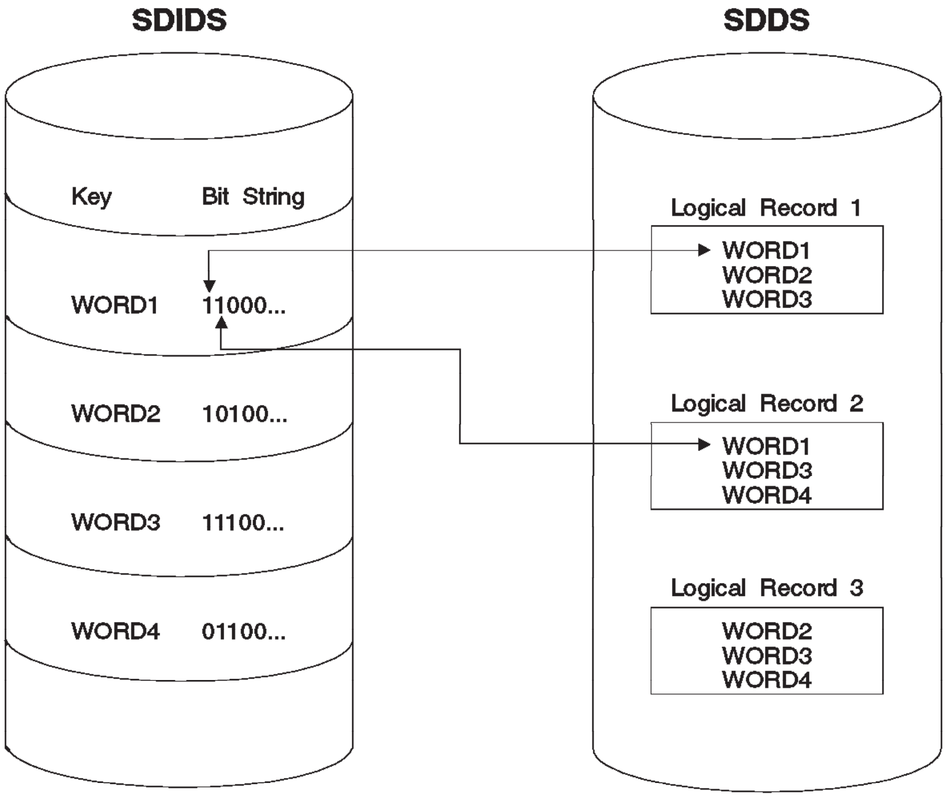


Figure 7. Relating the SDIDS to the SDDS.

This figure illustrates, for example, that if you enter a search argument of **WORD1**, the search mechanism reads the SDIDS record for **WORD1** and checks the bit string to determine which records contain **WORD1**. If you enter a search for **WORD1** or **WORD3** (to retrieve records with both words), the search mechanism reads the SDIDS for **WORD1** and **WORD3** and performs a logical AND operation on the bit string of each word.

Structured Description Log Data Set (SDLDS)

The Structured Description Log Data Set (SDLDS) is an optional data set that stores copies of the records written to the SDDS. The SDLDS provides a backup for changes made to the SDDS. If the SDDS is damaged, you can use the records in the SDLDS to reconstruct the changes made to the SDDS since its last backup.

The SDLDS is a VSAM relative record data set that consists of a single VSAM cluster and contains fixed-length records. Only the Tivoli Information Management for z/OS read/write database (database 5) has an SDLDS.

Your Database Options

One of the advantages of using the Tivoli Information Management for z/OS database is its enormous storage capacity. You can store a significant amount of data for tracking and reporting at an enterprise level. Another advantage is its ability to be customized to suit your specific needs and business processes. Not all organizations have the same needs for database access, searching, storage, or performance. If you are new to Tivoli Information Management for z/OS, or if you are an existing user looking at ways to improve your database operation, you should consider the options described in this section when evaluating what choices you should make for your overall database environment.

Multiple-Cluster SDDS

The data component of a VSAM key-sequenced data set (KSDS) can hold a maximum of 4GB (GB equals 1 073 741 824 bytes) of information. In Tivoli Information Management for z/OS, the SDDS component of your database is usually the largest data set. If you expect the SDDS to exceed 4GB, you can use up to 100 VSAM clusters for your SDDS, giving you enough space for 400GB of information. Unless you have more than a million logical records in your SDDS, you probably need not worry about exceeding the single-cluster limit. However, you may want to use multiple SDDS clusters for performance reasons.

Using a multiple-cluster SDDS does not change the way you use Tivoli Information Management for z/OS. The only exceptions are backup procedure changes, some session-parameters member changes, some VSAM resource definition changes, and some syntax changes needed for activating multiple SDDS clusters when running utility programs such as BLGUT1 and BLGUT7 . For information on the utility programs, refer to the *Tivoli Information Management for z/OS Operation and Maintenance Reference*.

For existing Tivoli Information Management for z/OS databases, you can run BLGUT20 to obtain statistics that can help you determine the total SDDS size and the number of logical records it contains. You can use these statistics to determine how close your SDDS is to 4GB and the number of logical records your database holds. For information on the BLGUT20 utility program, refer to the *Tivoli Information Management for z/OS Operation and Maintenance Reference*.

For details on setting up a multiple-cluster SDDS, refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference*.

SDDS Key Length

The SDDS consists of one or more key-sequenced VSAM data sets. When defining the SDDS, your system administrator specifies the length of the key field in the SDDS VSAM record as 7 or 8 bytes for a Tivoli Information Management for z/OS format database. Key 7 format generally provides better performance, and is the recommended format. Key 7 format is also required to perform some functions of Tivoli Information Management for z/OS, such as backing up your database using the utility BLGUT23.

The SDDSs for databases 0, 1, 2, and 3 must have a key length of 8.

Details on how the key is specified are provided in the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* .

Multiple-Cluster SDIDS

Like the SDDS, the SDIDS is a key-sequenced VSAM data set that can consist of either a single or multiple clusters. Each SDIDS can contain up to 4GB of data. The maximum amount of data you can store in the SDIDS is 400GB.

Each SDIDS cluster contains only a subset of the entire data set. A multiple-cluster SDIDS can be useful to help improve your overall database performance because the clusters can be accessed in parallel. If you store significant amounts of searchable data, you may benefit from setting up a multiple-cluster SDIDS to divide your records based on key contents. For example, you might choose to separate s-words from p-words, or some other arrangement.

When evaluating whether or not you should set up a multiple-cluster SDIDS, you should consider how records are updated in your organization (concurrently or not), what kind of search contention will exist for those records in the same cluster, the number of records that will exist per cluster, and how much read/write activity is anticipated for the records.

For details on setting up a multiple-cluster SDIDS, see the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* .

SDIDS Key Length

The SDIDS consists of one or more key-sequenced VSAM data sets. Your system administrator can specify, when defining the SDIDS, that the key length be 18 or 34 bytes long.

For search purposes, both the 18- and 34-byte key can be used with single-byte character set (SBCS) or double-byte character set (DBCS) data. The 34-byte key is highly recommended for DBCS users. The 18-byte key is not recommended for DBCS because it supports only 7 DBCS characters when doing a DBCS search. The 34-byte key supports up to 15 DBCS characters on a search.

If you use SBCS characters, you can use either an 18-byte or a 34-byte key. With the 18-byte, the search keyword size can be up to 16 characters. The search keyword can be up to 32 characters with the 34-byte key.

Note:

- If you plan to implement the Tivoli Service Desk Bridge (described in the *Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications*), your problem database (database 5) SDIDS must be defined with a 34–byte key.
- If you plan to implement the Interface to Tivoli Inventory (described in the *Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications*), the SDIDS for the Inventory database (database 6) must be defined with a 34–byte key.

For more information on how the key length setting affects searching, refer to the *Tivoli Information Management for z/OS User's Guide*. Details on how the key is specified when the SDIDS is defined are provided in the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference*.

Determining Your Backup/Recovery Approach

Tivoli Information Management for z/OS provides various ways for you to backup your data and recover it if necessary. Depending on your specific needs, the size of your data base, your desired frequency for backups, and your data base availability needs, you may decide that one approach is better than another. The methods available to you are:

- Use the BLGUT3 utility to restore the data base from the log data set created using BLGUT4 and pruned using BLGUT4LP. This method involves using the AMS REPRO command to create a backup of the SDDS and SDIDS. This method is commonly used and has been available for a number of years. The advantage of this method is that it is simple. The disadvantage is that the SDDS and SDIDS are unavailable while AMS REPRO copy is being created.
- Use the BLGUT23B utility to create a new master data set each time a backup is desired. The new master combined with a SDLDS large enough to hold all changes between runs of BLGUT23B form a complete backup copy of your data base. If it is necessary to restore the data base, BLGUT23R would be used after the SDLDS is offloaded with BLGUT4 and the one log processed using BLGUT23P.

  The advantage of this is that the data base is always available and the processing steps are simple. This approach may be ideal for small organizations that use a single cluster SDDS that in the past has been backed up using IDCAMS repro.

  The disadvantage of this is that the SDLDS may have to be very large to hold all the changes that occur to the data base between runs of BLGUT23B. Note, however, that a large SDLDS does not impact performance of Tivoli Information Management for z/OS. Multiple–cluster SDDS users would have to use a sort merge process to merge masters created by BLGUT23B to create a complete master.
- Use the complete series of BLGUT23 utilities to create and merge data into a master backup data set. The advantage here is that your data base is available 24 hours a day, 7 days a week. Because BLGUT23B can take several hours to run each time, this method would avoid having to rerun BLGUT23B to create a new master. BLGUT23B would only be run once when the initial master is created. Then the master would be kept current using BLGUT23P and BLGUT23U after the SDLDS is offloaded periodically using BLGUT4. This backup method may be ideal for large enterprises with multiple SDDS clusters or high update activity which prevents the SDLDS from containing all the changes between backups. The disadvantage is the need to frequently merge the offloaded logs into the master using BLGUT23P and BLGUT23U.
- The Automatic Log Save facility periodically offloads the SDLDS and the offloaded log is then used to update another Tivoli Information Management for z/OS data base. The Automatic Log Save facility enables organizations to maintain a complete shadow copy of their Tivoli Information Management for z/OS data base.

  The advantages are:

  - The production data base is always available (24 by 7).
  - A production data base can be recovered very quickly by simply altering the SDDS and SDIDS cluster names of the backup data base to match the production data base. Or, after deleting the corrupted SDDS and SDIDS, you can use repro to replace the production data base with the backup data base. After altering the cluster names or completing the repro and processing any remaining offloaded logs, the data base could be placed back into production.

- If the Automatic Log Save facility data base is defined in the session–parameters member as read–only, it can be used as reporting data base. This reduces the number of users on the production data base and should improve performance of the production data base.

The disadvantage of the Automatic Log Save facility is that it requires a duplicate SDDS and SDIDS.

Reuse of SDDS Position Numbers (Root VSAM Keys)

Each logical record (RNID) on a data base is assigned an SDDS position number (sometimes called a root VSAM key or a VSAM sequence number) to uniquely identify that record on the data base. When a new record is created, it is assigned a position number and stored in that location on the database. When the record is deleted, its SDDS position number becomes unassigned and its slot can be reused.

In earlier releases of Tivoli Information Management for z/OS, SDDS position numbers were always reused. In Release 1.1, however, the default is to NOT reuse the numbers. When SDDS position numbers are reused, records are kept close together on the database, although not in any particular order. By not reusing the SDDS positions, new records are always added to the end of the database. Thus, whether you have a new database or an existing database, the SDDS position numbers are not reused unless you specifically request this option.

An advantage of not reusing the SDDS position number is that a search results list does not need to be sorted in order to display records in system-assigned RNID order. This should improve search performance by reducing contention for the SDIDS because the search results list is already in system-assigned RNID order. On the other hand, if SDDS position numbers are not reused, any new records are always added to the end of the database, even though space may be available elsewhere where other records have been deleted. The unused space can be reclaimed by using IDCAMS to reorganize the SDDS whenever a significant number of records has been deleted.

The Set Database Options utility, BLGUT9, introduced in Release 1.1, allows you to select the option of reusing SDDS position numbers. See the discussion of BLGUT9 in the *Tivoli Information Management for z/OS Operation and Maintenance Reference* for additional information on this utility.

Creating Database Sorted by Record Number

You can use the BLGUT23 series of utilities to create a copy of an existing database. Then sort the copy of the database using RNID as the sort key. This sorted copy of the database can be used to create a database with the records in RNID order. If VSAM position numbers are not reused, then the database will stay in system-assigned RNID order. However, the order of user-assigned records will not be maintained.

Note: After performing this process you should not run BLGUT9.

Planning

- Your SDDS must use the key 7 format. If you are using the key 8 format, you must convert to the key 7 format using BLGUT7 before using this process. Your database will be unavailable while BLGUT7 runs.

- This procedure is not necessary for new (empty) databases, because they will start in system-assigned RNID order unless BLGUT9 is used to set the database option to reuse VSAM position numbers.
- Your database can be used up to the time that you free the SDDS(s) and SDIDS(s). The SDLDS will contains the updates that occur from the time BLGUT23B starts until you actually begin freeing the SDDS and SDIDS.
- You must run BLGUT1 as part of this process.
- Automatic Log Save users must re-synchronize the Send and Receive databases after performing this procedure.
- You may want to perform the following process on a small test database so that you can become familiar with the BLGUT23 series of utilities and perfect your JCL for merging (if you are using multiple SDDS clusters) and sorting of the master file that will be used as input to BLGUT23R.
- Because your database will be kept in system-assigned RNID order, you should review any table panels used to display a search results list. If those table panels used the sort option of PREFIX and the prefix was RNID you may want to change them to use a sort option of DATABASE instead. See the section on “Sorting the Search Results List” in the *Tivoli Information Management for z/OS Panel Modification Facility Guide*.

Procedure

To sort your existing database do the following:

1. Use BLGUT4 to offload the SDLDS. This offload load will not be used as part of creating the sorted database. However, it should be kept until BLGUT23B completes successfully.
2. Utilize BLGUT23B for each SDDS cluster. Users can continue to use the database while BLGUT23B is running.
3. If you have multiple SDDS clusters, merge the individual BLGUT23B backups into one master backup. See BLGUT23B in the *Tivoli Information Management for z/OS Operation and Maintenance Reference* for more information.
4. After BLGUT23B completes and you have merged the SDDS clusters (if you are using a multiple cluster SDDS), have your users stop using the database.
5. Free the SDDS(s) and SDIDS. Your users cannot access the database until the remainder of these steps are complete.
6. Run BLGUT4 to offload the SDLDS.
7. Run BLGUT23P to prune the offload log.
8. Run BLGUT23U to merge the pruned log with the master backup.
9. Sort the new updated master using the following sort control card

   ```
   SORT FIELDS=(9,8,CH,A),EQUALS,SIZE=Ennnnnn
   ```

   where `Ennnnn` is the total number of physical records in the master backup data set. Refer to the BLGUT23U SYSPRINT data set for the BLG21331I message. Typically, there are three physical records for each logical record. BLGUT20 will give you the actual average number of physical records of your data base. The above sort control card is for DFSORT™. You may have to modify it if you use a different sort program.

10. After the sort successfully completes, use IDCAMS to delete your SDDS(s) and SDIDS(s). Then define new (empty) clusters using the same attributes as your existing clusters.
11. Reallocate the SDDS(s) and SDIDS(s) using the UTIL option on the BLX-SP REALLOC command.
12. Run BLGUT23R using the **RESEQ** parameter.

    Note: Do not use a BLGLOGUP input data set.
13. After BLGUT23R completes successfully, run BLGUT1 to rebuild the SDIDS.
14. After BLGUT1 completes successfully, reallocate the SDDS(s) and SDIDS(s).
15. Access the database using Tivoli Information Management for z/OS; do a search and display the results on a table panel that has the sort option set to DATABASE. The records should appear in ascending or descending order depending on the sort order used on the table panel. (Ascending is the default.)
16. If the database appears the way you expect, you can allow your users to access the database. You might want to make a backup copy at this time using your preferred backup utility or BLGUT23B. If you use BLGUT23B, you can allow your users to access the database while it runs.

    Note: Automatic Log Save users should recreate their receive database by using the procedure described in “Restoring Synchronization without Using BLGUT1” on page 284.
17. Once this BLGUT23B successfully completes for all of your SDDS clusters (and the merge if necessary), discard the sorted master from which you created the sorted database and also discard any offloaded logs or older master backups.

11

Logical Database Partitioning

Logical Database Partitioning makes it possible for you to organize data in your Tivoli Information Management for z/OS database into “logical partitions”. These partitions can be completely isolated from each other when viewed by a user in a given partition, but accessible as a single database to selected authorized personnel. The ability to partition the Tivoli Information Management for z/OS database provides an additional measure of administrative control. For example, if you provide service to different divisions, you can separate the records of one division from the records of another division. This is done by assigning records to logical partitions.

The database administrator controls which users have access to which partitions. This is done by specifying partition names in privilege class records.

Note: Before adding partition information to privilege classes, ensure that the BLGTSX DD is allocated and that the SBLMTSX data set is specified in the BLGTSX concatenation. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for more information on allocating BLGTSX.

Primary Partition Name

A **Primary Partition Name** can be added to a privilege class record. This Primary Partition Name identifies the logical partition for users in that privilege class, and thus the records, to which these users have access. The Primary Partition Name is assigned to every record that is created. Searches and other record access functions automatically access only the records in the primary partition.

Owning Partition Name

When a user enters a record into the Tivoli Information Management for z/OS database, the Primary Partition Name from the current privilege class is collected as the **Owning Partition Name** in the record. Only users with access to that partition can access that record.

Note: If a privilege class record does not contain a Primary Partition Name, then database partitioning is not used and a user of this privilege class has access to every record in the database. Records are not assigned an Owning Partition if they are entered by a user using a privilege class that does not have a Primary Partition Name. This means that all such records in the database will be accessible to any users having a privilege class that does not contain a Primary Partition Name.

Secondary Partition List

If you are using Logical Database Partitioning and you want users in one partition to be able to access records in another partition, you can set up a **Secondary Partition List** in the users' privilege class record. The Secondary Partition List defines additional partitions that the privilege class can access. Users can access records in any of their secondary partitions; however, they cannot create any records in these partitions. A user cannot use record commands, such as UPDATE or DISPLAY to access a record in a secondary partition. Records in secondary partitions are only accessible through a search results list. See “Record Searches” on page 217 for information on how to access records in secondary partitions.

Authorizing a Privilege Class for Logical Database Partitions

This is a summary of how to set up your privilege class records for logical database partitioning:

- Create a privilege class record (or modify an existing privilege class record) and add a Primary Partition Name to the privilege class record. Any records entered into the database by users in that privilege class are assigned an Owning Partition Name corresponding to the Primary Partition Name from the privilege class.
- If the privilege class needs access to more than one partition, you can modify the privilege class record by adding a Secondary Partition List. This Secondary Partition List identifies partitions (in addition to the Primary Partition) that the privilege class can access. Records in secondary partitions are only accessible from a search results list.

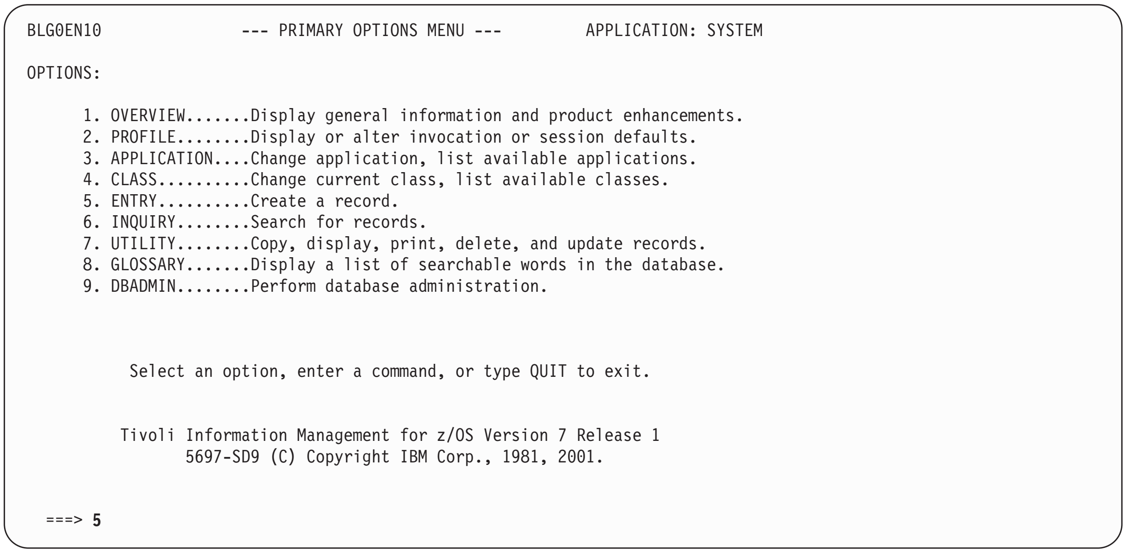
The following panels illustrate how to create a privilege class containing a Primary Partition name and add Secondary Partitions to that privilege class.

Notes:

1. If a privilege class does not contain a primary partition name, database partitioning is not used. Users in that class have access to all records in the database.
2. The following examples show the list processor panels used by Tivoli Information Management for z/OS Version 7.1 to collect and display data for new privilege class records (BLG0JU00 and BLG0Q100). If you are working with older records and have not run TSX BLGTPRIV, you may see the panels in use prior to Tivoli Information Management for z/OS Version 7.1. For information about BLGTPRIV, see the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference*. For more information about privilege classes, see “Managing Privilege Classes” on page 27.

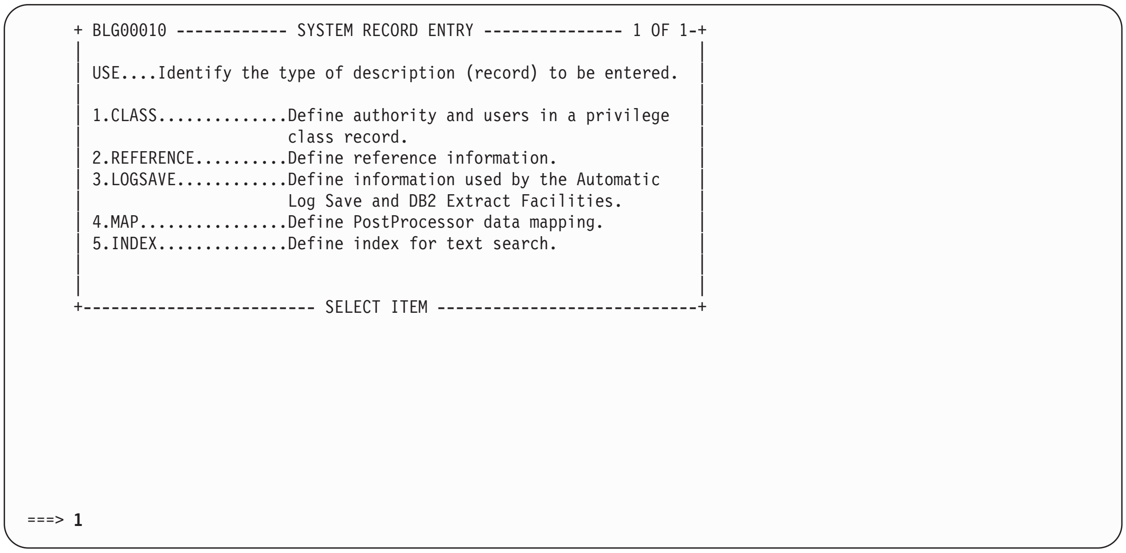
Note: If the application specified in the upper-right corner of the panel is `Management`, change the setting to `System`.

From the Primary Options Menu, BLG0EN10, select **ENTRY**. Type **5** and press **Enter**.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---         APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.



           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


  ===> 5
```

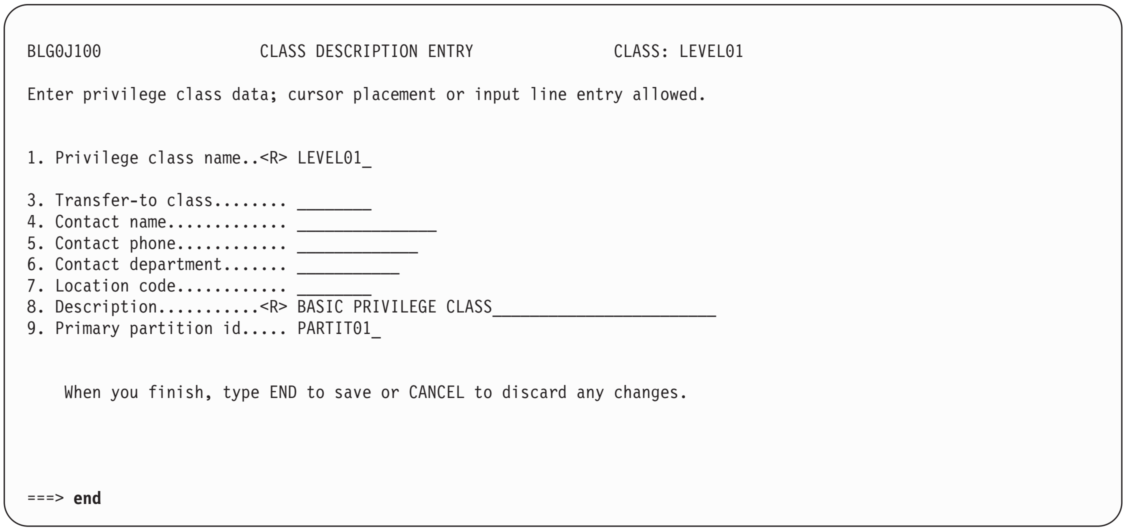
To create a privilege class record, type **1** and press **Enter**r.



```
       + BLG00010 ------------ SYSTEM RECORD ENTRY --------------- 1 OF 1-+
       |                                                                  |
       | USE....Identify the type of description (record) to be entered.  |
       |                                                                  |
       | 1.CLASS.............Define authority and users in a privilege    |
       |                     class record.                                |
       | 2.REFERENCE.........Define reference information.                |
       | 3.LOGSAVE...........Define information used by the Automatic     |
       |                     Log Save and DB2 Extract Facilities.         |
       | 4.MAP...............Define PostProcessor data mapping.           |
       | 5.INDEX.............Define index for text search.                |
       |                                                                  |
       |                                                                  |
       +------------------------ SELECT ITEM ----------------------------+







===> 1
```

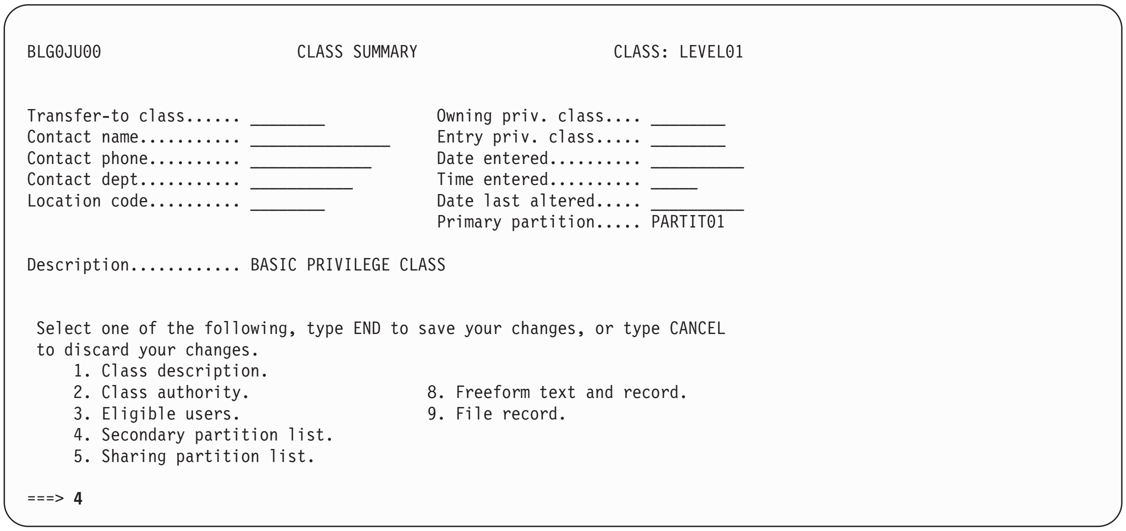
On BLG0J100, the Class Description Entry panel, complete the required fields and any others you choose. In the **Primary partition id** field, type the Primary Partition Name. In this example, a Primary Partition Name of **PARTIT01** is assigned to privilege class LEVEL01.

When you have entered the information, type **end** and press **Enter**.



```
BLG0J100                CLASS DESCRIPTION ENTRY              CLASS: LEVEL01

Enter privilege class data; cursor placement or input line entry allowed.


1. Privilege class name..<R> LEVEL01_

3. Transfer-to class........ ________
4. Contact name............. _______________
5. Contact phone............ _____________
6. Contact department....... ___________
7. Location code............ ________
8. Description..........<R> BASIC PRIVILEGE CLASS________________________
9. Primary partition id..... PARTIT01_


    When you finish, type END to save or CANCEL to discard any changes.




===> end
```

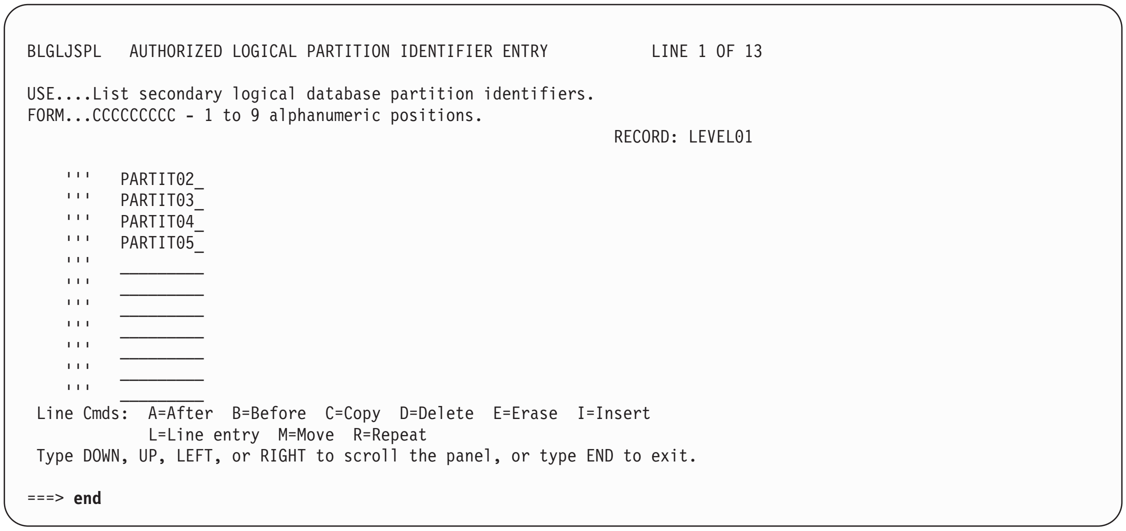
On BLG0JU00, the Class Summary panel, select **4** to specify a list of secondary partitions accessible to a user in this privilege class. Type **4** and press **Enter**.



```
BLG0JU00                   CLASS SUMMARY                     CLASS: LEVEL01


Transfer-to class...... ________            Owning priv. class.... ________
Contact name........... _______________     Entry priv. class..... ________
Contact phone.......... _____________       Date entered.......... __________
Contact dept........... ___________         Time entered.......... _____
Location code.......... ________            Date last altered..... __________
                                            Primary partition..... PARTIT01

Description............ BASIC PRIVILEGE CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.
     2. Class authority.                  8. Freeform text and record.
     3. Eligible users.                   9. File record.
     4. Secondary partition list.
     5. Sharing partition list.

===> 4
```

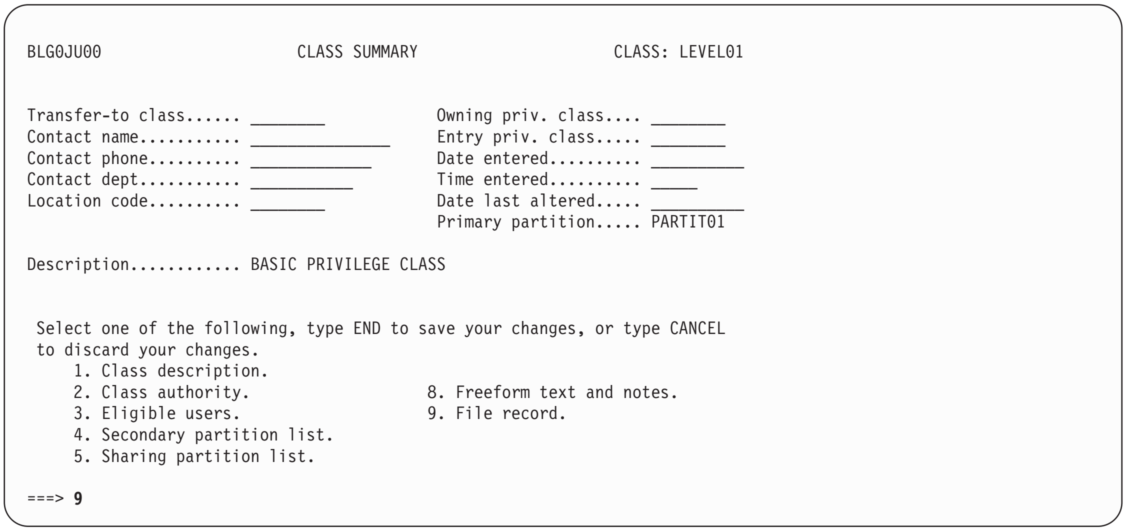
On BLGLJSPL, list other partitions which a user in this privilege class can access. In this example, users in privilege class LEVEL01 are given access to secondary partitions PARTIT02, PARTIT03, PARTIT04, and PARTIT05.

When you have completed the list, type **end** and press **Enter**.



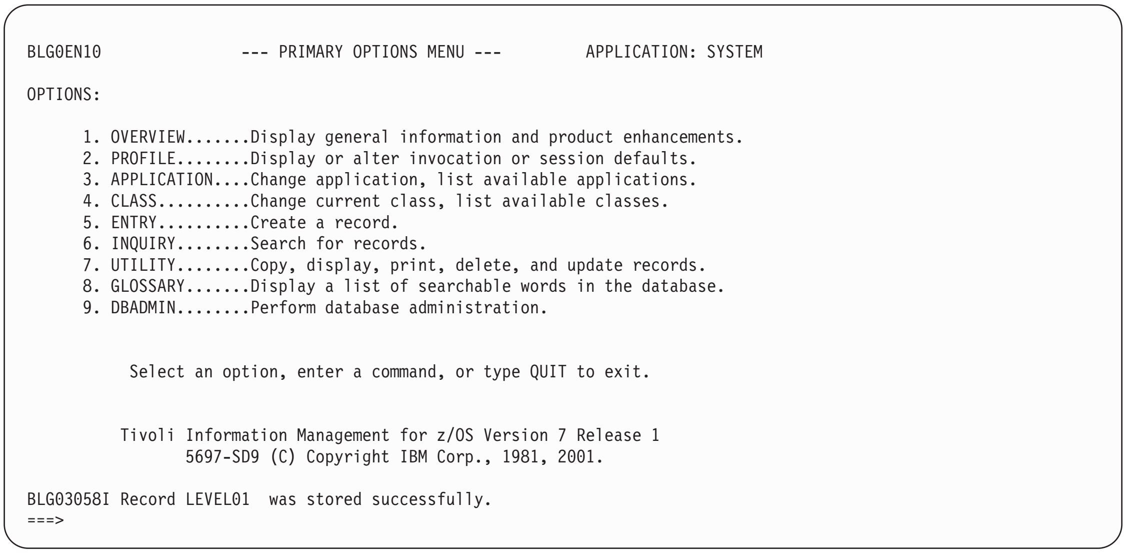
```
BLGLJSPL   AUTHORIZED LOGICAL PARTITION IDENTIFIER ENTRY           LINE 1 OF 13

USE....List secondary logical database partition identifiers.
FORM...CCCCCCCCC - 1 to 9 alphanumeric positions.
                                                               RECORD: LEVEL01

    '''   PARTIT02_
    '''   PARTIT03_
    '''   PARTIT04_
    '''   PARTIT05_
    '''   _________
    '''   _________
    '''   _________
    '''   _________
    '''   _________
    '''   _________
    '''   _________
 Line Cmds:  A=After  B=Before  C=Copy  D=Delete  E=Erase  I=Insert
             L=Line entry  M=Move  R=Repeat
 Type DOWN, UP, LEFT, or RIGHT to scroll the panel, or type END to exit.

===> end
```

You return to the Class Summary panel. To save the privilege class record, select **File record**. Type **9** and press **Enter**.



```
BLG0JU00                    CLASS SUMMARY                    CLASS: LEVEL01


Transfer-to class...... ________            Owning priv. class.... ________
Contact name........... _______________     Entry priv. class..... ________
Contact phone.......... _____________       Date entered.......... __________
Contact dept........... ___________         Time entered.......... _____
Location code.......... ________            Date last altered..... __________
                                            Primary partition..... PARTIT01

Description............ BASIC PRIVILEGE CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.
     2. Class authority.                   8. Freeform text and notes.
     3. Eligible users.                    9. File record.
     4. Secondary partition list.
     5. Sharing partition list.

===> 9
```

Note that a message confirms that record LEVEL01, the privilege class record that you have created, was stored successfully.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.

BLG03058I Record LEVEL01  was stored successfully.
===>
```

Privilege Class Selection

When a user selects a privilege class, the user is assigned the Primary Partition Name and any Secondary Partitions associated with that privilege class. If a user selects a different privilege class, the Primary Partition Name and Secondary Partitions associated with that privilege class are assigned.

A user can access the records in any partitions authorized by the privilege class. However, users can only create records in their primary partition; record commands and utility functions such as UPDATE or DISPLAY only access records in their primary partition. Records in secondary partitions are accessible from a search results list. By default, a search only searches records in a user’s primary partition, but secondary partitions can be added to the search as described in “Record Searches” on page 217.

The Invocation Class in a user’s profile designates a privilege class that the user automatically runs in when starting Tivoli Information Management for z/OS. If the user wishes to change the Invocation Class in their profile, the chosen privilege class must contain the user’s current Primary Partition as either its Owning Partition Name or as an entry in the Sharing Partition List (described in “Sharing Partition List” on page 214).

Although you cannot have duplicate RNIDs within a partition, you may have duplicate RNIDs across partitions. For that reason, care must be taken when specifying either the **Invocation Class** in a user’s profile or the **CLASS(NAME)** on the ISPSTART command because only unique privilege class RNIDs are supported via these mechanisms. Duplicate privilege class RNIDs are supported via the Privilege Class selection list (Option 4 on the Primary Options menu).

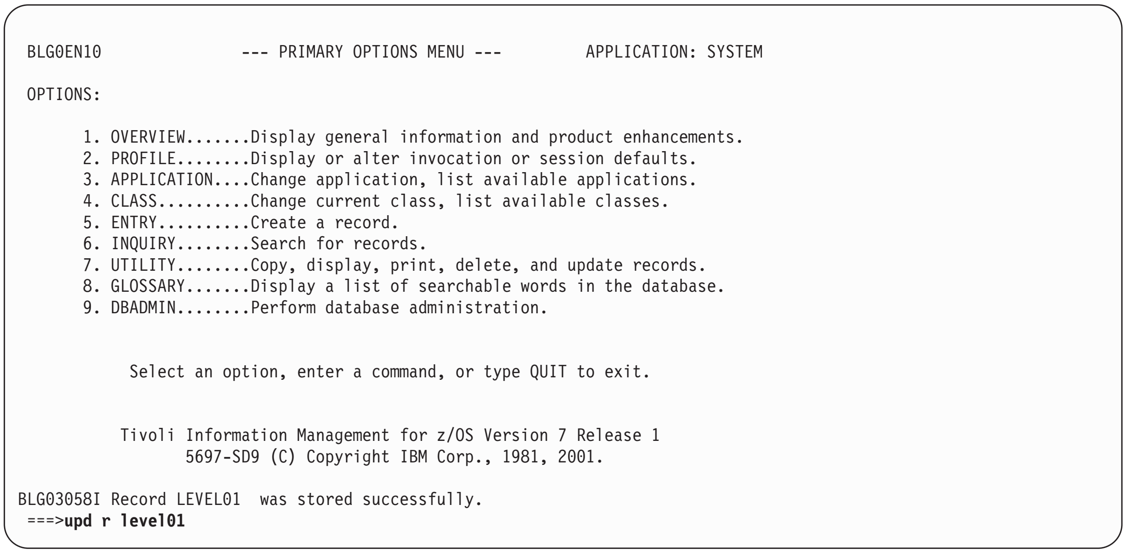
Universal Partition Access Authority

A user can have access to all partitions, and therefore all records in the Tivoli Information Management for z/OS database, when using a privilege class that has Universal Partition Access authority. A record entered by a user using a privilege class that has this authority is assigned the Primary Partition Name from the privilege class as its Owning Partition Name.

However, record commands and utility functions such as UPDATE or DISPLAY access all records in the data base. Because it is possible to have duplicate RNIDs across partitions, a user with Universal Partition Access authority can only access a record with a duplicate RNID by building a search results list that lists the record and working with the record from that list.

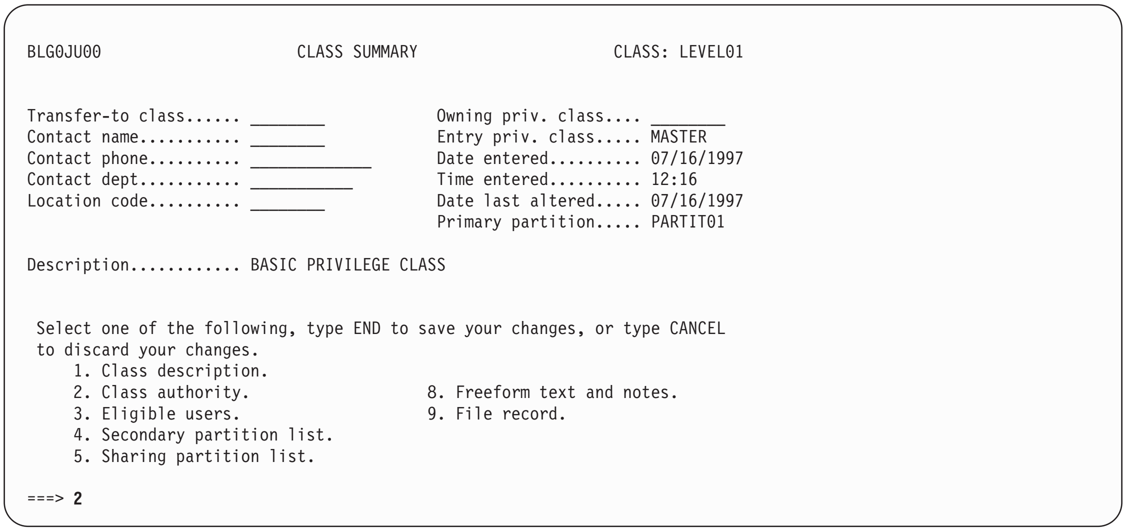
The following panels illustrate how to modify a privilege class record to assign it Universal Partition Access authority.

Update the Privilege Class record which you just created. Type **upd r level01** and press **Enter**.



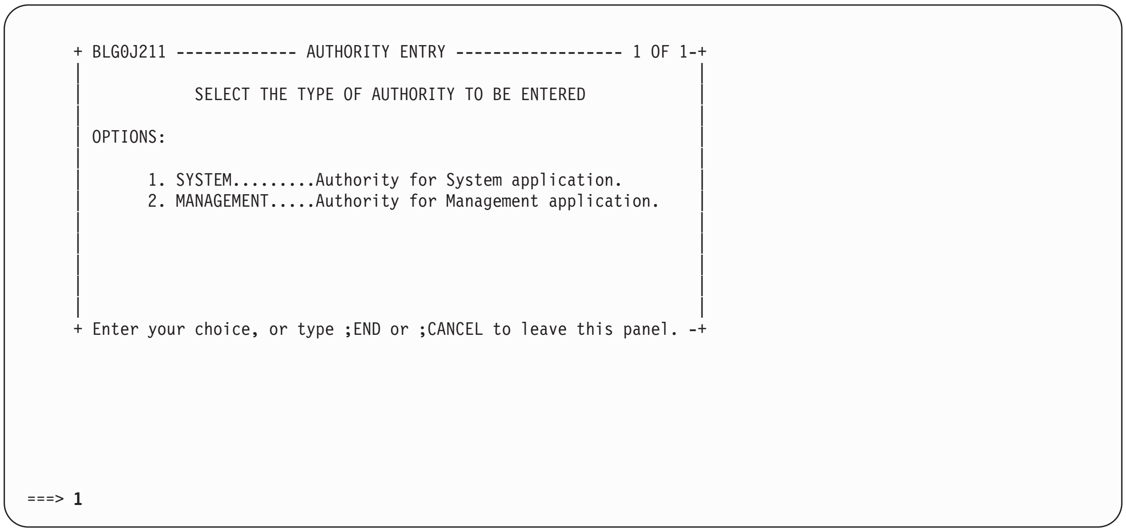
```
 BLG0EN10               --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

 OPTIONS:

       1. OVERVIEW.......Display general information and product enhancements.
       2. PROFILE........Display or alter invocation or session defaults.
       3. APPLICATION....Change application, list available applications.
       4. CLASS..........Change current class, list available classes.
       5. ENTRY..........Create a record.
       6. INQUIRY........Search for records.
       7. UTILITY........Copy, display, print, delete, and update records.
       8. GLOSSARY.......Display a list of searchable words in the database.
       9. DBADMIN........Perform database administration.


            Select an option, enter a command, or type QUIT to exit.


           Tivoli Information Management for z/OS Version 7 Release 1
                  5697-SD9 (C) Copyright IBM Corp., 1981, 2001.

BLG03058I Record LEVEL01  was stored successfully.
 ===>upd r level01
```

From the Class Summary panel, type **2** and press **Enter**.



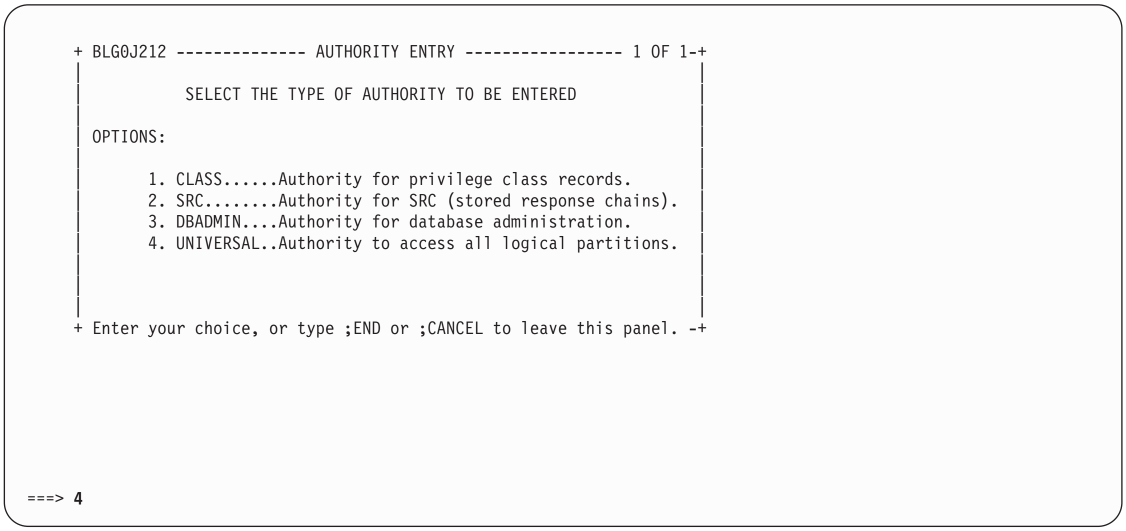
```
BLG0JU00                    CLASS SUMMARY                    CLASS: LEVEL01


Transfer-to class...... ________           Owning priv. class.... ________
Contact name........... ________           Entry priv. class..... MASTER
Contact phone.......... _____________      Date entered.......... 07/16/1997
Contact dept........... ___________        Time entered.......... 12:16
Location code.......... ________           Date last altered..... 07/16/1997
                                           Primary partition..... PARTIT01

Description............ BASIC PRIVILEGE CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.
     2. Class authority.                  8. Freeform text and notes.
     3. Eligible users.                   9. File record.
     4. Secondary partition list.
     5. Sharing partition list.

===> 2
```

Because this is a System Authority, type **1** and press **Enter**.



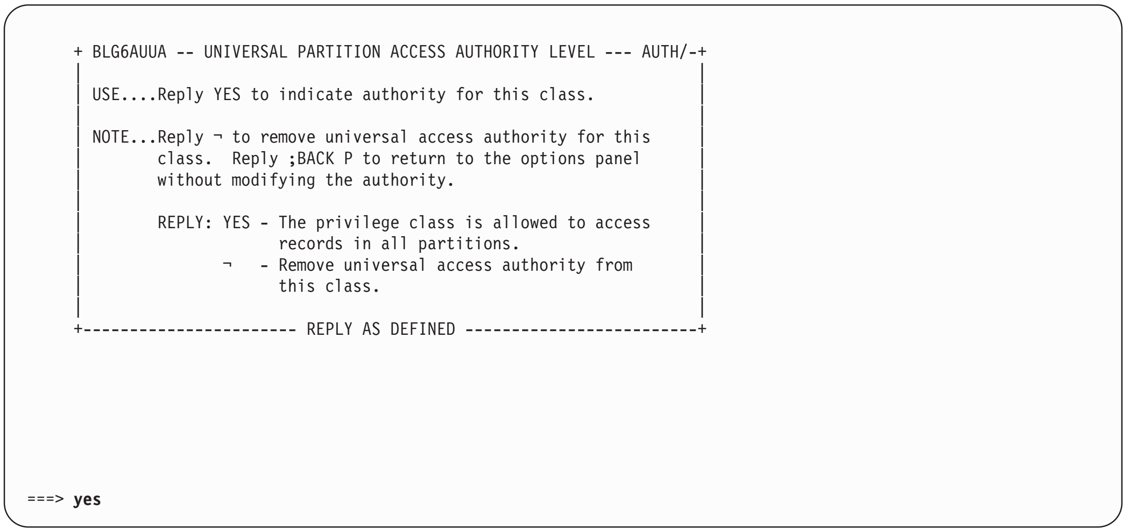
```
     + BLG0J211 ------------- AUTHORITY ENTRY ------------------ 1 OF 1-+
     |                                                                  |
     |            SELECT THE TYPE OF AUTHORITY TO BE ENTERED            |
     |                                                                  |
     | OPTIONS:                                                         |
     |                                                                  |
     |       1. SYSTEM.........Authority for System application.        |
     |       2. MANAGEMENT.....Authority for Management application.    |
     |                                                                  |
     |                                                                  |
     |                                                                  |
     |                                                                  |
     |                                                                  |
     + Enter your choice, or type ;END or ;CANCEL to leave this panel. -+






===> 1
```

On the Authority Entry panel, select **UNIVERSAL**. Type **4** and press **Enter**.



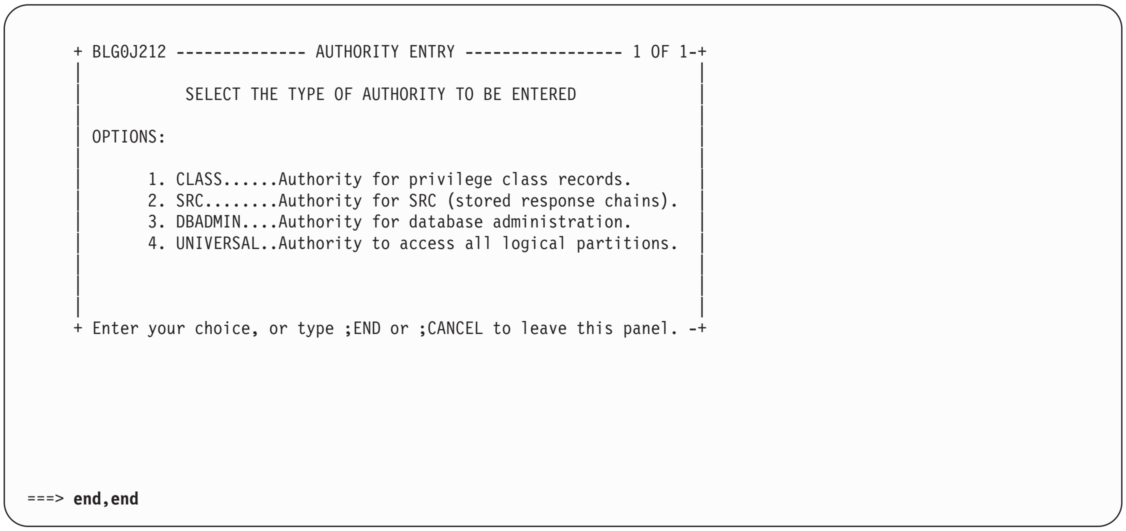
```
     + BLG0J212 -------------- AUTHORITY ENTRY ----------------- 1 OF 1-+
     |                                                                  |
     |           SELECT THE TYPE OF AUTHORITY TO BE ENTERED             |
     |                                                                  |
     | OPTIONS:                                                         |
     |                                                                  |
     |       1. CLASS......Authority for privilege class records.       |
     |       2. SRC........Authority for SRC (stored response chains).  |
     |       3. DBADMIN....Authority for database administration.       |
     |       4. UNIVERSAL..Authority to access all logical partitions.  |
     |                                                                  |
     |                                                                  |
     |                                                                  |
     + Enter your choice, or type ;END or ;CANCEL to leave this panel. -+






===> 4
```

Now you are ready to give Universal Partition Access authority. Type **YES** and press **Enter**.



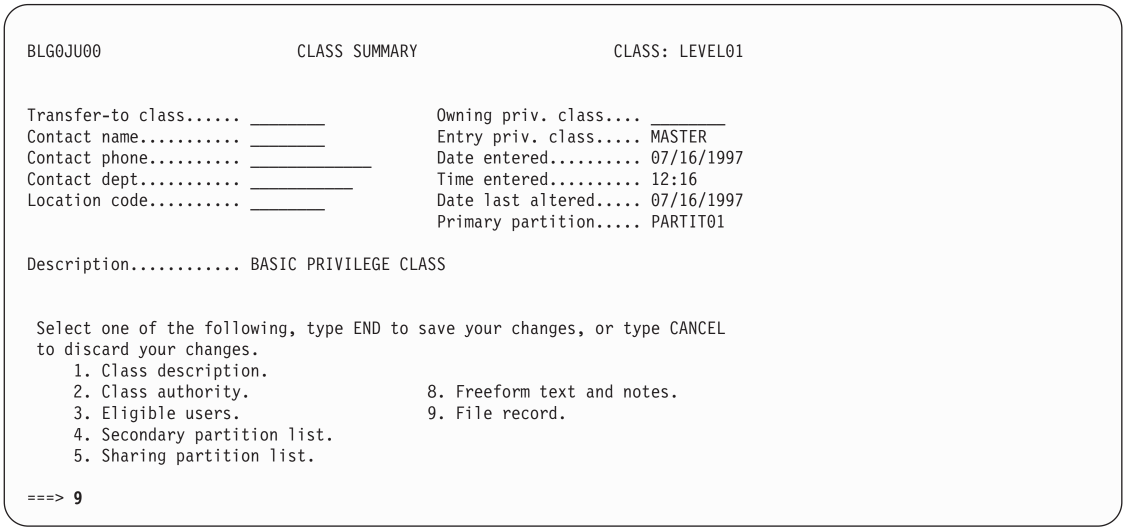
```
     + BLG6AUUA -- UNIVERSAL PARTITION ACCESS AUTHORITY LEVEL --- AUTH/-+
     |                                                                  |
     | USE....Reply YES to indicate authority for this class.           |
     |                                                                  |
     | NOTE...Reply ¬ to remove universal access authority for this     |
     |        class.  Reply ;BACK P to return to the options panel      |
     |        without modifying the authority.                          |
     |                                                                  |
     |        REPLY: YES - The privilege class is allowed to access     |
     |                     records in all partitions.                   |
     |               ¬   - Remove universal access authority from       |
     |                     this class.                                  |
     |                                                                  |
     +---------------------- REPLY AS DEFINED -------------------------+






===> yes
```

You return to the Authority Entry panel. Type **end,end** and press **Enter**.



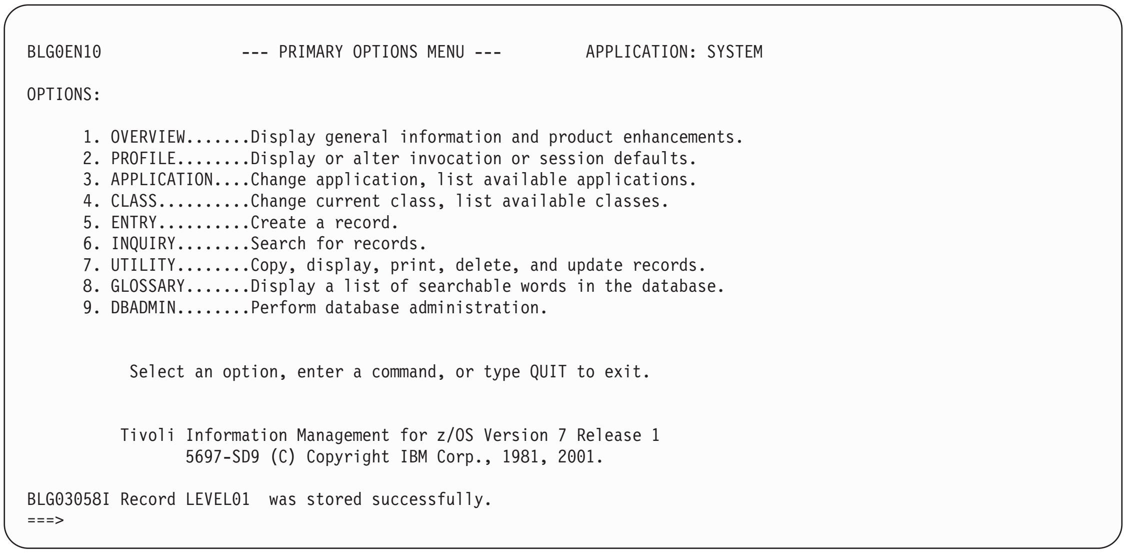
```
     + BLG0J212 -------------- AUTHORITY ENTRY ----------------- 1 OF 1-+
     |                                                                  |
     |          SELECT THE TYPE OF AUTHORITY TO BE ENTERED              |
     |                                                                  |
     | OPTIONS:                                                         |
     |                                                                  |
     |       1. CLASS......Authority for privilege class records.       |
     |       2. SRC........Authority for SRC (stored response chains).  |
     |       3. DBADMIN....Authority for database administration.       |
     |       4. UNIVERSAL..Authority to access all logical partitions.  |
     |                                                                  |
     |                                                                  |
     |                                                                  |
     + Enter your choice, or type ;END or ;CANCEL to leave this panel. -+






===> end,end
```

You return to the Class Summary panel. To save the privilege class record, select **File record**. Type **9** and press **Enter**.



```
BLG0JU00                    CLASS SUMMARY                    CLASS: LEVEL01


Transfer-to class...... ________           Owning priv. class.... ________
Contact name........... ________           Entry priv. class..... MASTER
Contact phone.......... _____________      Date entered.......... 07/16/1997
Contact dept........... ___________        Time entered.......... 12:16
Location code.......... ________           Date last altered..... 07/16/1997
                                           Primary partition..... PARTIT01

Description............ BASIC PRIVILEGE CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.
     2. Class authority.                  8. Freeform text and notes.
     3. Eligible users.                   9. File record.
     4. Secondary partition list.
     5. Sharing partition list.

===> 9
```

A message confirms that record LEVEL01, the privilege class record that you have modified, was stored successfully.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.

BLG03058I Record LEVEL01  was stored successfully.
===>
```

Global Partition

If you have a set of records which you would like to be accessible to all users, put all such records into a “Global” Partition. Records in a Global Partition are accessible by all privilege classes.

To set up a global partition, you must define a Global Partition Name. It can be defined in either of two ways:

- Use program exit BLG01448 (described in *Tivoli Information Management for z/OS Panel Modification Facility Guide*) to define a Global Partition Name.
- Specify a value for the GBLPID keyword on the BLGPARMS macro in a session-parameters member. This keyword, which is optional, defines a Global Partition Name. A value of up to nine alphanumeric characters can be specified for the GBLPID keyword. The *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* contains detailed information on the GBLPID keyword.

These are the rules for specifying a Global Partition Name:

- If a value is not provided for the GBLPID keyword in the session-parameters member and a Global Partition Name is not specified by program exit BLG01448, then no Global Partition exists.
- If a value is specified on the GBLPID keyword, that value is used instead of the value set by program exit BLG01448.
- If a value of GBLPID='' is specified, the Global Partition Name established by program exit BLG01448 is ignored.
- If the GBLPID keyword is omitted, the Global Partition Name set by program exit BLG01448 is used.

Once the Global Partition Name is defined, it can be assigned to a record as the Owning Partition in either of two ways:

- Add records to the database using a privilege class that has the Global Partition Name as its Primary Partition Name.
- Use program exit BLG01449 (described in the *Tivoli Information Management for z/OS Panel Modification Facility Guide*) to assign the Global Partition as the Owning Partition at the time the record is entered.

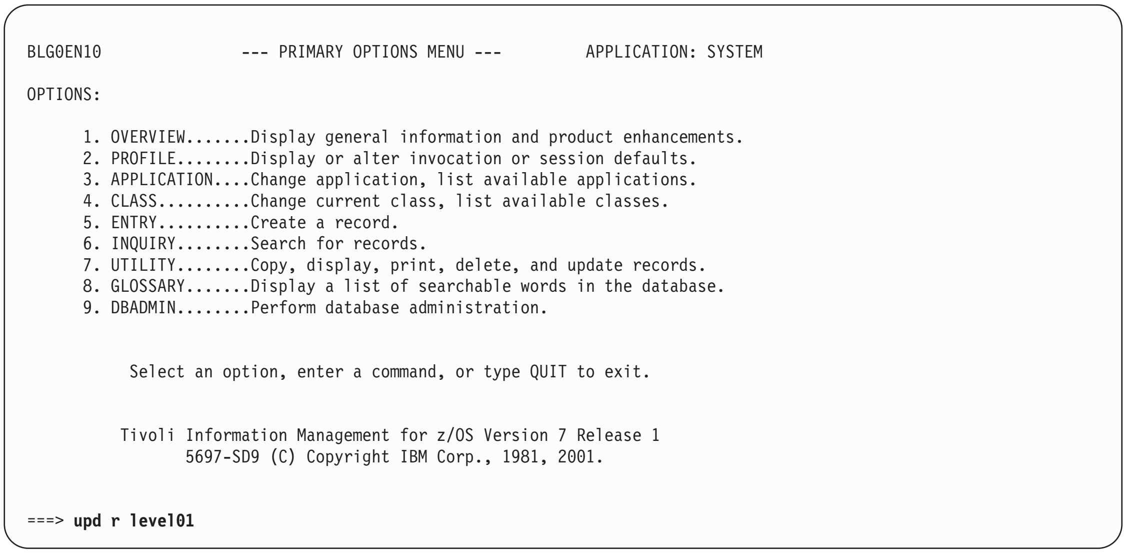
Sharing Partition List

The Owning Partition, once assigned to a record, must not be changed. Only if you have that record's Owning Partition Name as the Primary Partition Name in your privilege class can you use record access commands or utility functions to access the record. If you have the record's Owning Partition listed as a Secondary Partition in your privilege class, you can access the record via a search results list. If you also have the need to allow other partitions to have access to an individual record, you can set up a Sharing Partition List for that record. This Sharing Partition List permits an individual record to be shared by multiple partitions in addition to that identified by the record's Owning Partition Name. When the Primary Partition Name in your privilege class is listed in the Sharing Partition List for an individual record that is owned by a different partition, you have access to that record as though it were in your primary partition.

Tivoli Information Management for z/OS allows you to enter a list of Sharing Partitions for privilege class records. If you need to be able to enter Sharing Partitions for other types of records, you will need to use the Panel Modification Facility (PMF) to customize your panels. "PMF Considerations" on page 220 provides some characteristics about Sharing Partitions that you will need to know before you customize your system.

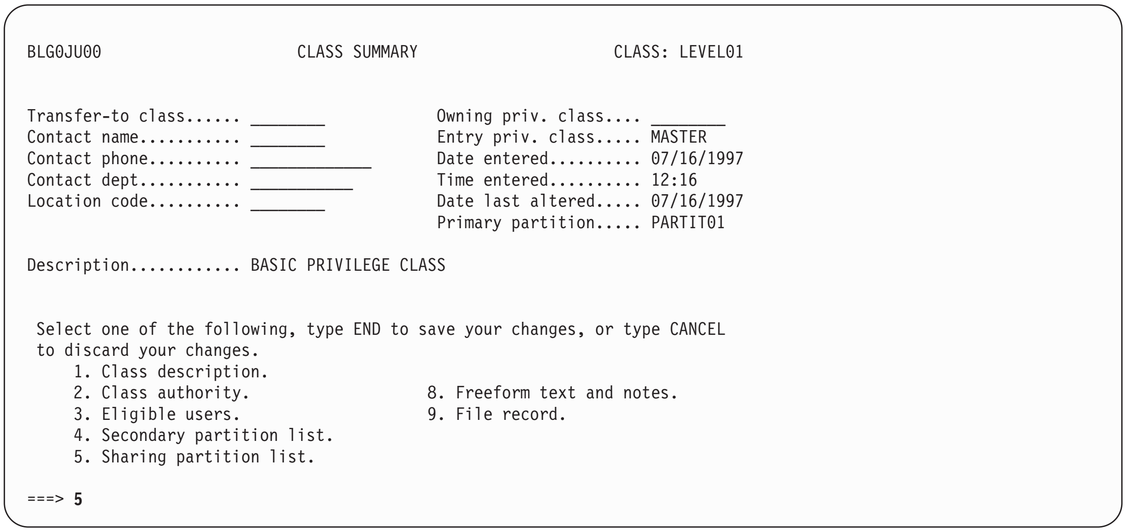
The following example shows you how to add Sharing Partitions to a privilege class record.

Update the privilege class record with which we have been working by typing **upd r level01** and press **Enter**.



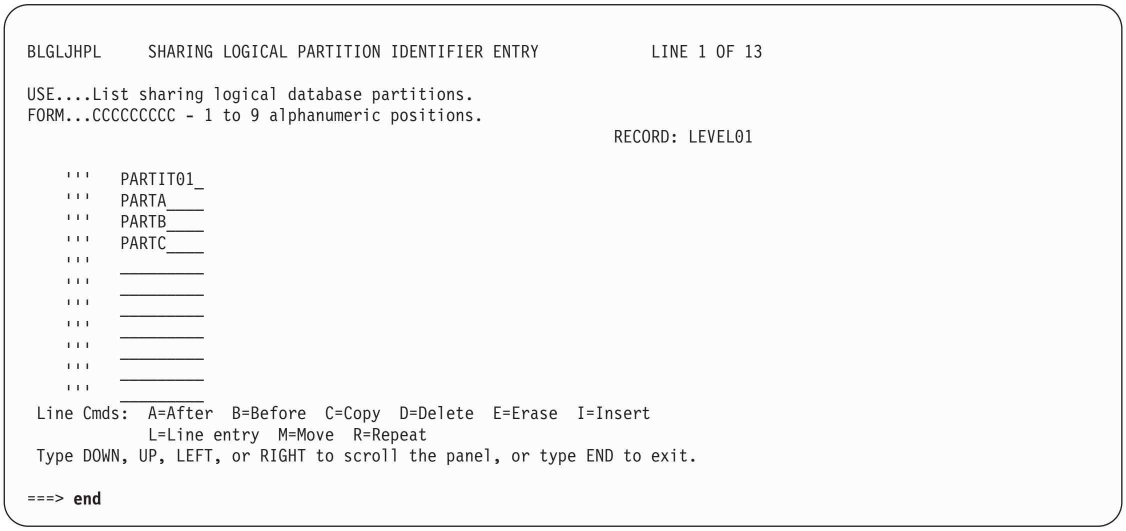
```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


===> upd r level01
```

To create a Sharing Partition List which identifies other partitions that share this record, type **5** and press **Enter**.



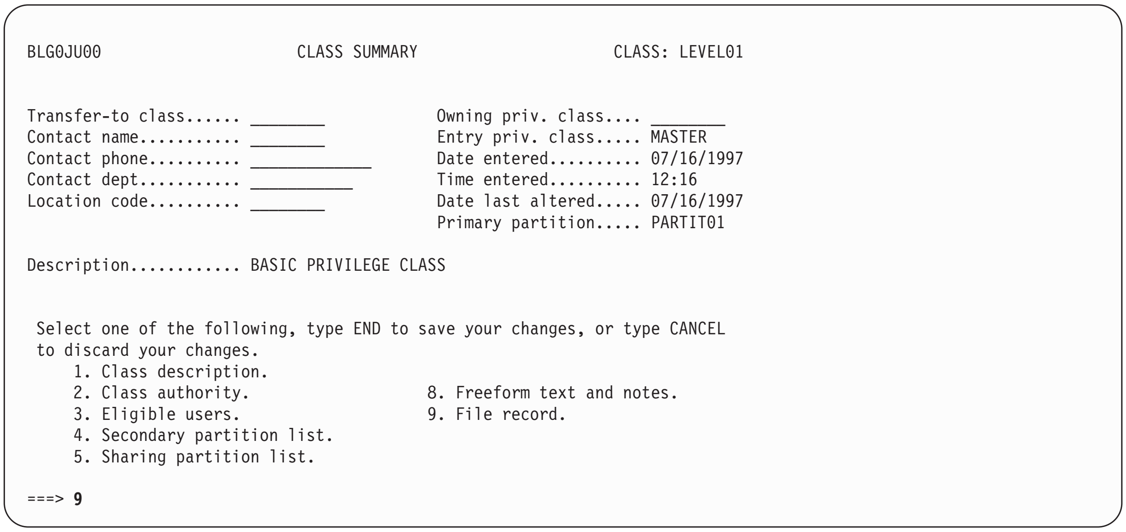
```
BLG0JU00                    CLASS SUMMARY                    CLASS: LEVEL01


Transfer-to class...... ________              Owning priv. class.... ________
Contact name........... ________              Entry priv. class..... MASTER
Contact phone.......... _____________         Date entered.......... 07/16/1997
Contact dept........... ___________           Time entered.......... 12:16
Location code.......... ________              Date last altered..... 07/16/1997
                                              Primary partition..... PARTIT01

Description............ BASIC PRIVILEGE CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.
     2. Class authority.                 8. Freeform text and notes.
     3. Eligible users.                  9. File record.
     4. Secondary partition list.
     5. Sharing partition list.

===> 5
```

For privilege class records, the Primary Partition Name from the privilege class being updated or created is automatically added to the Sharing Partition List. You can enter the names of additional partitions that share this record. In this example, PARTA, PARTB, and PARTC are added to the Sharing Partition list. When you have completed the list, type **end** and press **Enter**.



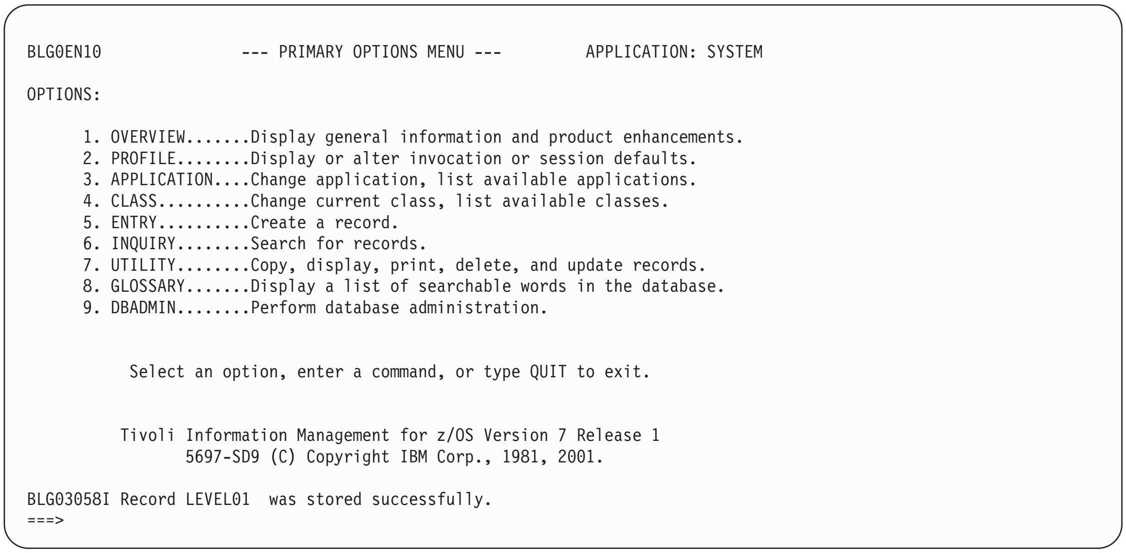
```
BLGLJHPL     SHARING LOGICAL PARTITION IDENTIFIER ENTRY           LINE 1 OF 13

USE....List sharing logical database partitions.
FORM...CCCCCCCCC - 1 to 9 alphanumeric positions.
                                                              RECORD: LEVEL01

    '''   PARTIT01_
    '''   PARTA____
    '''   PARTB____
    '''   PARTC____
    '''   _________
    '''   _________
    '''   _________
    '''   _________
    '''   _________
    '''   _________
    '''   _________
 Line Cmds:  A=After  B=Before  C=Copy  D=Delete  E=Erase  I=Insert
             L=Line entry  M=Move  R=Repeat
 Type DOWN, UP, LEFT, or RIGHT to scroll the panel, or type END to exit.

===> end
```

The Class Summary panel summarizes the information about this privilege class record. To save the privilege class record, select **File record**. Type **9** and press **Enter**.



```
BLG0JU00                    CLASS SUMMARY                    CLASS: LEVEL01


Transfer-to class...... ________            Owning priv. class.... ________
Contact name........... ________            Entry priv. class..... MASTER
Contact phone.......... _____________       Date entered.......... 07/16/1997
Contact dept........... ___________         Time entered.......... 12:16
Location code.......... ________            Date last altered..... 07/16/1997
                                            Primary partition..... PARTIT01

Description............ BASIC PRIVILEGE CLASS


 Select one of the following, type END to save your changes, or type CANCEL
 to discard your changes.
     1. Class description.
     2. Class authority.                   8. Freeform text and notes.
     3. Eligible users.                    9. File record.
     4. Secondary partition list.
     5. Sharing partition list.

===> 9
```

A message confirms that record LEVEL01, the record for which you have added Sharing Partition information, was stored successfully.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.

BLG03058I Record LEVEL01  was stored successfully.
===>
```

Logical Database Partitioning Considerations

Unique RNIDs by Partition

There is a record in each partition which contains the last system-assigned record number (also referred to as the last entry number) for that partition. This allows a unique value to be specified for the last entry number for each partition. There is also a record which contains the last entry number for non-partitioned records. System-assigned RNIDs are assigned based on the appropriate last entry number record from the SDIDS. If the user's privilege class has a Primary Partition Name, the partition's last entry number record is used when assigning a system-assigned RNID. Otherwise, the database last entry number record is used.

For user-assigned RNIDs, Tivoli Information Management for z/OS checks only the Primary and Global Partitions for duplicate RNIDs.

Record Searches

In general, when a user issues the SEARCH command, only records in the user's primary partition are returned. If a Global Partition is defined, then the records owned by the Global Partition are also returned. In addition, if you have customized Tivoli Information Management for z/OS to collect Sharing Partitions for individual records, those record are returned if you are authorized.

A user's privilege class also identifies secondary partitions that the user has authority to access. Records in secondary partitions are only accessible from a search results list. Databases 5, 7, 8, and 9 support secondary partitions. To add secondary partitions to a search, you must use the SEARCH command and specify the **=n*** operand (where **n** is the database identifier; for example, **=5***). The **=n*** operand on the SEARCH command causes all of the secondary partitions to which you have access to be searched.

Program exit BLG01450 (described in the *Tivoli Information Management for z/OS Panel Modification Facility Guide*) can be used to force the next search operation to be a multipartition search.

Note: API applications cannot perform multipartition searches because an API is unable to retrieve a record by its VSAM root key.

Because a search can access multiple partitions, and because it is possible to have records in your search results list with duplicate RNIDs, you may want to know the partition to which a particular record belongs before you access that record from a search results list. Your Search Results List panels can be customized to show the Owning Partition Name for each record in the list. A Search Results List panel BLG1TSRP that displays the Owning Partitions Names is shipped by Tivoli Information Management for z/OS as an example.

Copying Records between Partitions

When a record is copied from one partition to another, the Owning Partition Name in the copied record is set to the Primary Partition Name from the current privilege class. You can use program exit BLG01151 to delete the Sharing Partition List from a record when it is copied by adding the list processor root s-word associated with the Sharing Partition List to BLG1ACOP. "PMF Considerations" on page 220 contains the Sharing Partition List root s-word.

Record Ownership

Each record in the database contains three privilege class names: the Entry Class, the Owning Class, and the Transfer-to Class.

The Owning Class field in a record describes its privilege class ownership. Record ownership can be transferred across partitions by changing the **Transfer-to Class** field in the record. In order for the Transfer-to Class receiver to be able to access the record, the record whose ownership is being changed must contain the Primary Partition Name of the Transfer-to Class receiver either as its Owning Partition Name or as a member of its Sharing Partition List. In addition, the privilege class of the Transfer-to Class receiver must contain the Owning Partition Name of the record either as its Primary Partition or as a member of its Secondary Partition List.

It is not possible to differentiate between like-named privilege classes in different partitions.

Commands in a Logical Partition Environment

Although you cannot have records with duplicate RNIDs within a partition, records in different logical partitions may have duplicate RNIDs. If you are authorized by your privilege class (by Universal Partition Access or by Sharing Partitions) to records in more than one partition that have duplicate RNIDs, you cannot use Tivoli Information Management for z/OS record commands or utility functions (such as UPDATE or DISPLAY) to access these records. However, those records are accessible from a search results list. A search that includes secondary partitions may also result in a search results list that has records with duplicate RNIDs. Line commands can be used to access any record in the search results list.

The COPY, DISPLAY, PRINT, DELETE, and UPDATE Commands

The COPY, DISPLAY, PRINT, DELETE, and UPDATE commands retrieve the target record by issuing a search for the specified RNID. Only records with unique RNIDs can be processed by these commands and their corresponding utility functions.

In general, these commands access records in the user's primary partition (and Global Partition), in which case your records will have unique RNIDs. If your privilege class has Universal Partition Access authority or you use Sharing Partitions, it may be possible for you to use these commands or utility functions to access records with unique RNIDs in other partitions. If you have access to records with duplicate RNIDs, you must use a search results list in order to access these records.

The GLOSSARY Command

The GLOSSARY command shows only information relevant to the Primary, Secondary, and Global Partitions, unless the current privilege class has Universal Partition Access or MASTER authority. MASTER authority is required to display the last entry number values, including the value for LASTENTRYNUMBER.

The ORDER Command

Because Tivoli Information Management for z/OS does not apply database partitioning to the ORDER command, you should use the command alias and authorization function (described in “Defining Command Aliases and Authorization” on page 147) to restrict its usage.

The STATISTICS Command

Because Tivoli Information Management for z/OS does not apply database partitioning to the STATISTICS command, you should use the command alias and authorization function (described in “Defining Command Aliases and Authorization” on page 147) to restrict its usage.

Migrating Existing Records to a Logical Partition

Existing records can be logically partitioned via a user-written TSP.

Migrating Records Using a TSP

The TSP must perform the following processing:

- Execute a search to create a Search Results List (SRL) containing records that are to be grouped into a logical partition.
- Block update the records in the SRL.
- Use the TSP WORDFIX control line to add the Owning Partition s-word and p-word, including the partition name, to each record in the list.

  Note: WORDFIX is not supported in a TSX.

For additional information on how to add data to a record using the TSP WORDFIX control line, refer to the *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference*. An example of how to perform a search and block update of a group of records can also be found in that document.

Note: You can damage your existing database if you do not use the WORDFIX control line correctly. For information on the security measures you can use to protect against its misuse, refer to the discussion of data integrity and security using TSPs in the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* .

Each partition will have a record in the SDIDS, similar to the database LASTENTRYNUMBER record, to keep track of the last system-assigned RNID in that partition. The key of the last entry number for a partition has the following format:

```
PTID/ptidname
```

When the first record with a system-assigned RNID is filed in a partition and the last entry number record for the partition does not exist in the SDIDS, the following occurs:

- A last entry number record is created for the partition.
- The initial RNID for the partition last entry number record will be 00000001 if there is no record in the SDIDS with that PTID/ptidname or there is no database LASTENTRYNUMBER record. Otherwise it will be the value of LASTENTRYNUMBER. A user with MASTER authority can use the GLOSSARY command to inspect the last entry number records for the database and its partitions if they exist.

Migrating Records Using FLATTEN and UNFLATTEN

You can also migrate records using the FLATTEN and UNFLATTEN control lines. If the privilege class of the user performing the UNFLATTEN specifies a primary partition, the Primary Partition Name becomes the Owning Partition Name of the unflattened records. If the user's privilege class does not contain a Primary Partition Name, the unflattened records will not be assigned an Owning Partition Name. The *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference* provides additional information about the FLATTEN and UNFLATTEN control lines.

PMF Considerations

If you are using logical database partitioning, you should know that the p-word PTID/ is used by Tivoli Information Management for z/OS to designate partition identifiers. Your program administrator should check your dictionary to see if your installation is using PTID/. If so, your administrator should use program exit BLG01448 to change the default p-word for the partition.

The following table contains a list of the s-words and p-words (and their major characteristics) associated with Tivoli Information Management for z/OS logical database partitioning.

Table 35.

Field Name	s-word	s-word index	p-word	p-word index	prefix validation pattern
Owning Partition	XIM0I0CPO	X'0174'	PTID/	X'03C9'	
Sharing Partition List	XIM0I0CH	X'011D'	PTID/	X'011D'	SSV8
Primary Partition	XIM0I0CPP	X'011B'		X'03C7'	SSV8
Secondary Partition List	XIM0I0CS	X'0118'		X'03C7'	SSV8

Predefined Validation Patterns

Equal sign response validation patterns that are available are described in the *Tivoli Information Management for z/OS Panel Modification Facility Guide*.

Archiver Considerations

If you are using the Archiver function (described in “The Tivoli Information Management for z/OS Archiver” on page 337), you must be aware of the restrictions described in “Logical Database Partition Considerations” on page 363.

12

Performing Database Administration

As the program administrator, you need database administration authority. This authority allows you to perform database administration tasks, which include DBCLEANUP, CHECK IN, and Tivoli Service Desk Bridge Cleanup (TSD Bridge Cleanup). It also allows you to create, update, and delete certain Tivoli Information Management for z/OS records that have special functions like the ALIAS and data model records.

Data model records, which include validation, data view and data attribute records, are described in *Tivoli Information Management for z/OS Panel Modification Facility Guide*. The LOGSAVE record is described in the Data Propagation Tasks section in “Propagating Tivoli Information Management for z/OS Data” on page 231. The ALIAS record is described in “Defining User Line Commands” on page 137. The COMMAND record is described in “Defining Command Aliases and Authorization” on page 147. The remainder of this chapter describes the DBCLEANUP, CHECK IN, and TSD Bridge Cleanup tasks.

If a system interrupt or ABEND occurs while a user is filing a record, the record file process may be unsuccessful. The record may be only partially filed, or it may be marked as being either busy or deleted from the database. You can restore these records using DBCLEANUP.

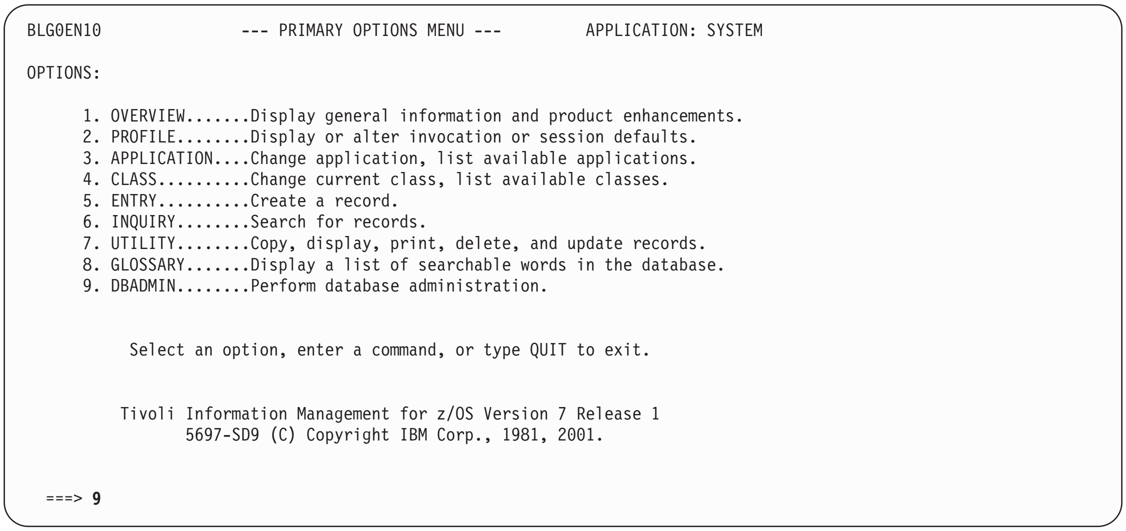
Applications running with the Tivoli Information Management for z/OS application program interfaces can access records in the Tivoli Information Management for z/OS database. It is possible for an application to check out a record from the database and fail to check it back in when processing is complete. You can use the CHECK IN option of database administration to check in these records to the database. You perform CHECK IN by requesting a list of checked out records and selecting from that list the records to be checked in. You can obtain a list of all the records checked out of a database or a list of all the records checked out to an application.

If you are using the Tivoli Service Desk Bridge program (described in the *Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications*), records that are pending some type of processing must be locked until the processing is complete. The record locking is automatic, with no external interface. A Tivoli Information Management for z/OS cleanup function is provided to permit a database administrator to unlock records pending completion of processing. This function would be used if the record seems to be permanently locked pending completion of a transaction.

Performing DBCLEANUP

DBCLEANUP attempts to restore any partially filed records that are marked busy or deleted as a result of errors found in a previous file process. DBCLEANUP calls the BLG01336 program exit, which attempts to file any partially filed records. In addition, this program exit attempts to finish filing any records marked busy or partially deleted. If the file process on the busy or partially deleted records is not completed using DBCLEANUP, you can run the BLGUT1 utility, which is documented in the *Tivoli Information Management for z/OS Operation and Maintenance Reference*.

From the Primary Options Menu for the System application, select **DBADMIN**. Type **9** and press **Enter**.

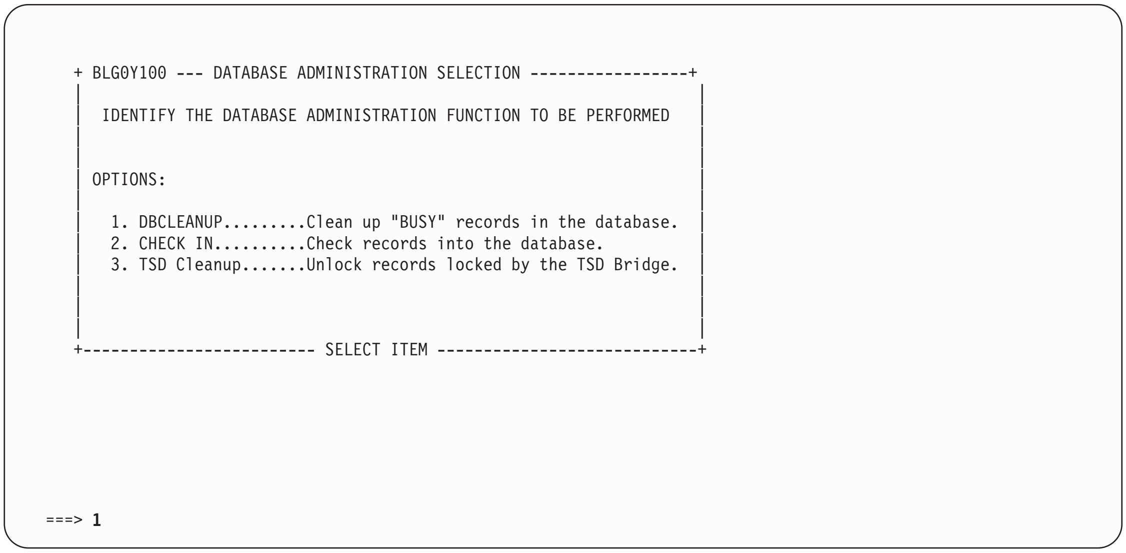


```
BLG0EN10              --- PRIMARY OPTIONS MENU ---         APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


  ===> 9
```

The Database Administration Selection panel appears. Select **DBCLEANUP**. Type **1** and press **Enter**.

Any partially filed records are processed. A message indicates how many records were processed successfully and how many failed. To find out how to clean up any records that failed, refer to the *Tivoli Information Management for z/OS Diagnosis Guide*.

Type **INIT** and press **Enter** to return to the Primary Options Menu.



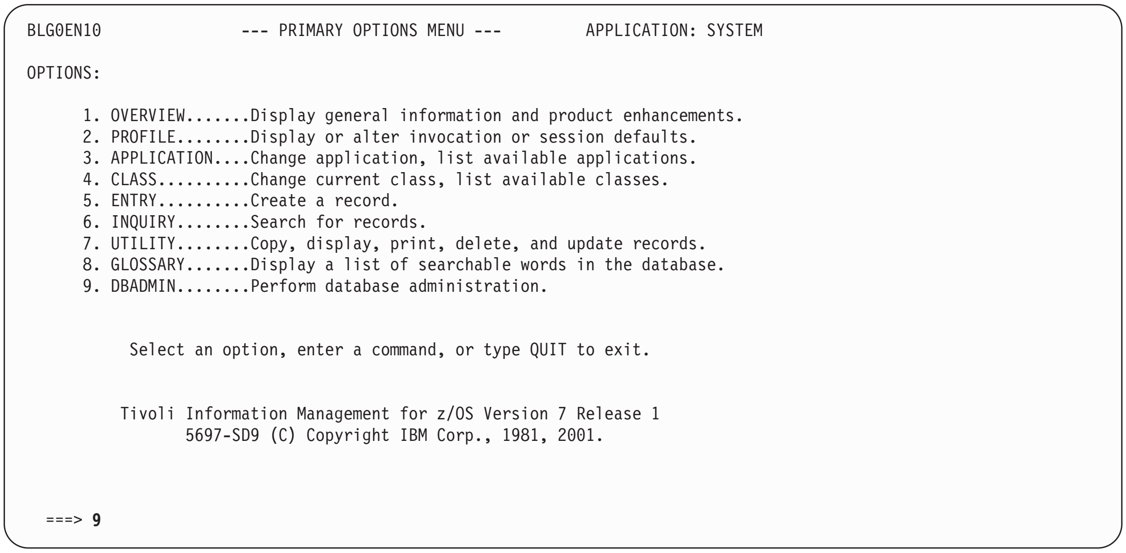
```
   + BLG0Y100 --- DATABASE ADMINISTRATION SELECTION -----------------+
   |                                                                  |
   |  IDENTIFY THE DATABASE ADMINISTRATION FUNCTION TO BE PERFORMED   |
   |                                                                  |
   |                                                                  |
   | OPTIONS:                                                         |
   |                                                                  |
   |   1. DBCLEANUP.........Clean up "BUSY" records in the database. |
   |   2. CHECK IN..........Check records into the database.          |
   |   3. TSD Cleanup.......Unlock records locked by the TSD Bridge.  |
   |                                                                  |
   |                                                                  |
   |                                                                  |
   +------------------------- SELECT ITEM ----------------------------+







===> 1
```

Performing CHECK IN

CHECK IN is used to check in to the database records that have been checked out by an application running with an application program interface. When an application running with an application program interface checks out a record, no other application can use that record. If this application fails to check in the record, you should then use CHECK IN.

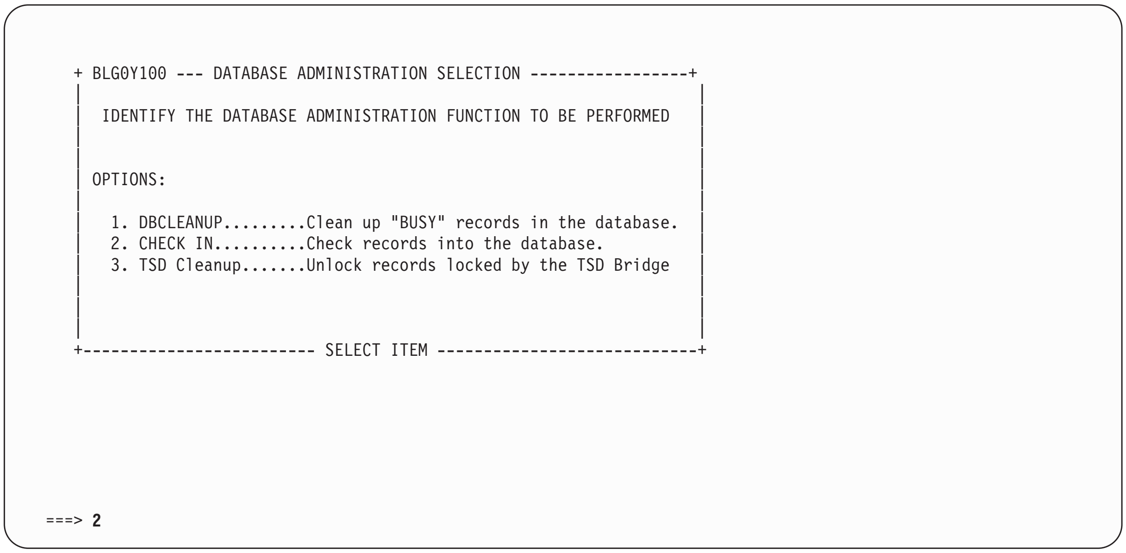
With database administration authority you can obtain a list of checked out records and select records to be checked in from the list. A Checked Out Record list is compiled based on the data you enter on the Check In panel. You can request a list of all records checked out of a database, or you can ask for a list of all records checked out to an application ID. If you know that an application has not checked in a specific record, you can enter the record's ID on the Check In panel. The Checked Out Record list is displayed, and by selecting the record from the list, you check it in.

On the Primary Options Menu for the System application, select **DBADMIN**. Type **9** and press **Enter**.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.



  ===> 9
```

On BLG0Y100, the Database Administration Selection panel, select **CHECK IN**. Type **2** and press **Enter**.

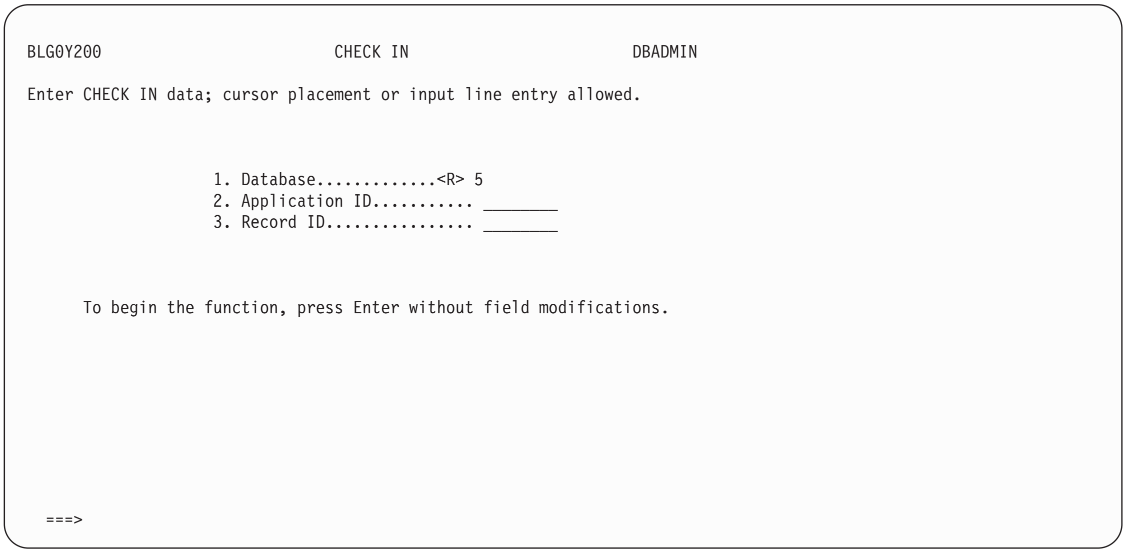


```
     + BLG0Y100 --- DATABASE ADMINISTRATION SELECTION -----------------+
     |                                                                  |
     |  IDENTIFY THE DATABASE ADMINISTRATION FUNCTION TO BE PERFORMED   |
     |                                                                  |
     |                                                                  |
     | OPTIONS:                                                         |
     |                                                                  |
     |   1. DBCLEANUP.........Clean up "BUSY" records in the database.  |
     |   2. CHECK IN..........Check records into the database.          |
     |   3. TSD Cleanup.......Unlock records locked by the TSD Bridge   |
     |                                                                  |
     |                                                                  |
     |                                                                  |
     +------------------------ SELECT ITEM ----------------------------+




  ===> 2
```

On BLG0Y200, the Check In panel, you must specify a database. The **Database** field is primed with the database number in your profile. If you press **Enter** after completing this field, the Checked Out Record list includes all records checked out of the specified database.

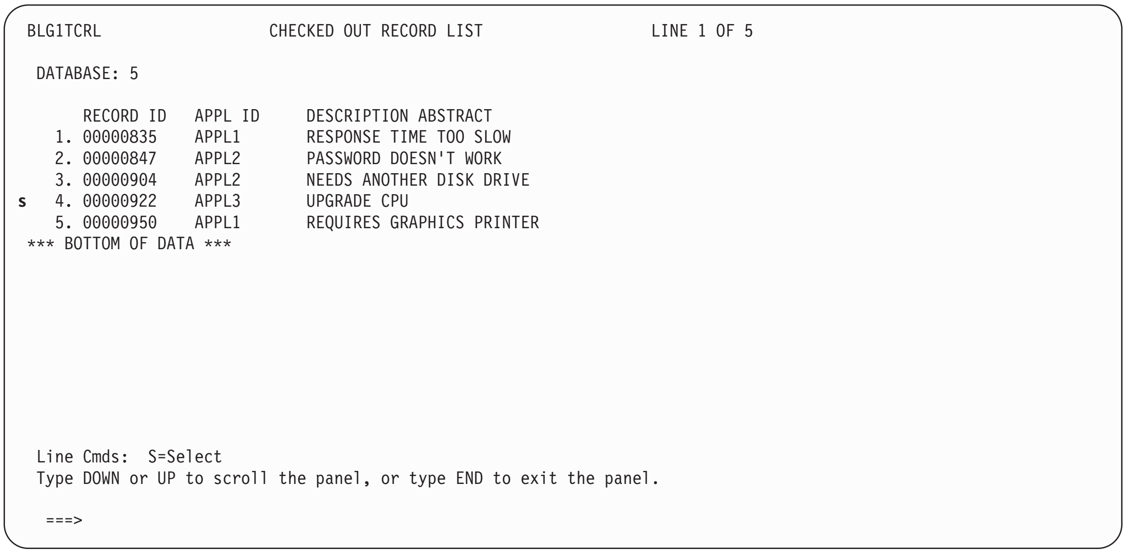
To obtain a list of records checked out to a specific application, put the database number in the Database field and type the application **ID** in the Application ID field.

To restrict the list to a specific record, put the database number in the Database field and type the record **ID** in the Record ID field.



```
BLG0Y200                        CHECK IN                        DBADMIN

Enter CHECK IN data; cursor placement or input line entry allowed.



                   1. Database............<R> 5
                   2. Application ID........... ________
                   3. Record ID................ ________



      To begin the function, press Enter without field modifications.








  ===>
```

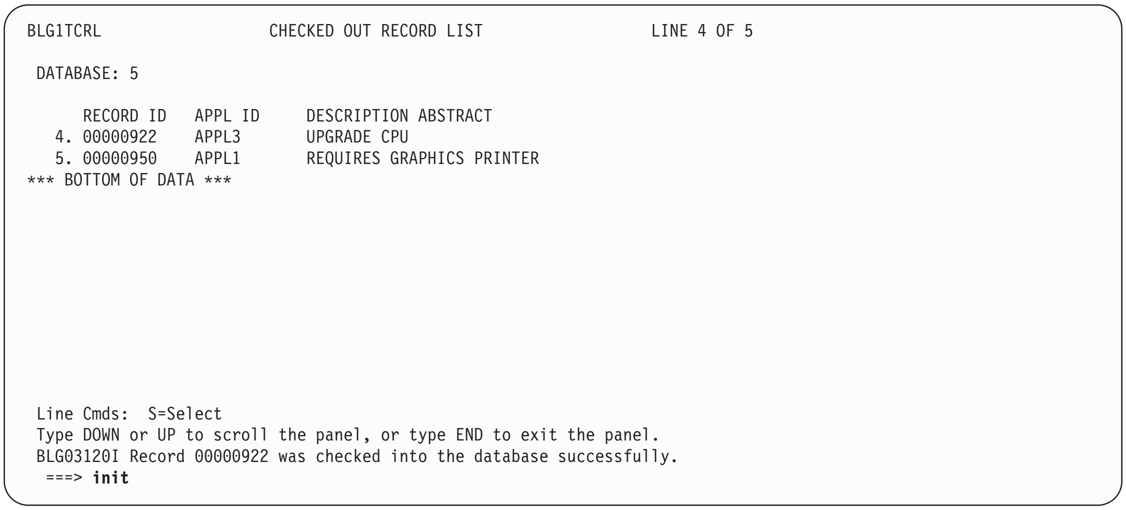
On BLG1TCRL, the Checked Out Record List panel, select the record to be checked in. Move the cursor to the line command area next to the record ID that you want to check in. Type **s** and press **Enter**.



```
BLG1TCRL                 CHECKED OUT RECORD LIST                 LINE 1 OF 5

 DATABASE: 5

      RECORD ID   APPL ID     DESCRIPTION ABSTRACT
   1. 00000835    APPL1       RESPONSE TIME TOO SLOW
   2. 00000847    APPL2       PASSWORD DOESN'T WORK
   3. 00000904    APPL2       NEEDS ANOTHER DISK DRIVE
s  4. 00000922    APPL3       UPGRADE CPU
   5. 00000950    APPL1       REQUIRES GRAPHICS PRINTER
*** BOTTOM OF DATA ***









 Line Cmds:  S=Select
 Type DOWN or UP to scroll the panel, or type END to exit the panel.

  ===>
```

A message that the record has been checked in appears.

To return to the Primary Options menu, type **init** and press **Enter**r.



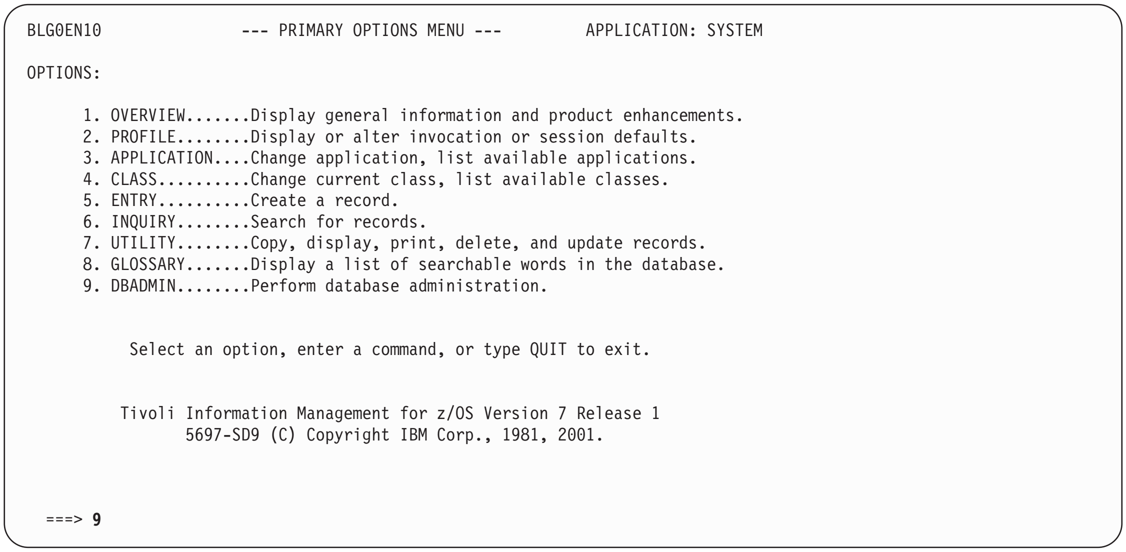
```
BLG1TCRL                 CHECKED OUT RECORD LIST                 LINE 4 OF 5

 DATABASE: 5

      RECORD ID   APPL ID     DESCRIPTION ABSTRACT
   4. 00000922    APPL3       UPGRADE CPU
   5. 00000950    APPL1       REQUIRES GRAPHICS PRINTER
*** BOTTOM OF DATA ***










 Line Cmds:  S=Select
 Type DOWN or UP to scroll the panel, or type END to exit the panel.
 BLG03120I Record 00000922 was checked into the database successfully.
  ===> init
```

Performing Tivoli Service Desk Cleanup

The Tivoli Service Desk Bridge provides a means of exchanging problem records between Tivoli Information Management for z/OS and Tivoli Service Desk. The *Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications* provides detailed information about using the Tivoli Service Desk Bridge to exchange information between Tivoli Information Management for z/OS and Tivoli Service Desk.

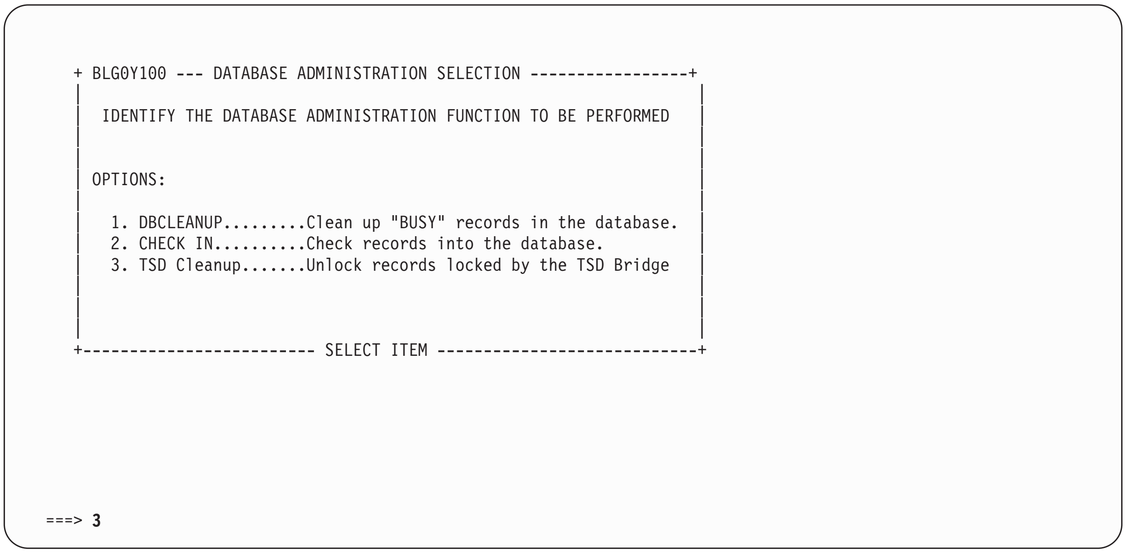
When one of the functions of the Tivoli Service Desk Bridge is being used, the record in the TSD390 database is locked until the processing is complete. The record locking is automatic, with no external interface. A cleanup function, the TSD Bridge Cleanup function, is available to the database administrator and can be used if the record seems to be permanently locked.

On the Primary Options Menu for the System application, select **DBADMIN**. Type **9** and press Enter.



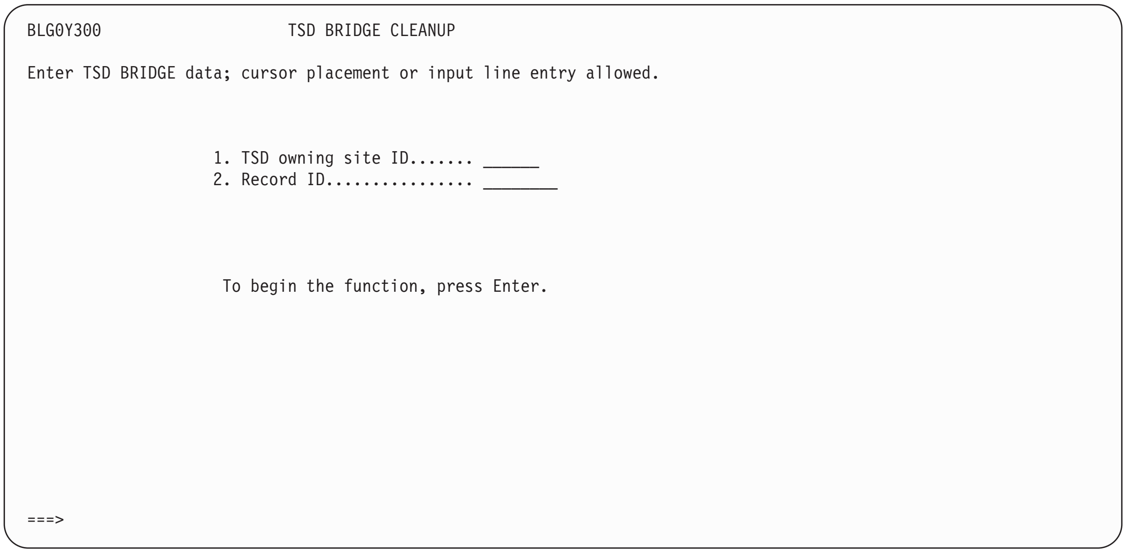
```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.



  ===> 9
```

On BLG0Y100, the Database Administration Selection panel, select **TSD Cleanup**. Type **3** and press **Enter**.



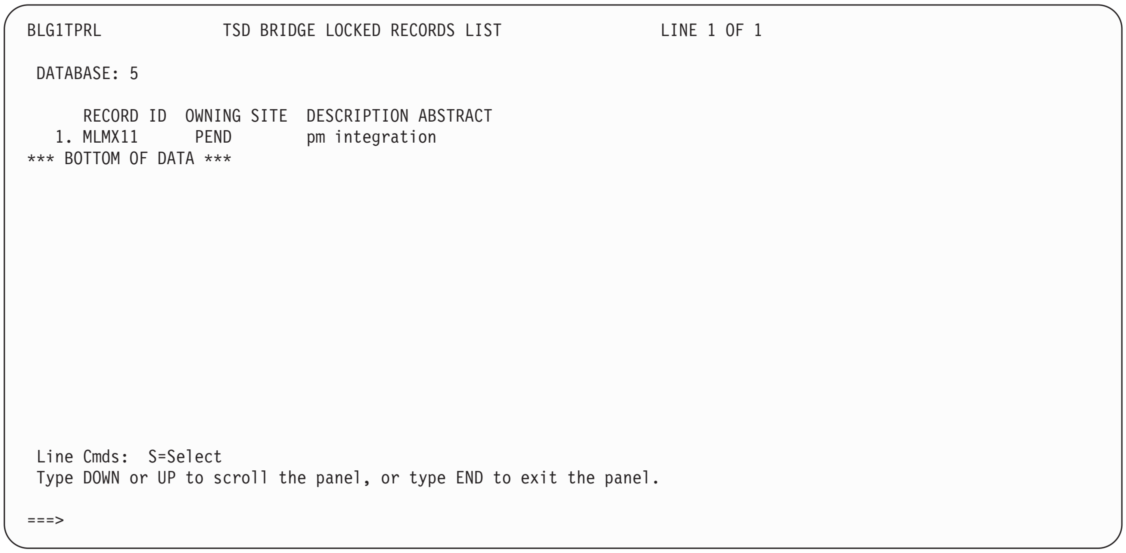
```
     + BLG0Y100 --- DATABASE ADMINISTRATION SELECTION -----------------+
     |                                                                  |
     |   IDENTIFY THE DATABASE ADMINISTRATION FUNCTION TO BE PERFORMED  |
     |                                                                  |
     |                                                                  |
     | OPTIONS:                                                         |
     |                                                                  |
     |   1. DBCLEANUP.........Clean up "BUSY" records in the database.  |
     |   2. CHECK IN..........Check records into the database.          |
     |   3. TSD Cleanup.......Unlock records locked by the TSD Bridge   |
     |                                                                  |
     |                                                                  |
     |                                                                  |
     +------------------------- SELECT ITEM ----------------------------+






  ===> 3
```

On BLG0Y300, the TSD Bridge Cleanup panel, you can specify a specific Record ID, or else you can enter an owning site in order to narrow the search for records that are pending completion of Tivoli Service Desk transactions to those that also contain that owning site. If you specify neither a TSD owning site ID or a Record ID, you will search for all records that are pending Tivoli Service Desk transaction completion.



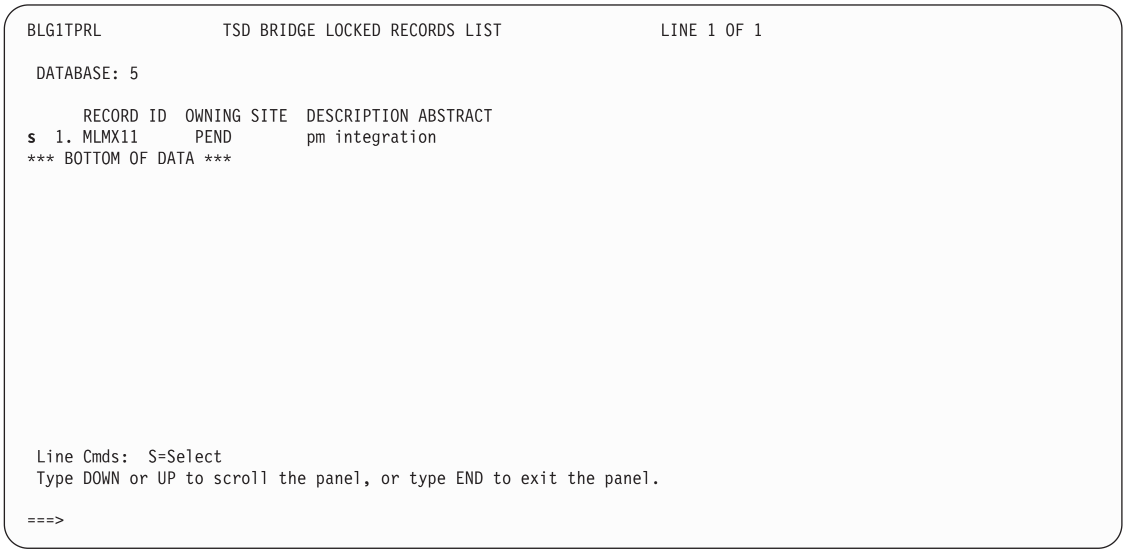
```
BLG0Y300                    TSD BRIDGE CLEANUP

Enter TSD BRIDGE data; cursor placement or input line entry allowed.



                   1. TSD owning site ID....... ______
                   2. Record ID................ ________




                    To begin the function, press Enter.










===>
```

A search results list of records that are pending completion of a Tivoli Service Desk transaction is presented.



```
BLG1TPRL             TSD BRIDGE LOCKED RECORDS LIST                 LINE 1 OF 1

 DATABASE: 5

      RECORD ID  OWNING SITE  DESCRIPTION ABSTRACT
   1. MLMX11      PEND        pm integration
*** BOTTOM OF DATA ***











 Line Cmds:  S=Select
 Type DOWN or UP to scroll the panel, or type END to exit the panel.

===>
```

Use the **S** line command to select a record and press **Enter**. The pending Tivoli Service Desk transaction will be canceled.



```
BLG1TPRL             TSD BRIDGE LOCKED RECORDS LIST                 LINE 1 OF 1

 DATABASE: 5

      RECORD ID  OWNING SITE  DESCRIPTION ABSTRACT
s  1. MLMX11      PEND        pm integration
*** BOTTOM OF DATA ***











 Line Cmds:  S=Select
 Type DOWN or UP to scroll the panel, or type END to exit the panel.

===>
```

13

Propagating Tivoli Information Management for z/OS Data

Tivoli Information Management for z/OS provides the Automatic Log Save Facility, the DB2 Extract Facility, and the LOGSAVE record so that you can propagate your Tivoli Information Management for z/OS production data to other Tivoli Information Management for z/OS or DB2 databases, or to both Tivoli Information Management for z/OS and DB2 databases.

This chapter provides an introduction to data propagation in the Tivoli Information Management for z/OS environment and gives a high-level overview of the Automatic Log Save Facility and the DB2 Extract Facility. It includes a detailed description of the LOGSAVE record and a discussion of synchronization.

Understanding Data Propagation

Data propagation involves applying the changes made to one set of data to the copy of that data in another database system or to another database in the same database system. Thus, multiple sets of data can be kept consistent.

In the Tivoli Information Management for z/OS environment, you can propagate data:

- From one Tivoli Information Management for z/OS database to other Tivoli Information Management for z/OS databases. Use the Automatic Log Save Facility.
- From Tivoli Information Management for z/OS to DB2 databases. Use the DB2 Extract Facility.
- From Tivoli Information Management for z/OS to other Tivoli Information Management for z/OS databases and to DB2 databases. Use the Automatic Log Save Facility and the DB2 Extract Facility.

The Automatic Log Save Facility provides a means to propagate changes in a production Tivoli Information Management for z/OS database to other Tivoli Information Management for z/OS databases. The DB2 Extract Facility provides a means to propagate changes in a production Tivoli Information Management for z/OS database to DB2 databases. In the Tivoli Information Management for z/OS environment, the production database is referred to as the *send database*. The send database must be running with a structured description log data set (SDLDS), which captures a copy of all changes made in the database. Changes in the send database result from successfully creating, updating, and deleting records.

A Tivoli Information Management for z/OS database that receives propagated data from a send database is referred to as a *receive database*. A receive database cannot be running with an SDLDS. A DB2 database that receives propagated data from a send database is referred to as the DB2 database. The process of propagation is transparent to the end user working on any of the databases: send, receive, or DB2.

Data propagation by either the Automatic Log Save Facility or the DB2 Extract Facility relies on a special type of Tivoli Information Management for z/OS record, the LOGSAVE record. The LOGSAVE record provides a place for you to store your processing options, such as how often to propagate changes and whether to synchronize send and receive processing. See “Understanding the LOGSAVE Record” on page 244 for details on the LOGSAVE record.

The DB2 Extract Facility also uses relational data mapping tables (RDMTs) to determine how to map your Tivoli Information Management for z/OS records to the DB2 database. You create RDMTs with BLGUT8 and store them in a data set that is in your BLGRFT concatenation.

Both data propagation facilities use batch facilities and network job entry (NJE). To initiate data propagation, you submit a batch job. The batch job offloads the SDLDS to a sequential data set, processes the sequential data set, and includes the processed data as in-stream data in another batch job that is routed to the MVS image where the receive database or the DB2 database resides. If the receive batch job is successful, it submits a batch job for routing to the MVS image where the send database resides to start the next send operation. So the receive is dependent upon the success of the send, and the next send is dependent upon the success of the receive. If the send encounters an error, the receive job is not submitted; if the receive job encounters an error, the next send job is not submitted.

Figure 8 illustrates data propagation to a Tivoli Information Management for z/OS receive database and summarizes the steps involved in setting up the process. Figure 9 on page 233 illustrates data propagation to a DB2 database and summarizes the steps involved in setting up the process. Figure 10 on page 233 illustrates data propagation to both a Tivoli Information Management for z/OS receive database and a DB2 database.

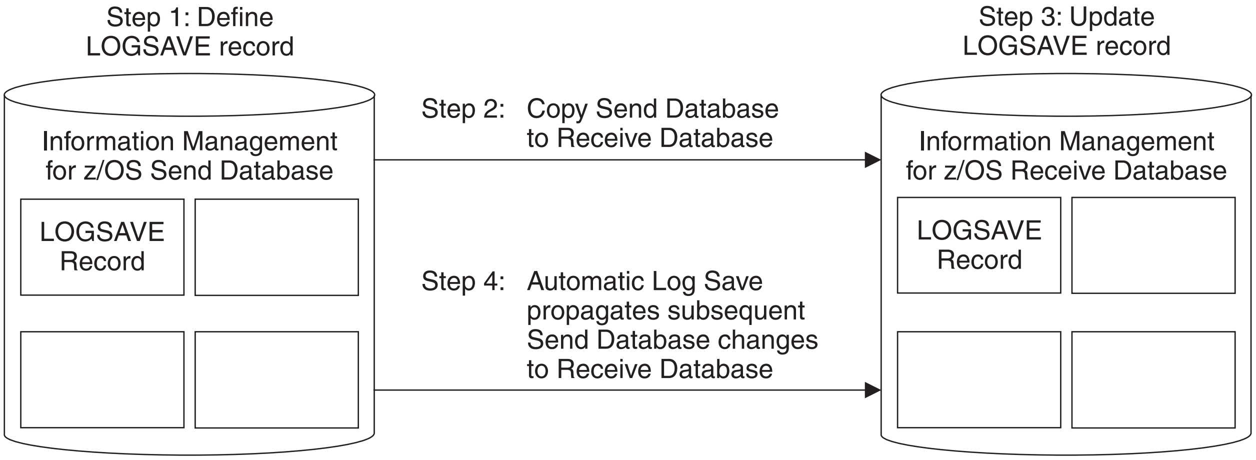


Figure 8. Data Propagation from Tivoli Information Management for z/OS to Tivoli Information Management for z/OS

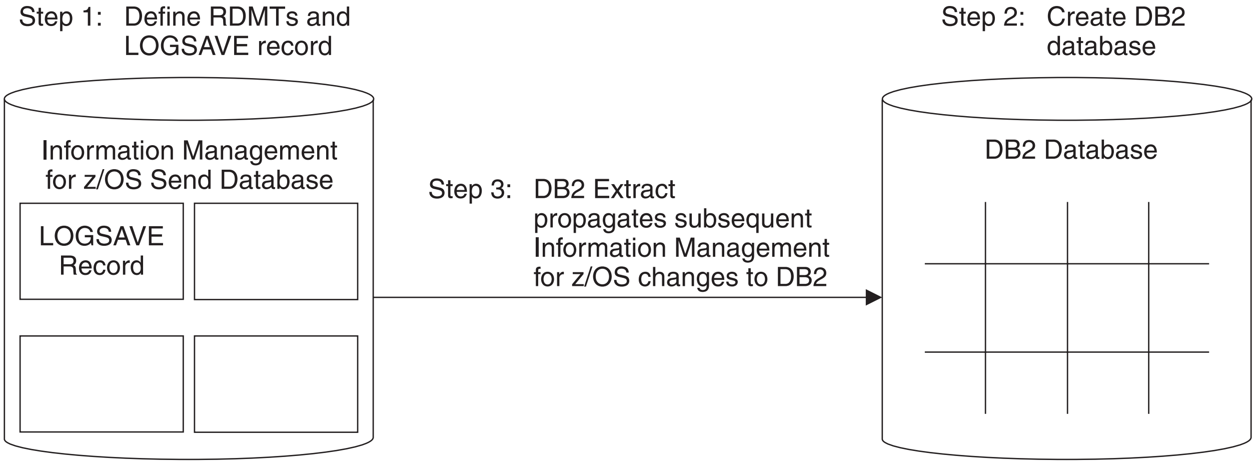

Step 1: Define RDMTs and LOGSAVE record
Step 2: Create DB2 database
Information Management for z/OS Send Database
LOGSAVE Record
Step 3: DB2 Extract propagates subsequent Information Management for z/OS changes to DB2
DB2 Database

Figure 9. Data Propagation from Tivoli Information Management for z/OS to DB2

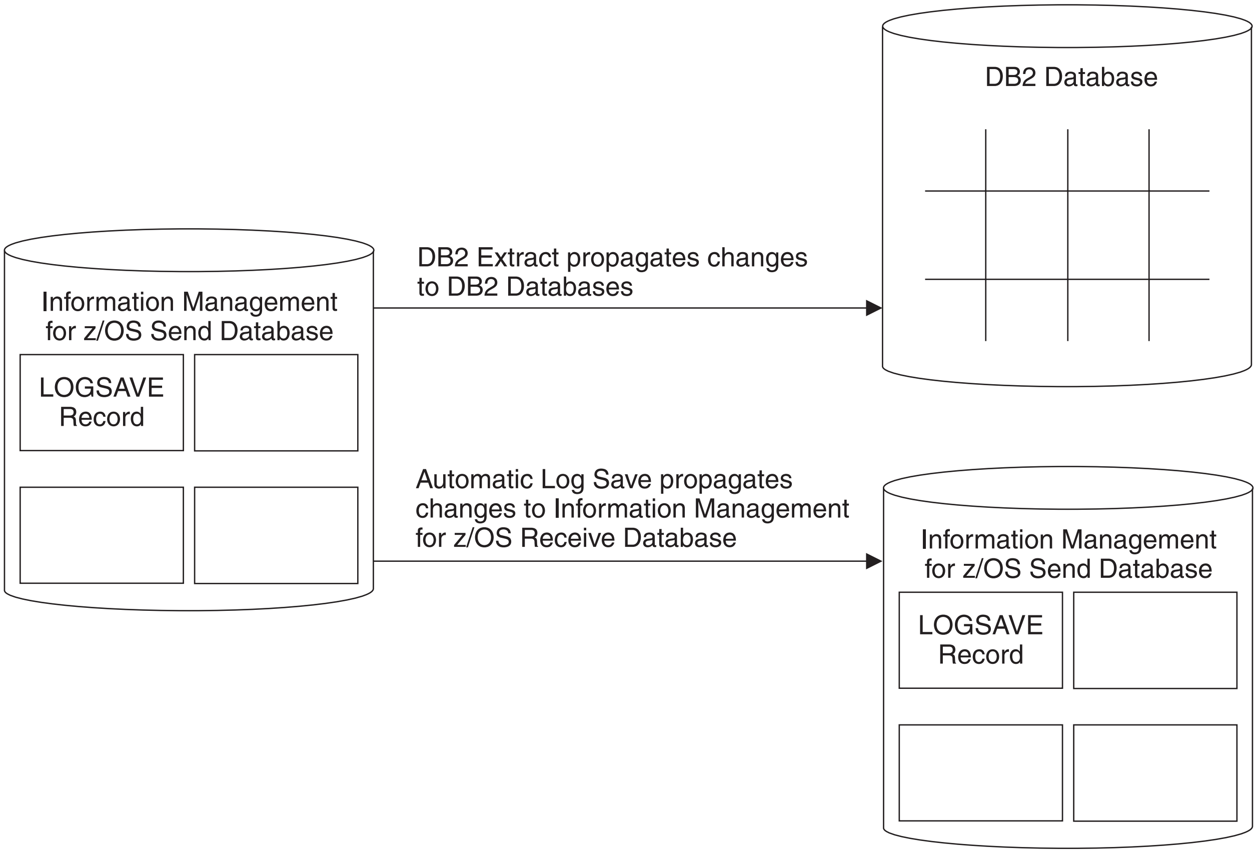

DB2 Database
Information Management for z/OS Send Database
LOGSAVE Record
DB2 Extract propagates changes to DB2 Databases
Automatic Log Save propagates changes to Information Management for z/OS Receive Database
Information Management for z/OS Send Database
LOGSAVE Record

Figure 10. Data Propagation from Tivoli Information Management for z/OS to both Tivoli Information Management for z/OS and DB2. This figure does not include the steps for setting up the two data propagation processes.

Mode of Propagation

Both the Automatic Log Save Facility and the DB2 Extract Facility support asynchronous propagation. This means that updates are applied at a later time and not within the same unit-of-work as the create, update, and delete functions on the send database. The amount of time that elapses between an update on the send database and the propagation of the update is partially determined by the value you specify for the **Time interval in minutes** field in the LOGSAVE record. (Other factors that determine the time interval are discussed in "Frequency of Data Propagation".) Because a time interval elapses between an update on the send database and the update's propagation, the propagated data is suitable for decision support applications, such as queries and reports.

You can propagate Tivoli Information Management for z/OS data to DB2 running on the same MVS image as the send database or on a different image. You can also propagate Tivoli Information Management for z/OS data to Tivoli Information Management for z/OS running on the same MVS image as the send database or on a different image.

At this time, there is no utility for propagating existing Tivoli Information Management for z/OS records to DB2. Because DB2 Extract uses the SDLDS to capture changes on the send database, only those Tivoli Information Management for z/OS records that are created or modified after the DB2 Extract Facility is started are propagated.

Choosing the Data to Be Propagated

For the Automatic Log Save Facility, you can choose to propagate either all of the records from the SDLDS or only those records that contain a particular prefix and data combination. Propagating only those records that contain a particular prefix and data combination is referred to as *filtering*. Filtering is done by the Automatic Log Save Receive function. You specify the filter prefix and filter data on the LOGSAVE record for the receive database.

For the DB2 Extract Facility, you choose the records and fields to propagate when you create the RDMTs that map your Tivoli Information Management for z/OS records to DB2 tables. You use BLGUT8 to build an RDMT for each type of Tivoli Information Management for z/OS record you want to propagate to DB2. In the input statements to BLGUT8, you map the fields in the record to DB2 tables and columns. You store the RDMTs in a data set that is in your BLGRFT concatenation and list their names in the LOGSAVE record.

Ensuring the Integrity of the Propagated Data

Propagated data is meant to be a copy of production data. Therefore, you want to ensure that all changes made to the production data are propagated to the receiving system in the same order in which they were made on the production system. Synchronization enables you to do this. See "Maintaining Synchronization" on page 249 for a discussion of synchronization and how it is maintained.

After the production data is propagated to the receiving database, use the propagated data as read-only to preserve its integrity.

Frequency of Data Propagation

Several factors determine the frequency of data propagation. You specify one of the factors in the **Time interval in minutes** field of the LOGSAVE record. Other factors that affect the frequency are how long it takes to:

- Offload the SDLDS
- Build the receive job
- Route and process the receive job
- Route the job to start the next send

Consider the following factors as well when deciding on a value for the time interval:

- Your network configuration and the amount of traffic it carries

- Whether you are synchronizing the Automatic Log Save Send and Receive functions or the DB2 Extract Send and Update Utilities
- The information about Tivoli Information Management for z/OS enqueues in "Performance and Environmental Considerations" on page 242

Using Propagated Data

Once propagation is established, you can develop programs that use the propagated Tivoli Information Management for z/OS or DB2 data for reports and decision support purposes. Figure 11 and Figure 12 on page 236 show these programs as read-only to preserve the integrity of the Tivoli Information Management for z/OS receive and DB2 databases. Any updates made to these databases are not propagated back to the send database.

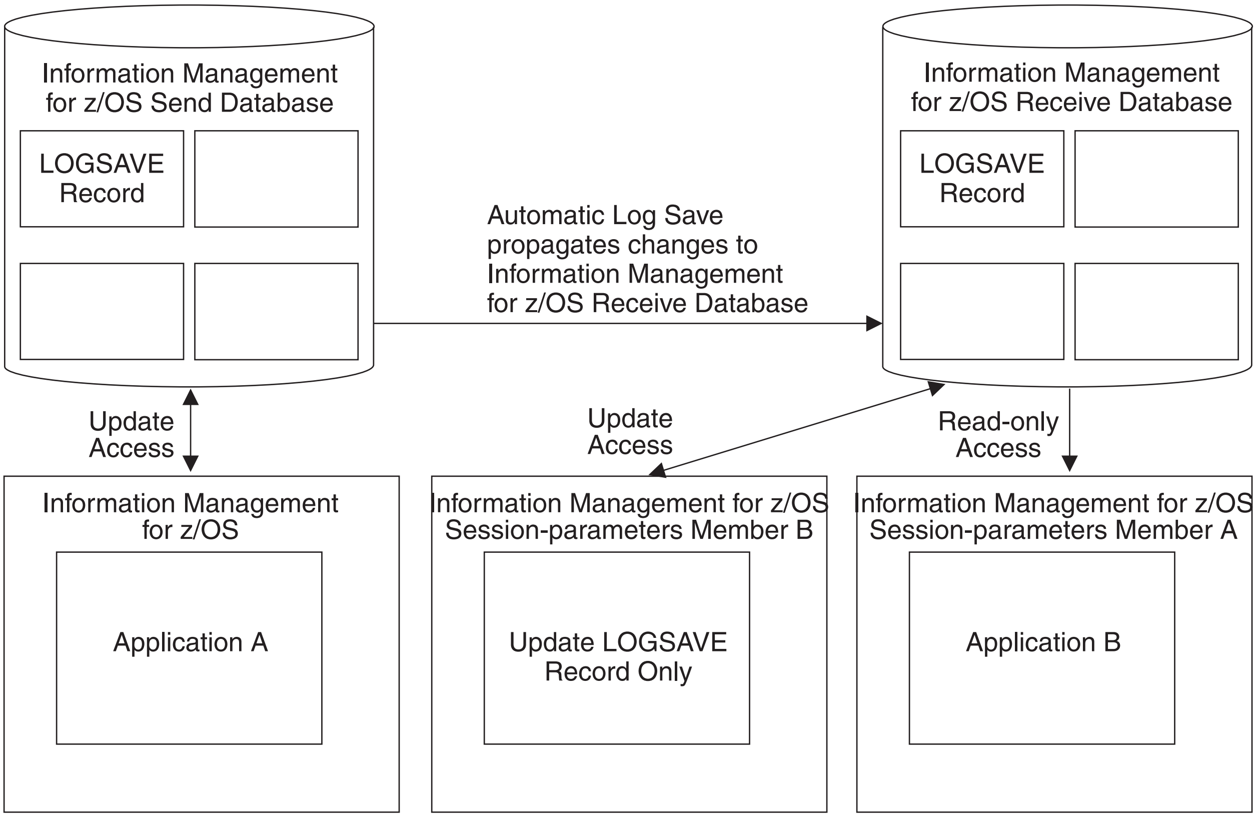

Information Management for z/OS Send Database
LOGSAVE Record
Information Management for z/OS Receive Database
LOGSAVE Record
Automatic Log Save propagates changes to Information Management for z/OS Receive Database
Update Access
Update Access
Read-only Access
Information Management for z/OS
Application A
Information Management for z/OS Session-parameters Member B
Update LOGSAVE Record Only
Information Management for z/OS Session-parameters Member A
Application B

Figure 11. Access to Data Propagated to Tivoli Information Management for z/OS. Application A could be, for example, an interactive user; Application B could be an RFT.

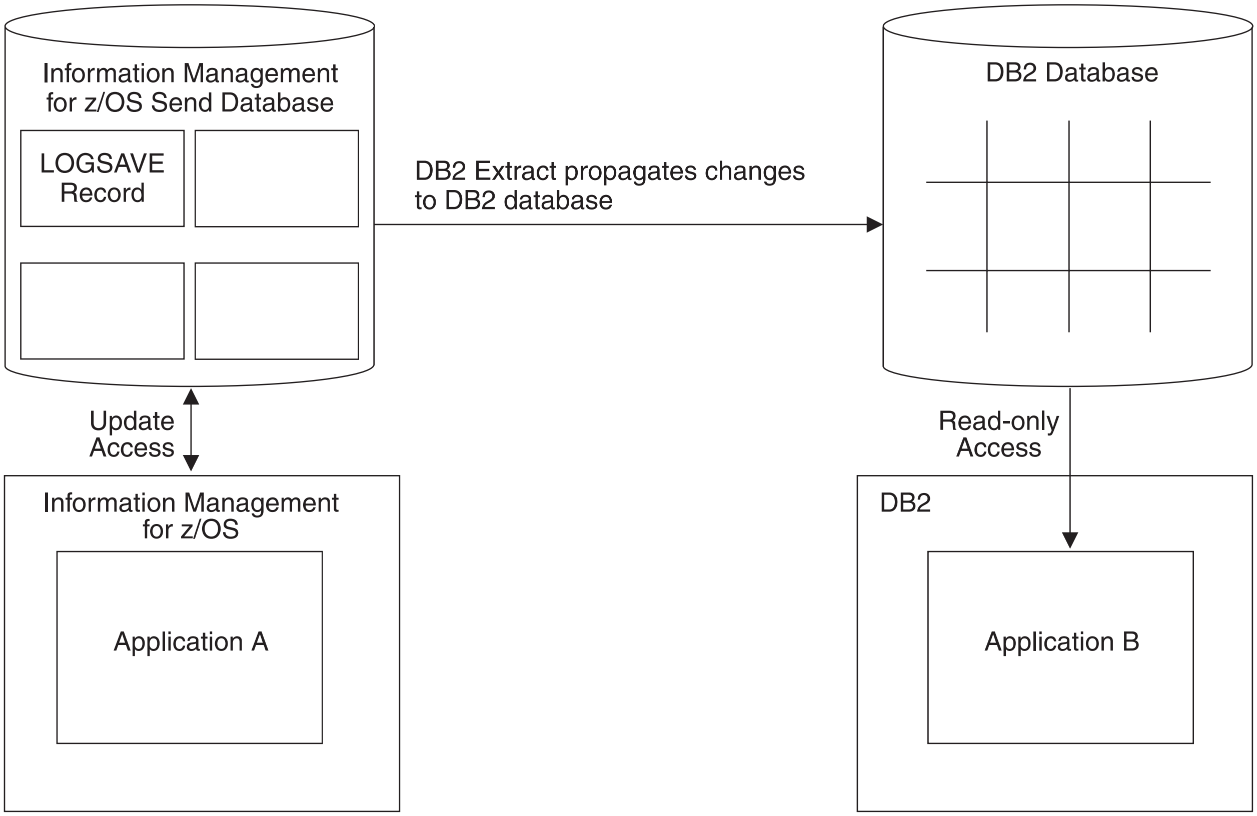

Information Management for z/OS Send Database
LOGSAVE Record
DB2 Extract propagates changes to DB2 database
DB2 Database
Update Access
Read-only Access
Information Management for z/OS
Application A
DB2
Application B

Figure 12. Access to Data Propagated to DB2. Application A could be, for example, an application program; Application B could be a QMF™ user.

Automatic Log Save Facility

The Automatic Log Save Facility propagates Tivoli Information Management for z/OS data to other Tivoli Information Management for z/OS databases. It consists of the Automatic Log Save Send Function and the Automatic Log Save Receive Function. This section gives a brief overview of the two functions. For information about using the Automatic Log Save Facility, see Propagating to a Tivoli Information Management for z/OS Database. Figure 13 on page 238 illustrates the major components and data flows of the Automatic Log Save Facility.

Automatic Log Save Send Function

The Automatic Log Save Send Function offloads the SDLDS to a sequential data set, referred to as the *send data set*, and includes the data from the send data set in-stream as part of a job to perform the Automatic Log Save Receive Function. After successful processing, the SDLDS is empty.

The Automatic Log Save Send Function reads the LOGSAVE record from the send database to determine the following:

- Send data set characteristics
- Whether to synchronize the send and receive functions
- How long to wait before offloading the SDLDS

You start the Automatic Log Save Send Function by submitting JCL BLMSASD. This JCL starts Tivoli Information Management for z/OS in batch and starts TSP BLGDUMP1, which calls user exits BLGUT4WT and BLGUT4EX.

See “Understanding Automatic Log Save Send Processing” on page 255 for details on the Automatic Log Save Send Function.

Automatic Log Save Receive Function

The Automatic Log Save Receive Function receives the send data set from a Tivoli Information Management for z/OS user running the Automatic Log Save Send Function into the *receive data set* and adds the records contained in the receive data set to the receive database.

The Automatic Log Save Receive Function reads the LOGSAVE record to determine the following:

- Receive data set characteristics
- Whether to synchronize the send and receive functions
- How long to wait before processing the receive data set
- Which records in the receive data set to filter into the receive database.

The Automatic Log Save Receive Function is started by JCL BLMSARV, which calls user exits BLGUT3WT and BLGUT3EX. BLMSARV is built and submitted by JCL BLMSASD or BLMSASDA.

See “Understanding Automatic Log Save Receive Processing” on page 256 for details on the Automatic Log Save Receive Function.

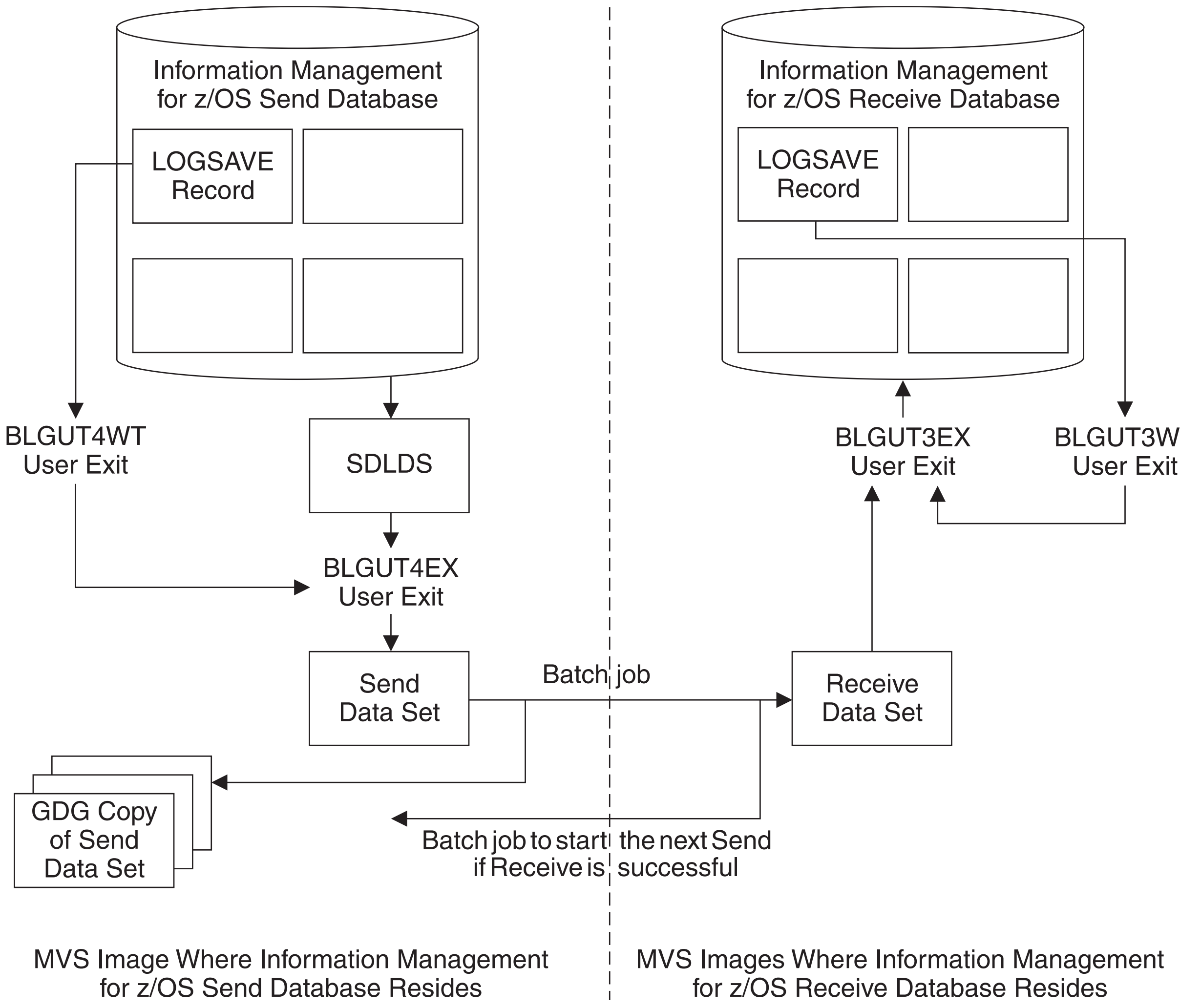


Figure 13. Automatic Log Save Facility: Major Components and Data Flows

DB2 Extract Facility

The DB2 Extract Facility propagates Tivoli Information Management for z/OS data to a DB2 database. It consists of the DB2 Extract Send Utility and the DB2 Update Utility. This section gives a brief overview of the two utilities. For information about using the DB2 Extract Facility, see Propagating to a DB2 Database. Figure 14 on page 240 illustrates the major components and data flows of the DB2 Extract Facility.

DB2 Extract Send Utility

The DB2 Extract Send Utility generates SQL statements to reflect activity in the Tivoli Information Management for z/OS send database. It uses the send data set created by user exit BLGUT4EX and RDMTs created by BLGUT8 to construct the SQL statements. It writes the SQL statements to the SQL data set.

The DB2 Extract Send Utility reads the LOGSAVE record to determine the following:

- Send data set name
- Whether to synchronize the send and update utilities
- How long to wait before offloading the SDLDS
- The authorization ID for the DB2 tables
- The names of the RDMTs to use

- SQL data set characteristics.

You start the DB2 Send Utility by submitting JCL BLMSDSD. This JCL starts Tivoli Information Management for z/OS in batch and starts TSP BLGTDBXM, which calls user exits BLGUT4WT, BLGUT4EX, and BLMSSGEN.

See “Understanding DB2 Extract Send Processing” on page 288 for details on the DB2 Extract Send Utility

DB2 Update Utility

The DB2 Update Utility starts DSNTEP2, a PL/I program supplied by DB2, to update the DB2 database with the SQL statements sent by the DB2 Send Utility.

The DB2 Update Utility is started by JCL BLMSRCV. BLMSRCV is built and submitted by BLMSSND.

See “Understanding DB2 Update Processing” on page 290 for details on the DB2 Update Utility.

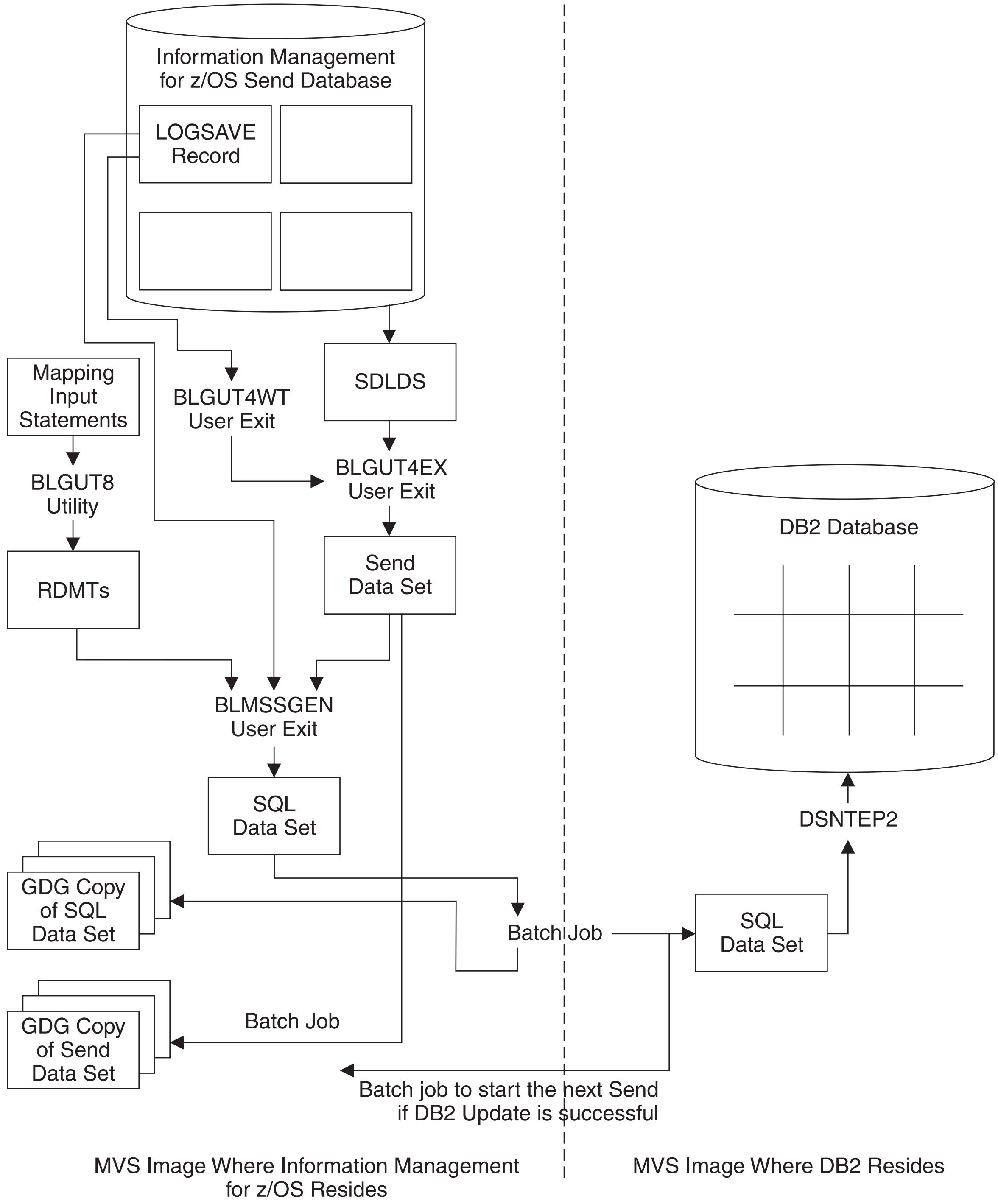


Figure 14. DB2 Extract Facility: Major Components and Data Flows

Considerations for Using Both Data Propagation Facilities

You can use both data propagation facilities, the Automatic Log Save Facility and the DB2 Extract Facility, simultaneously in one send database. You can do this because, as Figure 15 on page 242 illustrates, both facilities use the same:

- LOGSAVE record
- Send data set created by BLGUT4EX

In other words, both data propagation facilities use a piece of the Automatic Log Save Send Function: the piece that offloads the SDLDS.

A successful Automatic Log Save Receive Function starts the next Automatic Log Save Send Function. When you are using both facilities, if the Automatic Log Save Receive Function fails, both facilities stop. However, if DB2 Update Utility fails but the Automatic Log Save Receive Function succeeds, the Automatic Log Save Facility continues. Therefore, another Automatic Log Save Send Function occurs, and another DB2 Send Utility occurs as well. In this situation, the DB2 Send Utility does not stop until the Automatic Log Save Facility encounters an error or you disable the LOGSAVE record.

To ensure the integrity of your DB2 data when you are using both facilities simultaneously, develop, test, and implement procedures for:

- Monitoring the operation of both facilities
- Recovering from errors

Error recovery is discussed in Chapter 14 and Chapter 15.

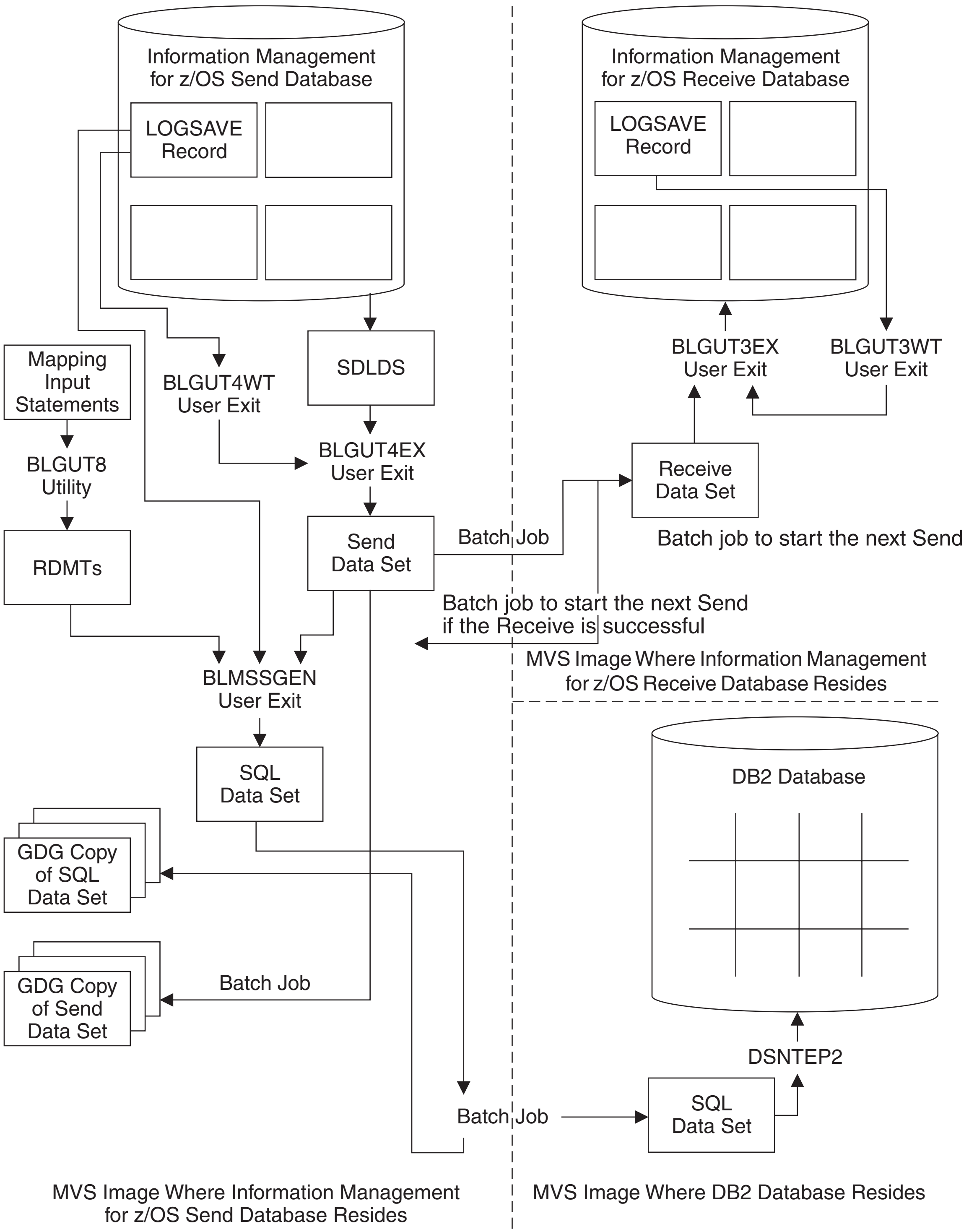

Information Management for z/OS Send Database
LOGSAVE Record
Information Management for z/OS Receive Database
LOGSAVE Record
Mapping Input Statements
BLGUT4WT User Exit
SDLDS
BLGUT3EX User Exit
BLGUT3WT User Exit
BLGUT8 Utility
BLGUT4EX User Exit
Receive Data Set
RDMTs
Send Data Set
Batch Job
Batch job to start the next Send
Batch job to start the next Send if the Receive is successful
MVS Image Where Information Management for z/OS Receive Database Resides
BLMSSGEN User Exit
SQL Data Set
DB2 Database
GDG Copy of SQL Data Set
GDG Copy of Send Data Set
Batch Job
DSNTEP2
Batch Job
SQL Data Set
MVS Image Where Information Management for z/OS Send Database Resides
MVS Image Where DB2 Database Resides

Figure 15. Data Flows When Propagating Data to Both Tivoli Information Management for z/OS and DB2

Performance and Environmental Considerations

Before you set up either the Automatic Log Save Facility or the DB2 Extract Facility, consider the following points:

- The Automatic Log Save Receive Function performs better if the receive database is not shared.

- To prevent duplicate records on the receive database, do *not* create any records on the receive database.
- BLGUT3EX obtains an exclusive enqueue on the receive database when adding records to the database. The value of the COGENQ parameter for the session-parameters member determines how long the enqueue is held. The default value for COGENQ is 10. If you increase the value for COGENQ, you increase the number of records BLGUT3EX can process each time it obtains an enqueue at the expense of other users. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for more information about session-parameters members and the COGENQ parameter.
- BLGUT4EX obtains an exclusive enqueue on the send database when it offloads the SDLDS to the send data set. Therefore, no records can be filed in the send database while the SDLDS is being offloaded. The **RECS=** and the **%FULL=** parameters on user exit BLGUT4EX enable you to control how long the exclusive enqueue is held. As shipped, TSPs BLGDUMP1 and BLGTDBXM specify a parameter of **RECS=20**. This means that BLGUT4EX obtains an exclusive enqueue on the SDLDS, processes 20 physical records, then releases the enqueue so that other users can access the SDLDS. BLGUT4EX then re-obtains the enqueue and processes 20 more physical records from the SDLDS, and so on. If you increase the number of records BLGUT4EX can process each time it obtains an exclusive enqueue, you increase the speed of BLGUT4EX at the expense of other users. If you decrease the number of records BLGUT4EX can process each time it obtains an exclusive enqueue, you decrease the impact that BLGUT4EX has on other users at the expense of slowing down BLGUT4EX. See Data Propagation User Exits for more information on the parameters.
- The DB2 Update Utility submits dynamic SQL statements with a COMMIT statement after the DELETE, INSERT, and UPDATE statements for each RNID processed. Refer to the *IBM® Database 2 Version 2 Administration Guide* for information on the size and type of locks DB2 obtains.
- Batch jobs are used to start the offloading of the SDLDS for data propagation and to transfer, receive, and load the offloaded data. A successful offload causes the receive batch job to be built, submitted, and routed to the receiving system. If the receive batch job successfully loads the data, it submits a batch job to start another offload. Batch jobs can be delayed or can fail, sometimes without notifying you. Therefore, you need to implement procedures for:
  - Checking the status of the batch jobs
  - Checking the status messages from the batch jobs
  - Checking the output from the batch jobs
  - Error recovery
  - Restarting data propagation
- If you want to use a different approach to data propagation, you can tailor the JCL that is provided in the sample library (SBLMSAMP). For example, you might want the Automatic Log Save Facility to stop after one successful receive operation. To do that, tailor the Automatic Log Save Facility as described in “Disabling Automatically Starting the Next Send” on page 277. Or you might want to use the Automatic Log Save Facility to maintain a backup copy of your SDLDS in a GDG but not to propagate your data. To do that, tailor the Automatic Log Save Facility as described in “Using the Automatic Log Save Facility for Automatic Backup” on page 276. In addition to those changes, you can, of course, make others, such as change the JCL to use a locally available method of data transfer instead of NJE.

Understanding the LOGSAVE Record

The LOGSAVE record is a Tivoli Information Management for z/OS record that contains data used by both the DB2 Extract Facility and the Automatic Log Save Facility.

To create, update, and delete a LOGSAVE record, you must have database administrator authority; to display or print the record, requires no authority.

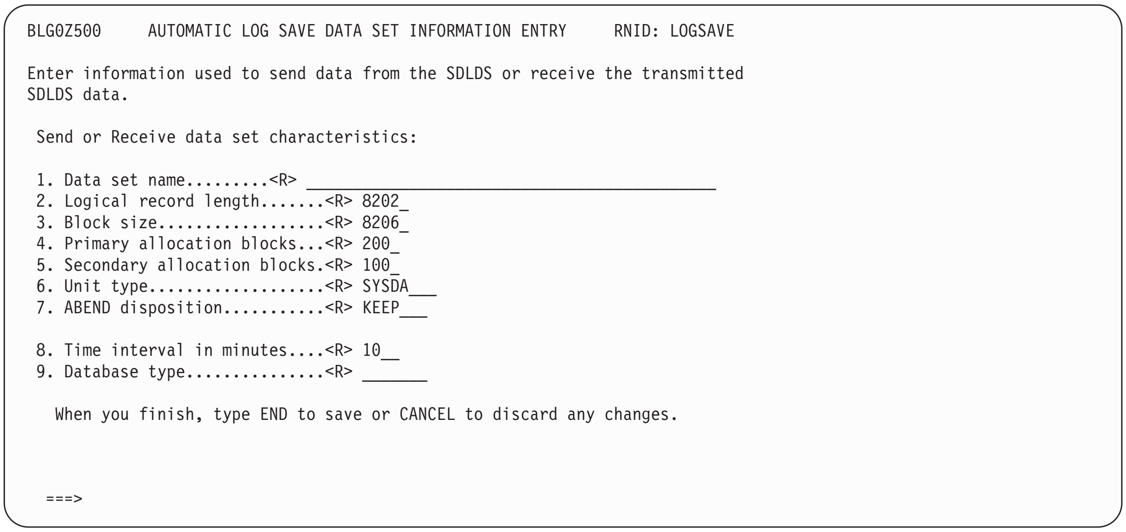
A Tivoli Information Management for z/OS database can contain only one LOGSAVE record. The Automatic Log Save Facility uses the information on panels BLG0Z500, Automatic Log Save Data Set Information Entry, and BLG0Z510, Automatic Log Save Synchronize and Filter Entry. The DB2 Extract Facility uses information on those panels plus information on panels BLG0Z530, DB2 Extract Information Entry, and BLGLSQMP, Relational Data Mapping Table Entry.

Panels BLG0Z500, BLG0Z510, BLG0ZU50, BLG0Z530, and BLGLSQMP are provided to assist you in creating a LOGSAVE record. To prevent errors, do not change the logic of these panels or any of the other panels that are involved with creating or updating the LOGSAVE record.

The following descriptions of the LOGSAVE record’s fields are organized by panel.

Fields on Panel BLG0Z500

On this panel, you define the name and DCB characteristics for the send or receive data set. If the **Database type** field contains *send*, the data set defined is the send data set. If the **Database type** field contains *receive*, the data set defined is the receive data set. Both the Automatic Log Save and the DB2 Extract Facilities use the send data set. Only the Automatic Log Save Facility uses the receive data set.



```
BLG0Z500     AUTOMATIC LOG SAVE DATA SET INFORMATION ENTRY     RNID: LOGSAVE

Enter information used to send data from the SDLDS or receive the transmitted
SDLDS data.

 Send or Receive data set characteristics:

 1. Data set name.........<R> ____________________________________________
 2. Logical record length.......<R> 8202_
 3. Block size..................<R> 8206_
 4. Primary allocation blocks...<R> 200_
 5. Secondary allocation blocks.<R> 100_
 6. Unit type...................<R> SYSDA___
 7. ABEND disposition...........<R> KEEP___

 8. Time interval in minutes....<R> 10__
 9. Database type...............<R> _______

   When you finish, type END to save or CANCEL to discard any changes.



  ===>
```

Database type

Specifies whether the record is for a send database or a receive database. You can have only one LOGSAVE record in a Tivoli Information Management for z/OS database. For the Automatic Log Save Facility, you create the LOGSAVE record on a send database and copy it to the receive database where you update it. Acceptable values are *send* and *receive*.

Data set name

Specifies the name of a sequential data set.

- For a send database, this data set is the one that the SDLDS is offloaded to. This data set is created by user exit BLGUT4EX using the DCB characteristics specified on this panel.
- For a receive database, this data set is the one that the send data set is received as by JCL BLMSARV.

Logical record length
Specifies the length of the logical records in the sequential data set. If you set this field to 0, it defaults to 8202. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for information on how to determine this value.

Block size
Specifies the block size of the data set. This value must be valid for the specified logical record length. If you set this field to 0, it defaults to 8206. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for information on how to determine this value.

Primary allocation blocks
Specifies how many blocks to allocate for the data set. This value is used only by the Automatic Log Save Send function. The number of blocks to allocate depends on the volume of records written to the SDLDS and how often you do send processing. The greater the volume of data written to the SDLDS and the longer the time interval between send processing operations, the larger the data set needs to be.

Secondary allocation blocks
Specifies how many blocks to allocate if the send data set runs out of primary allocation space. This value is used only by the Automatic Log Save Send function.

Unit type
Specifies the unit on which the send data set is to be allocated. This value is used only by the Automatic Log Save Send function.

ABEND disposition
Specifies what to do with the data set created by the send function or allocated by the receive function if either ends abnormally. Valid values are:

DELETE
Specifies that the data set is not needed, and its space on the volume is to be released if the function ends abnormally.

KEEP Specifies that the data set is to be kept on the volume if the function ends abnormally.

CATLG
Specifies that an entry pointing to the data set is to be placed in the system or user catalog when the data set is allocated, and the data set is to be kept if the function ends abnormally. Any missing index levels are created.

UNCATLG
Specifies that the entry pointing to the data set in the system or user catalog, and any unneeded indexes (with the exception of the highest level), are to be deleted.

Time interval in minutes
Specifies how many minutes to wait before a send operation or a receive operation. When the send TSP (BLGDUMP1 or BLGTDBXM) or the receive TSP

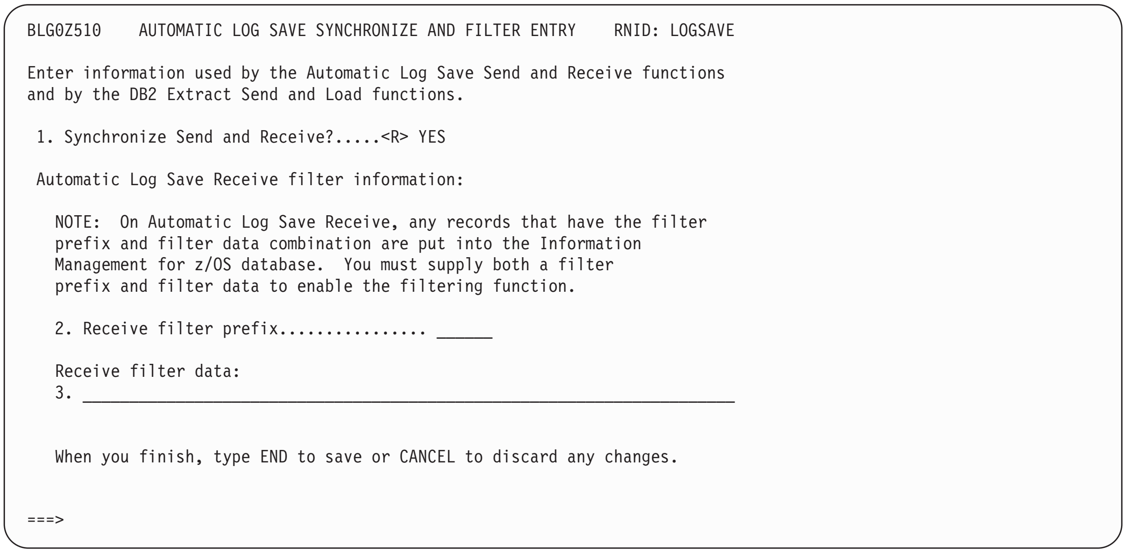
(BLGLOAD1) is started, the TSP reads the LOGSAVE record, then waits for the specified time interval (user exit BLGUT4WT for send or BLGUT3WT for receive). After the time interval elapses, control returns to the TSP, and the log or receive data set is processed. Consider your network configuration, how much traffic it carries, and the amount of time it will take to do an Automatic Log Save Receive or DB2 Update operation when you choose a value for the time interval.

If you use both the Automatic Log Save Facility and the DB2 Extract Facility, both facilities operate on the same time interval.

Fields on Panel BLG0Z510

On this panel, you specify whether to synchronize the Automatic Log Save Send and Receive functions or the DB2 Extract Send and Update utilities. If you use both facilities, the choice you make for synchronization affects both of them.

On this panel, you can also specify filter information for the Automatic Log Save Receive function. The LOGSAVE record is for the Automatic Log Save Receive function if the **Database type** field on panel BLG0Z500 contains *receive*.



```
BLG0Z510    AUTOMATIC LOG SAVE SYNCHRONIZE AND FILTER ENTRY    RNID: LOGSAVE

Enter information used by the Automatic Log Save Send and Receive functions
and by the DB2 Extract Send and Load functions.

 1. Synchronize Send and Receive?.....<R> YES

 Automatic Log Save Receive filter information:

   NOTE:  On Automatic Log Save Receive, any records that have the filter
   prefix and filter data combination are put into the Information
   Management for z/OS database.  You must supply both a filter
   prefix and filter data to enable the filtering function.

   2. Receive filter prefix................ ______

   Receive filter data:
   3. ______________________________________________________________________


   When you finish, type END to save or CANCEL to discard any changes.


===>
```

Synchronize Send and Receive?
Specifies whether to synchronize the Automatic Log Save Send and Receive functions or the DB2 Extract Send and Update Utilities.

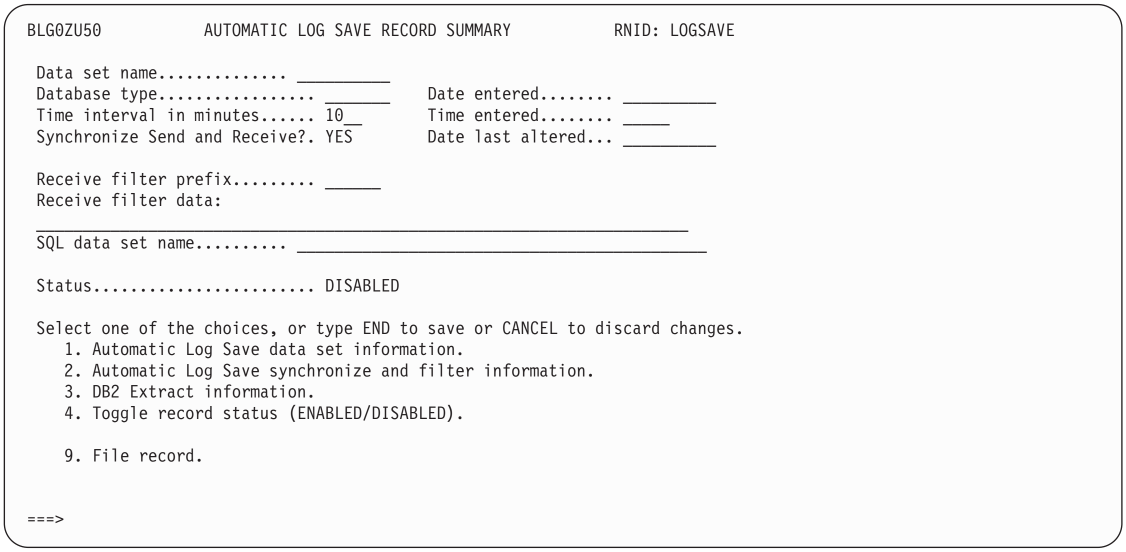
Receive filter prefix
Specifies the prefix that is in the records you want to receive. This field is used only by the Automatic Log Save Receive Function. This field and the **Receive filter data** field identify records for receive processing. This field is used only when you specify *receive* as the database type and you specify a value for the **Receive filter data** field. If you have a value in this field but no value in the **Receive filter data** field, this field is ignored.

Receive filter data
Specifies the data that is associated with the prefix in the records you want to receive. Together, this field and the **Receive filter prefix** field identify records for receive processing. This field is used only by the Automatic Log Save Receive Function. This field is used only when you specify *receive* as the database type and you specify a value for the **Receive filter prefix** field. If you have a value in this

field but no value in the **Receive filter prefix** field, this field is ignored. The data in this field is converted to uppercase before using it to perform filter processing on Log Save *receive*.

Fields on Panel BLG0ZU50

On this panel, you can review a summary of the information in the LOGSAVE record and quickly change the status of the record.

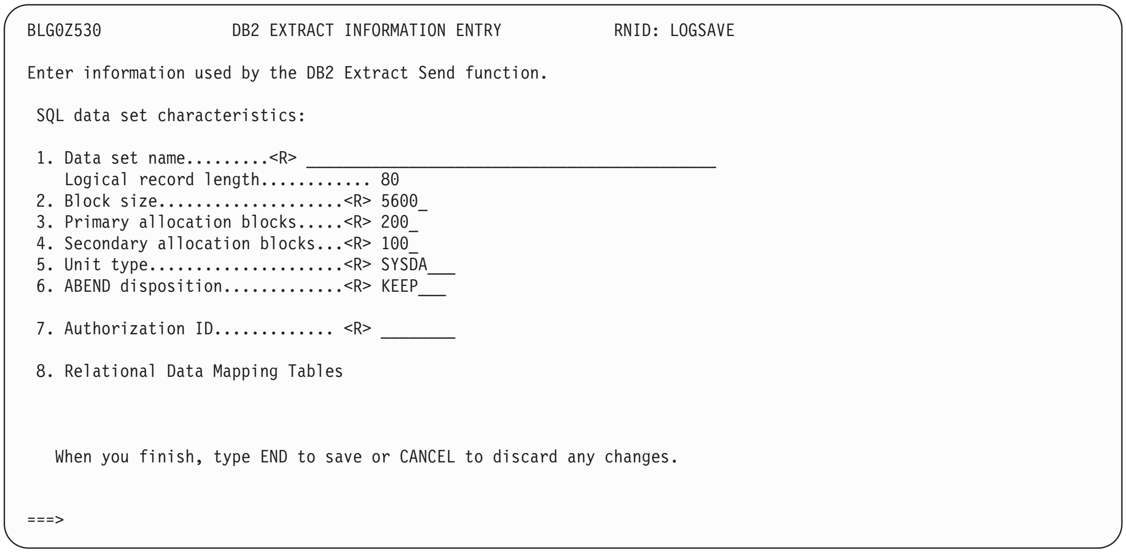


```
BLG0ZU50          AUTOMATIC LOG SAVE RECORD SUMMARY          RNID: LOGSAVE

 Data set name.............. __________
 Database type................ _______    Date entered........ __________
 Time interval in minutes...... 10__      Time entered........ _____
 Synchronize Send and Receive?. YES       Date last altered... __________

 Receive filter prefix......... ______
 Receive filter data:
 ______________________________________________________________________
 SQL data set name.......... ____________________________________________

 Status........................ DISABLED

 Select one of the choices, or type END to save or CANCEL to discard changes.
    1. Automatic Log Save data set information.
    2. Automatic Log Save synchronize and filter information.
    3. DB2 Extract information.
    4. Toggle record status (ENABLED/DISABLED).

    9. File record.


===>
```

Status Specifies whether the LOGSAVE record is *enabled* or *disabled*. The status must be enabled for Automatic Log Save processing or DB2 Extract processing, or both, to occur. Select **Toggle record status** to set the status to *enabled* or *disabled*.

Fields on Panel BLG0Z530

On this panel, you define the name and DCB characteristics for the SQL data set. The DB2 Extract Send Utility writes SQL statements to the SQL data set.



```
BLG0Z530             DB2 EXTRACT INFORMATION ENTRY           RNID: LOGSAVE

Enter information used by the DB2 Extract Send function.

 SQL data set characteristics:

 1. Data set name.........<R> ____________________________________________
    Logical record length........... 80
 2. Block size....................<R> 5600_
 3. Primary allocation blocks.....<R> 200_
 4. Secondary allocation blocks...<R> 100_
 5. Unit type.....................<R> SYSDA___
 6. ABEND disposition.............<R> KEEP___

 7. Authorization ID............ <R> ________

 8. Relational Data Mapping Tables



   When you finish, type END to save or CANCEL to discard any changes.


===>
```

Data set name
Specifies the name of the data set the DB2 Extract Send Utility is to allocate and use when generating SQL statements. This name must match the name you specify in the JCL member BLMSSND that processes the data set.

Logical record length
Specifies the length of the logical records in the SQL data set. This field is a display-only field and is set to 80.

Block size
Specifies the block size of the data set. This value must be valid for the logical record length. The SQL data set has a record format of fixed blocked.

Primary allocation blocks
Specifies how many blocks to allocate for the data set. Consider these factors to determine the size:

- The volume of data written to the SDLDS
- The time interval between send processing operations
- How many types of records you are generating SQL statements for
- How many fields you have mapped from those records.

The greater those factors are, the larger the data set needs to be. Consider, also, that two SQL statements (DELETE and INSERT) are generated for each Tivoli Information Management for z/OS record found in the SDLDS as a result of a create or update function. (For each record found in the SDLDS as a result of a delete function, only a DELETE statement is generated.) If the Tivoli Information Management for z/OS record maps to several DB2 tables, a pair of SQL statements (DELETE and INSERT) is generated for each table.

Secondary allocation blocks
Specifies how many blocks to allocate if the SQL data set runs out of primary allocation space.

Unit type
Specifies the unit on which the SQL data set is to be allocated.

ABEND disposition
Specifies what to do with the SQL data set created by the DB2 Extract Send Utility if it ends abnormally. Valid values are:

DELETE
Specifies that the data set is not needed and its space on the volume is to be released if the function ends abnormally.

KEEP Specifies that the data set is to be kept on the volume if the function ends abnormally.

CATLG
Specifies that an entry pointing to the data set is to be placed in the system or user catalog when the data set is allocated, and the data set is to be kept if the function ends abnormally. Any missing index levels are created.

UNCATLG
Specifies that the entry pointing to the data set in the system or user catalog, and any unneeded indexes (with the exception of the highest level), are to be deleted.

Authorization ID
Specifies the authorization ID that is used for all DB2 tables referenced by the generated SQL statements in the SQL data set.

Relational Data Mapping Tables
This selection takes you to panel BLGLSQMP where you specify all the RDMTs

you want to use for mapping Tivoli Information Management for z/OS management records and fields to DB2 tables and columns. The RDMTs are built by BLGUT8 and are stored in a data set in your BLGRFT concatenation. List an RDMT name for each type of Tivoli Information Management for z/OS record you want propagated to DB2.

Maintaining Synchronization

Under normal conditions, each send data set or SQL data set is received and loaded before the next send occurs. However, if you start a second send before the previous send is loaded (because of an error or data loss), a synchronization error can occur. A synchronization error indicates that the previous send data set or SQL data set was not successfully loaded.

Synchronization monitors the order in which the send data sets are loaded into the receive database and the SQL data sets update the DB2 database. A synchronization error notifies you that data is being loaded or updated out of order. Synchronization is maintained by sequence numbers stored in the following places:

Send database's SDIDS

The SDIDS record with the key of LAST_SDLDS_SEnt contains the sequence number. This sequence number is incremented by 1, then written in the send data set. The first time BLGUT4EX offloads the SDLDS, this key does not exist. BLGUT4EX initializes the sequence number to 1 and writes the record in the send data set. After BLGUT4EX successfully offloads the SDLDS, the record is stored in the SDIDS. The next time BLGUT4EX runs, the sequence number is read from the SDIDS, incremented by 1, and that value (2) is put in the send data set.

You can use the GLOSSARY command to determine the value stored in this SDIDS record. Issue the IRC **glossary,,up max, find last** from the Primary Options Menu to display the value. If your database is partitioned, you need a privilege class with master or universal access authority to see the LAST_SDLDS_SENT information in the glossary. Refer to the *Tivoli Information Management for z/OS User's Guide* for more information about the GLOSSARY command.

Receive database's SDIDS

The SDIDS record with the key of LAST_SDLDS_SENT contains the sequence number. This sequence number is 1 less than the sequence number the receive database expects to find in the next receive data set. (When the send data set is received by the Automatic Log Save Receive function, it becomes the receive data set.) The first time BLGUT3EX processes a receive data set, this key does not exist; BLGUT3EX initializes the sequence number to 1. After BLGUT3EX successfully receives and processes the receive data set, the record is stored in the SDIDS. The next time BLGUT3EX runs, the sequence number is read from the SDIDS, incremented by 1, and that value (2) is the sequence number that is expected to be in the receive data set.

You can use the GLOSSARY command to determine the value stored in this SDIDS record. Issue the IRC **glossary,,up max, find last** from the Primary Options Menu to display the value. If your database is partitioned, you need a privilege class with master or universal access authority to see the LAST_SDLDS_SENT information in the glossary. Refer to the *Tivoli Information Management for z/OS User's Guide* for more information about the GLOSSARY command.

DB2 table SYNCH_TABLE

You create a one-column table named *authorization_id*.SYNCH_TABLE in the DB2 database. The column is named SYNCH_NUMBER. Then you prime this column with the sequence number that is in the send database's SDIDS. If the SDIDS of the send database does not contain a record with a key of LAST_SDLDS_SENT, prime this column with 0. Based on the sequence number in the send data set, the DB2 Extract Send Utility generates SQL statements that check the value in this table. (This table must never contain more than one row.)

The SQL data set

The first SQL statement in this data set does the synchronization checking. If the check fails, the DB2 Update Utility stops. If the check is successful, the updating continues and the last SQL statement in the SQL data set updates the value in SYNCH_NUMBER by 1.

For example, the first time BLGUT4EX processes, the send database's SDIDS does not contain a sequence number, so the number that is written in the send data set is 1. If your value for *authorization_id* is BLM2DBX, the DB2 Extract Send Utility generates the following SQL statements, which appear as the first and last statements in the SQL data set:

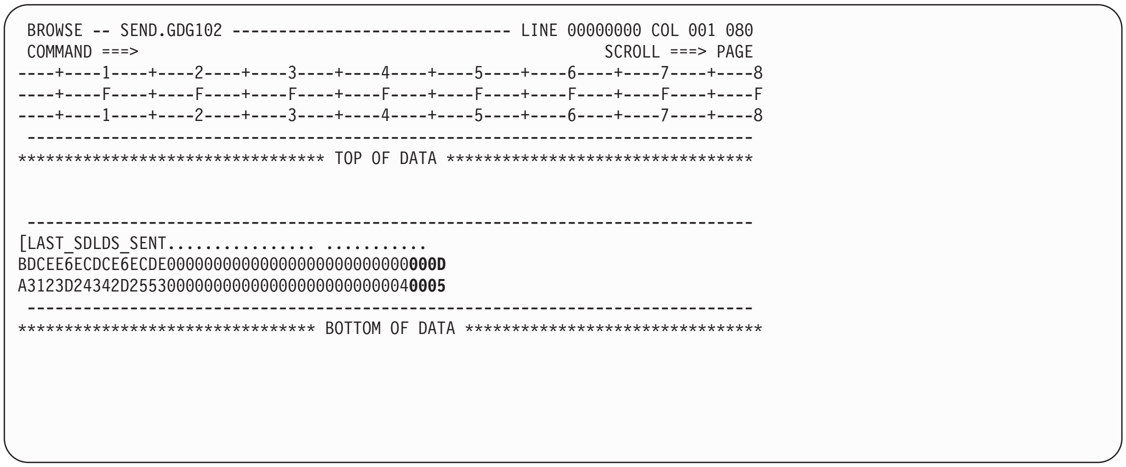
```
UPDATE BLM2DBX.SYNCH_TABLE
SET SYNCH_NUMBER = SYNCH_NUMBER/(SYNCH_NUMBER - SYNCH_NUMBER)
WHERE SYNCH_NUMBER ¬= 00000000;
⋮

UPDATE BLM2DBX.SYNCH_TABLE
SET SYNCH_NUMBER = 00000001
WHERE SYNCH_NUMBER = 00000000;
```

If SYNCH_NUMBER does not contain a value of 0, the first statement results in message DSNT408I SQLCODE = -802, a 0 divide error. The DB2 Update Utility stops with a return code of 0008 and sends a failure message.

If SYNCH_NUMBER contains a value of 0, the DB2 Update Utility sends a success message and update processing continues. If none of the subsequent SQL statements ends abnormally, the last statement updates SYNCH_NUMBER to 1.

The send and receive data set

The first record of the send and receive data sets contain the LAST_SDLDS_SENT record in hex. (When the send data set is received by the Automatic Log Save Receive function, it becomes the receive data set.) The sequence number in this record is equal to the value that was stored in the SDIDS record in the send database after the send data set was created. The sequence number in this record is also equal to the value in the SDIDS record in the receive database plus 1. For example, the first time BLGUT4EX processes, this record contains a value of 1. Depending on the length of the SDIDS key for the database, this record has a key that is either 18 or 34 bytes long, followed by 8 bytes of miscellaneous data, followed by 4 bytes that contain the sequence number. For a 34-byte SDIDS key, the sequence number is in columns 43 through 46. See Figure 16 on page 251 for an illustration. For an 18-byte key, the sequence number is in columns 27 through 30.



```
 BROWSE -- SEND.GDG102 ------------------------------ LINE 00000000 COL 001 080
 COMMAND ===>                                                  SCROLL ===> PAGE
----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
----+----F----+----F----+----F----+----F----+----F----+----F----+----F----+----F
----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
 ------------------------------------------------------------------------------
********************************* TOP OF DATA **********************************


 ------------------------------------------------------------------------------
[LAST_SDLDS_SENT................ ...........
BDCEE6ECDCE6ECDE00000000000000000000000000000D
A3123D24342D2553000000000000000000000000040005
 ------------------------------------------------------------------------------
******************************** BOTTOM OF DATA ********************************
```

Figure 16. 34-byte SDIDS Key. The sequence number in columns 43 through 46 is X'00D5', which converts to decimal 213.

If synchronization is lost while propagating data to a Tivoli Information Management for z/OS receive database, receive and send processing stop and a message is sent to the user ID specified in JCL BLMSARV. See “Restoring Synchronization” on page 280 for more information about Automatic Log Save synchronization and how to restore it.

If synchronization is lost while propagating data to a DB2 database, the DB2 Extract Facility stops unless you are running both the DB2 Extract Facility and the Automatic Log Save Facility. If you are running both, only the DB2 Update Utility stops. A message is sent to the user ID specified in JCL BLMSRCV. See “Recovering from Errors” on page 318 for more information about DB2 Extract synchronization and how to restore it.

14

Propagating to a Tivoli Information Management for z/OS Database

This information is for the program administrator responsible for implementing the Automatic Log Save Facility. You must understand Terminal Simulator Panels (TSPs), the Panel Modification Facility (PMF), and job control language (JCL) to do the tasks described in this chapter. For information about TSPs, refer to the *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference*. For information about PMF, refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide*. For information about JCL, refer to the following publications:

- *MVS/ESA JCL Reference*
- *MVS/ESA JCL User's Guide*

You must also know how to use BLGUT1, the BLX-SP operator commands, and IDCAMS. For information about BLGUT1 and the BLX-SP operator commands, refer to the *Tivoli Information Management for z/OS Operation and Maintenance Reference*. For information about IDCAMS, refer to the following publications:

- *DFSMS/MVS® Access Method Services for the Integrated Catalog Facility*
- *DFSMS/MVS Macro Instructions for Data Sets*
- *DFSMS/MVS Using Data Sets*

The Automatic Log Save Facility provides two Tivoli Information Management for z/OS functions:

Automatic Log Save Send

This function periodically offloads the Tivoli Information Management for z/OS SDLDS to a sequential data set. You can set up this function to automatically send the sequential data set to other Tivoli Information Management for z/OS databases, thereby propagating your production Tivoli Information Management for z/OS data.

Automatic Log Save Receive

This function is started automatically by a successful Automatic Log Save Send function. This function can run on the same Tivoli Information Management for z/OS system as the send function, but with a different session-parameters member, or it can run on a different Tivoli Information Management for z/OS system. You can set up the receive function to receive all the records in the offloaded SDLDS or to receive all the records except those with a specified prefix and data combination. The records in the receive databases provide copies of your production data that you can use for reporting.

A successful receive function starts the next send function.

You can also set up the send function to only offload the SDLDS to a generation data group (GDG) to automatically back up your production database and empty your SDLDS.

Before you can use the Automatic Log Save Facility, you must define a LOGSAVE record on your Tivoli Information Management for z/OS send database. To propagate data to a receive database, you must copy the record from the send database to the receive database and update it. A Tivoli Information Management for z/OS database can have only one LOGSAVE record. Therefore, a database can be either a send database or a receive database; it cannot be both a send and a receive database.

Implementing the Automatic Log Save Facility

Before you implement Automatic Log Save processing, you must understand the process and perform the following tasks:

1. Read "Propagating Tivoli Information Management for z/OS Data" on page 231 for a discussion of data propagation and a description of the LOGSAVE record.
2. Read this chapter for details of Automatic Log Save Send processing, Automatic Log Save Receive processing, and the implementation tasks.
3. Decide how your installation will use the Automatic Log Save Facility.
   - Do you want to propagate your production data to a receive database?
   - What user ID will run the send function?
   - What user ID will run the receive function?
   - Which session-parameters member will contain the send database?
   - Which session-parameters member will contain the receive database?
   - Which session-parameters member on the receive database will have read/write access for updating the LOGSAVE record?
   - How often do you want to offload the SDLDS of the send database?
   - Do you want to change how long an enqueue on the SDLDS is held for offloading?
   - Do you want to synchronize the Send and Receive functions?
   - What user ID will receive status messages?
   - What user ID will receive the job outputs?
   - Do you want to receive all the records you send?
   - Do you want to receive a specific subset?
   - How many backup levels of the GDG will you keep?
   - What procedures will you use for backup and disaster recovery?
   - How will the automatic offload of the SDLDS fit in with your current database recovery procedures?
   - Do you want to disable automatically starting the next Automatic Log Save Send operation?
   - Do you want to automatically offload the SDLDS to backup your production database without propagating the data to a receive database?
   - Do you want to propagate Tivoli Information Management for z/OS data to both DB2 and Tivoli Information Management for z/OS? If so, how will your Automatic Log Save processing fit in with your DB2 Extract processing?

- How do you want to handle error recovery?

4. Develop and test your backup and disaster recovery procedures.
5. Develop procedures for monitoring the operation of the Automatic Log Save Facility.

Understanding Automatic Log Save Send Processing

The Automatic Log Save Send function reads the LOGSAVE record to obtain the following information:

- Sequential data set characteristics, including:
  - Data set name
  - Logical record length
  - Block size
  - Primary allocation blocks
  - Secondary allocation blocks
  - Unit type
  - ABEND disposition
- Database type
- How long to wait before offloading the SDLDS
- Whether to tag the offload data set with a sequence number so that send and receive processing can be synchronized
- Whether send processing is enabled or disabled.

The Automatic Log Save Send function is started by JCL BLMSASD. This JCL does the following tasks:

1. Starts TSP BLGDUMP1, which does the following tasks. See “Data Propagation User Exits” on page 323 for more information about the user exits.
   a. Calls user exit BLGUT4WT to:
      1) Read the LOGSAVE record.
      2) Verify that the LOGSAVE record is for a send database.
      3) Verify that the status of the LOGSAVE record is enabled.
      4) Put selected information from the LOGSAVE record into the TSCA variable data area.
      5) Wait the length of time specified in the LOGSAVE record; then return to the TSP.

   b. Calls user exit BLGUT4EX to:
      1) Read information from the TSCA variable data area.
      2) Allocate the send data set.
      3) If you are synchronizing the send and receive functions:
         a) Obtain the sequence number from the SDIDS.
         b) Add 1 to the sequence number.
         c) Write the sequence number as the first record in the send data set.
      4) Move records from the SDLDS into the send data set.
      5) If you are synchronizing the send and receive functions, store the sequence number in the SDIDS.
2. Sends a status message about the success or failure of the offload to the user ID specified in the JCL.
3. Copies the send data set to the GDG.

4. Builds and routes jobstream BLMSARV by concatenating the following pieces:
   a. JCL member BLMSARV
   b. The send data set as in-stream data (uses TSO TRANSMIT and the MVS utility IEBGENER)
   c. JCL member BLMSAL1
5. Send function is complete.

If you want to propagate data to DB2 as well as to Tivoli Information Management for z/OS, see Propagating to a DB2 Database after you finish reading this one.

Understanding Automatic Log Save Receive Processing

The Automatic Log Save Receive function reads the LOGSAVE record to obtain the following information:

- Sequential data set characteristics, including:
  - Data set name
  - Logical record length
  - Block size
  - ABEND disposition
- Database type
- How long to wait before processing the receive data set
- Whether to expect the send data set to be tagged with a sequence number so that send and receive processing can be synchronized
- Whether receive processing is enabled or disabled
- What filter information, if any, to use. If no filter information is specified, all the records are received, except the LOGSAVE record.

The Automatic Log Save Receive Function is started by BLMSARV, the JCL generated and submitted by the Automatic Log Save Send Function. This JCL does the following tasks:

1. Deletes any existing receive data set.
2. Processes the in-stream send data set and uses TSO RECEIVE to receive the data into the receive data set.
3. Starts TSP BLGLOAD1, which does the following tasks. See “Data Propagation User Exits” on page 323 for more information about the user exits.

   a. Calls user exit BLGUT3WT to:
      1) Read the LOGSAVE record.
      2) Verify that the LOGSAVE record is for a receive database.
      3) Verify that the status of the LOGSAVE record is enabled.
      4) Put selected information from the LOGSAVE record into the TSCA variable data area.
      5) Wait the length of time specified in the LOGSAVE record; then return to the TSP.

   b. Calls user exit BLGUT3EX to:

      1) Read information from the TSCA variable data area.

      2) If you are synchronizing the send and receive functions:
         a) Obtain the sequence number from the SDIDS.

b) Add 1 to the sequence number.

c) Verify that this sequence number matches the sequence number in the receive data set.

3) Convert the records in the receive data set to Tivoli Information Management for z/OS records for storage in the receive Tivoli Information Management for z/OS database. The records that are converted for storage are those that contain the data specified by the **Receive filter prefix** and the **Receive filter data** fields in the LOGSAVE record. If those fields are blank, all the records in the receive data set are converted and stored, except for the LOGSAVE record.

4) If you are synchronizing the send and receive functions, store the sequence number in the SDIDS.

4. Submits JCL member BLMSASDA. It is routed to the MVS image where the send database resides to begin the next send function.
5. Sends a status message about the success or failure of the onload to the user ID specified in the JCL.
6. Routes the job output to the user ID specified on the /*ROUTE PRINT card in the JCL.

Setting Up the Automatic Log Save Send Function

The tasks that you need to do to set up the Automatic Log Save Send Function depend upon whether you plan to propagate data to Tivoli Information Management for z/OS only, DB2 only, or to both Tivoli Information Management for z/OS and DB2.

To set up the Automatic Log Save Send Function for data propagation only to Tivoli Information Management for z/OS, do the following tasks:

1. Verify that the production database is running with an SDLDS. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for more information.
2. Create a LOGSAVE record for a send database on the production database. See “Creating a LOGSAVE Record for a Send Database” on page 259 for information on this task.
3. Create a GDG for the send data set. See “Creating a GDG” on page 264 for more information on this task.
4. Prepare the JCL. See “Preparing the JCL” on page 264 for more information on this task.
5. Modify the BLGDUMP1 TSP that offloads the SDLDS if you want to change how long it can hold an enqueue on the SDLDS. (A default of RECS=20 is specified as a parameter to BLGUT4EX.) See “Modifying the BLGDUMP1 TSP” on page 268 for more information on this task.
6. Set up the Automatic Log Save Receive Function. See “Setting Up the Automatic Log Save Receive Function” on page 269 for more information on this task.

To set up the Automatic Log Save Send Function for data propagation only to DB2, do the following tasks:

1. Verify that the production database is running with an SDLDS. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for more information.

2. Create a LOGSAVE record for a send database on the production database. See "Creating a LOGSAVE Record for a Send Database" on page 259 for information on this task.
3. Do the setup for the DB2 Extract Facility in "Setting Up the DB2 Extract Facility" on page 290.

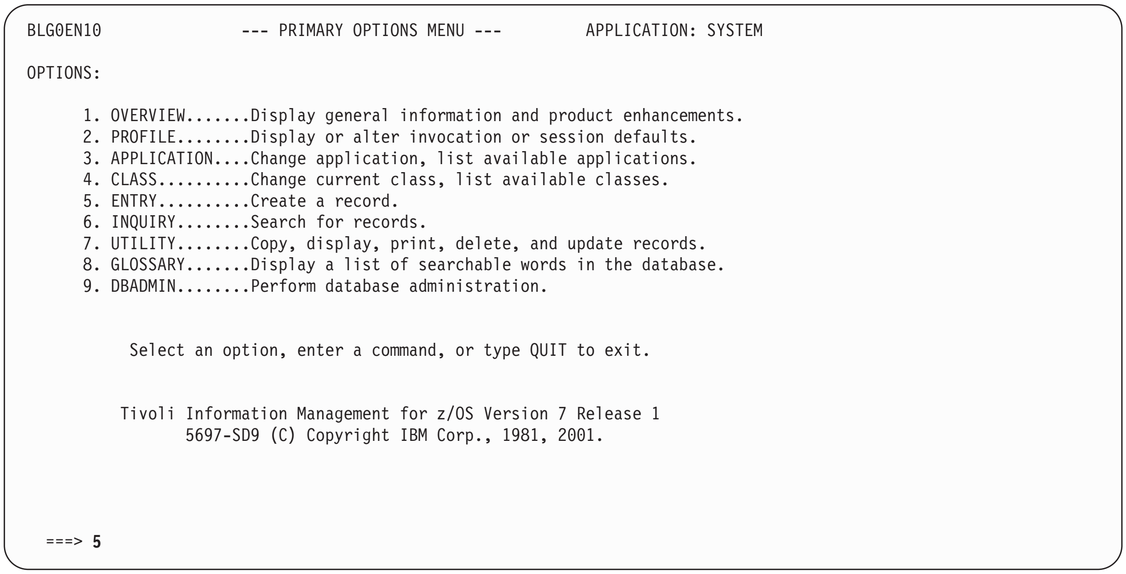
To set up the Automatic Log Save Send Function for data propagation to both Tivoli Information Management for z/OS and DB2, do the following tasks:

1. Verify that the production database is running with an SDLDS. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for more information.
2. Create a LOGSAVE record for a send database on the production database. See "Creating a LOGSAVE Record for a Send Database" on page 259 for information on this task.
3. Prepare the JCL as described in "Preparing the JCL" on page 264. Make the additional changes described in "Propagating Data to Both Tivoli Information Management for z/OS and DB2" on page 316.
4. Set up the Tivoli Information Management for z/OS Receive database. See "Setting Up the Automatic Log Save Receive Function" on page 269 for more information on this task.
5. Do the setup for the DB2 Extract Facility in "Setting Up the DB2 Extract Facility" on page 290.

Creating a LOGSAVE Record for a Send Database

When you create a LOGSAVE record for a send database, you must specify information about the sequential data set, the time interval in minutes that the TSP will delay before offloading the SDLDS, and whether you want to synchronize the send and receive functions. Before you file the record, you must enable it. If you file the record with its status identified as disabled and then start the BLGDUMP1 TSP to offload the SDLDS, the TSP encounters an error.

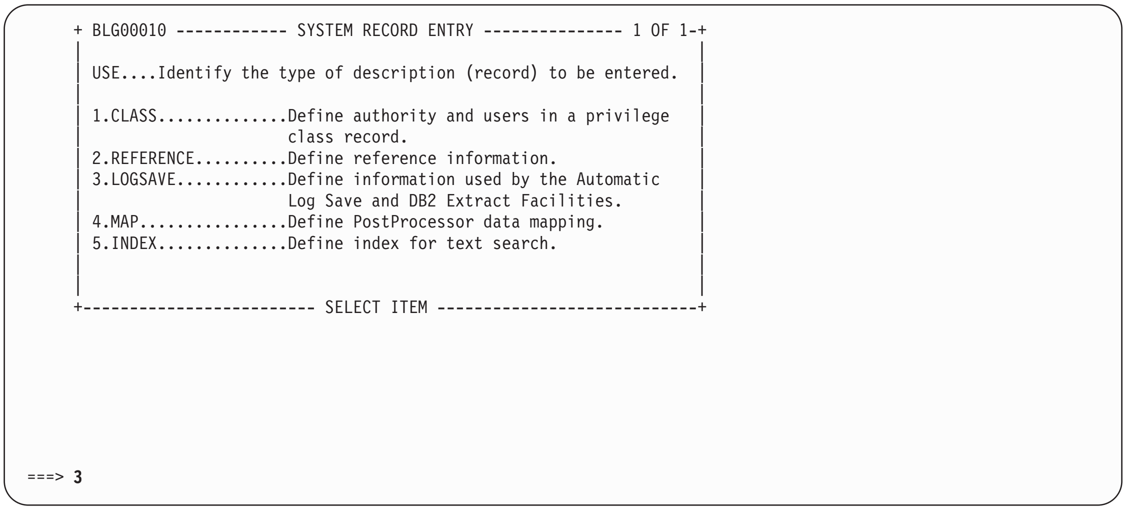
To create the LOGSAVE record, select **Entry** from the Primary Options Menu for the System application. Type **5** and press **Enter**.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.




  ===> 5
```

On BLG00010, the System Record Entry panel, select the kind of record you want to create.

For this exercise, select **Logsave**. Type **3** and press **Enter**.



```
      + BLG00010 ------------ SYSTEM RECORD ENTRY --------------- 1 OF 1-+
      |                                                                  |
      | USE....Identify the type of description (record) to be entered. |
      |                                                                  |
      | 1.CLASS.............Define authority and users in a privilege    |
      |                     class record.                                |
      | 2.REFERENCE.........Define reference information.                |
      | 3.LOGSAVE...........Define information used by the Automatic     |
      |                     Log Save and DB2 Extract Facilities.         |
      | 4.MAP...............Define PostProcessor data mapping.           |
      | 5.INDEX.............Define index for text search.                |
      |                                                                  |
      |                                                                  |
      +------------------------- SELECT ITEM ----------------------------+






===> 3
```

The next panel, the Automatic Log Save Data Set Information Entry panel, contains several required fields, some of which are primed. You can accept these values or change them. You must complete the **Data set name** field and the **Database type** field.

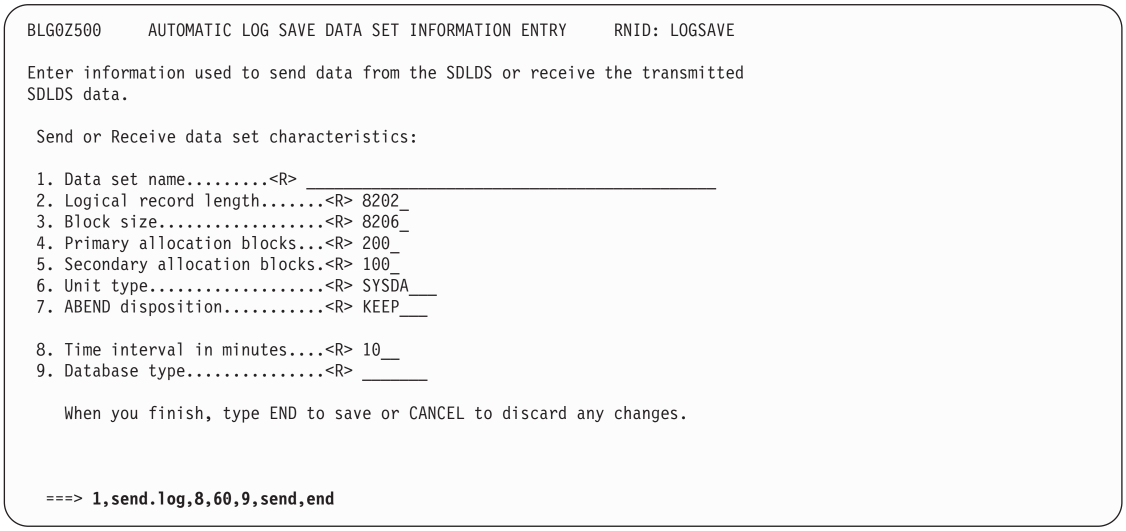
The amount of time you specify in the **Time interval in minutes** field is the amount of time the send TSP delays after being started by a successful receive operation. The actual time between offloads of the SDLDS is equal to the value in this field plus the time it takes to do each of the following tasks:

- Offload the SDLDS
- Build the receive JCL
- Route the receive JCL to the receive system
- Do the receive operation
- Route the next send JCL from the receive system to the send system

For this example, name the data set SEND.LOG; change the **Time interval in minutes** field to 60 minutes. Because you are creating the record for a send database, type this IRC:

```
1,send.log,8,60,9,send,end
```

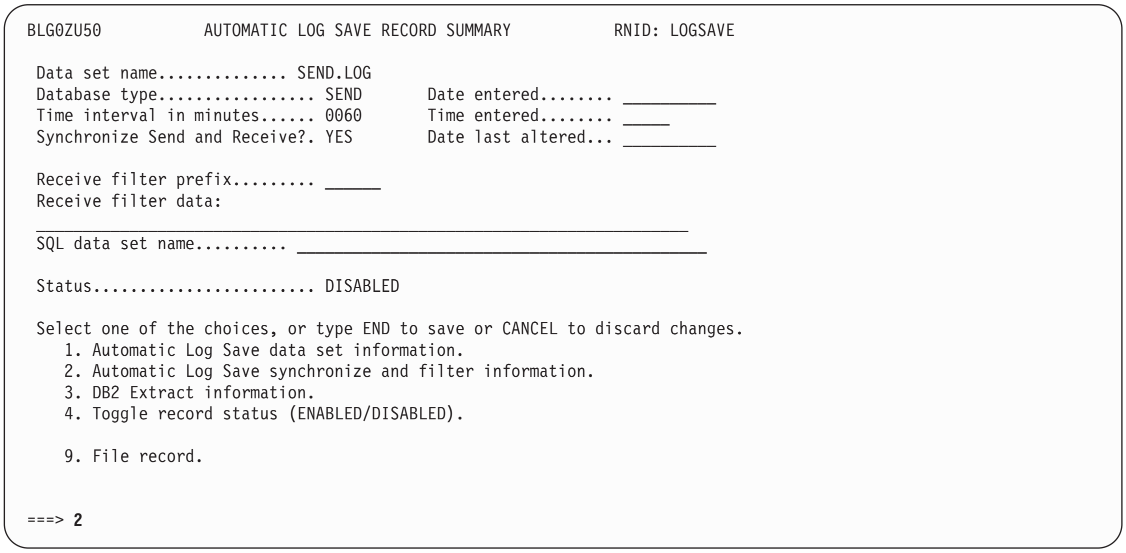
and press **Enter**.



```
BLG0Z500     AUTOMATIC LOG SAVE DATA SET INFORMATION ENTRY     RNID: LOGSAVE

Enter information used to send data from the SDLDS or receive the transmitted
SDLDS data.

 Send or Receive data set characteristics:

 1. Data set name.........<R> ____________________________________________
 2. Logical record length.......<R> 8202_
 3. Block size..................<R> 8206_
 4. Primary allocation blocks...<R> 200_
 5. Secondary allocation blocks.<R> 100_
 6. Unit type...................<R> SYSDA___
 7. ABEND disposition...........<R> KEEP___

 8. Time interval in minutes....<R> 10__
 9. Database type..............<R> _______

    When you finish, type END to save or CANCEL to discard any changes.



  ===> 1,send.log,8,60,9,send,end
```

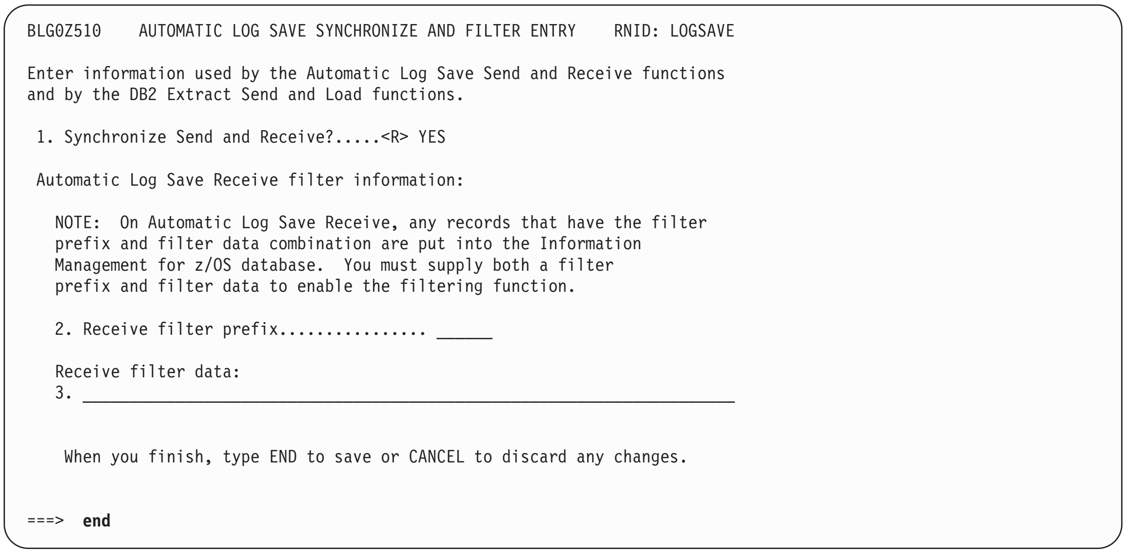
The Automatic Log Save Record Summary panel appears. The **Synchronize Send and Receive?** field defaults to YES. If you do not want to synchronize the send and receive functions, select **Automatic Log Save synchronize and filter information**. Type **2** and press **Enter**.

See "Ensuring the Integrity of the Propagated Data" on page 234 and "Maintaining Synchronization" on page 249 for information about synchronization.



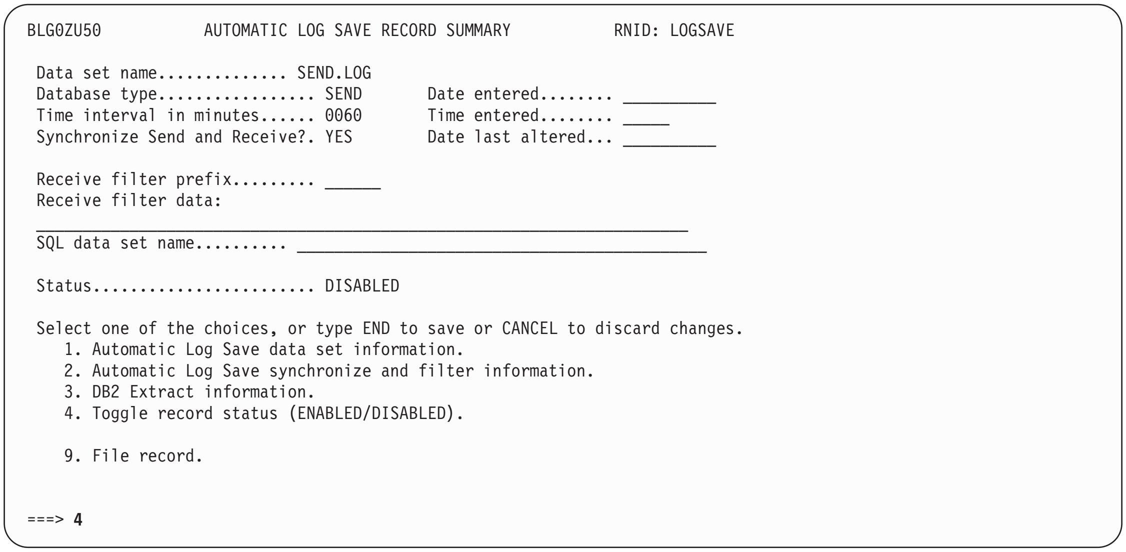
```
BLG0ZU50          AUTOMATIC LOG SAVE RECORD SUMMARY          RNID: LOGSAVE

 Data set name............. SEND.LOG
 Database type................ SEND      Date entered........ __________
 Time interval in minutes...... 0060      Time entered........ _____
 Synchronize Send and Receive?. YES       Date last altered... __________

 Receive filter prefix......... ______
 Receive filter data:
 ______________________________________________________________________
 SQL data set name.......... ____________________________________________

 Status........................ DISABLED

 Select one of the choices, or type END to save or CANCEL to discard changes.
    1. Automatic Log Save data set information.
    2. Automatic Log Save synchronize and filter information.
    3. DB2 Extract information.
    4. Toggle record status (ENABLED/DISABLED).

    9. File record.


===> 2
```

The **Synchronize Send and Receive?** field is primed with **YES**. If you do not want the functions synchronized, change this field to **NO**. Filter information does not apply to a send database, so do not complete those fields. Type **end** and press **Enter**.



```
BLG0Z510    AUTOMATIC LOG SAVE SYNCHRONIZE AND FILTER ENTRY    RNID: LOGSAVE

Enter information used by the Automatic Log Save Send and Receive functions
and by the DB2 Extract Send and Load functions.

 1. Synchronize Send and Receive?.....<R> YES

 Automatic Log Save Receive filter information:

   NOTE:  On Automatic Log Save Receive, any records that have the filter
   prefix and filter data combination are put into the Information
   Management for z/OS database.  You must supply both a filter
   prefix and filter data to enable the filtering function.

   2. Receive filter prefix................ ______

   Receive filter data:
   3. ______________________________________________________________________


    When you finish, type END to save or CANCEL to discard any changes.


===>  end
```

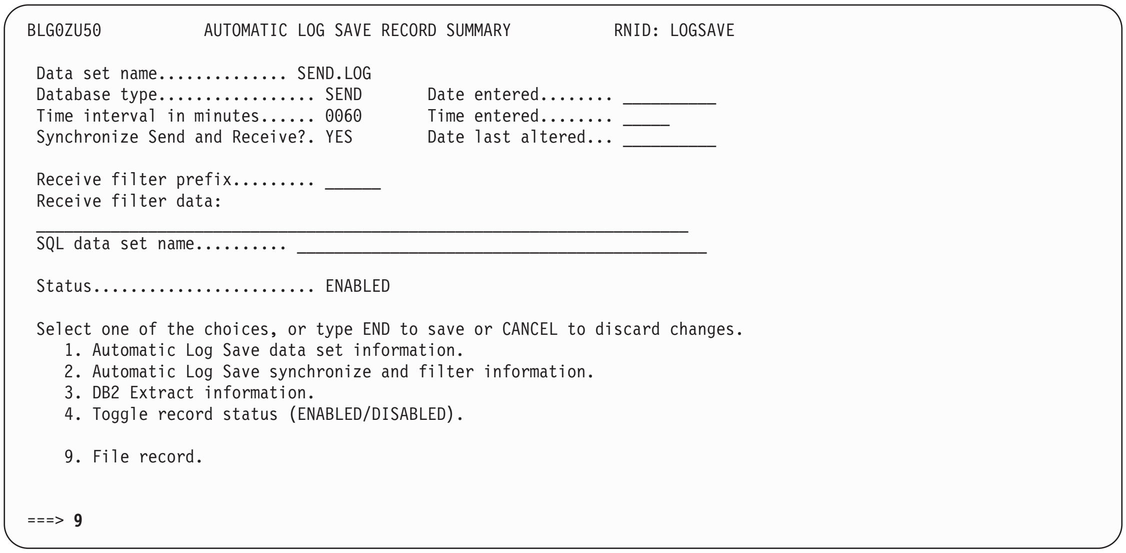
The Automatic Log Save Record Summary panel appears. Notice that the **Status** is disabled. To enable the record for send processing, type **4** and press **Enter**.



```
BLG0ZU50          AUTOMATIC LOG SAVE RECORD SUMMARY          RNID: LOGSAVE

 Data set name............. SEND.LOG
 Database type................ SEND      Date entered........ __________
 Time interval in minutes...... 0060      Time entered........ _____
 Synchronize Send and Receive?. YES       Date last altered... __________

 Receive filter prefix......... ______
 Receive filter data:
 ______________________________________________________________________
 SQL data set name.......... ____________________________________________

 Status....................... DISABLED

 Select one of the choices, or type END to save or CANCEL to discard changes.
    1. Automatic Log Save data set information.
    2. Automatic Log Save synchronize and filter information.
    3. DB2 Extract information.
    4. Toggle record status (ENABLED/DISABLED).

    9. File record.


===> 4
```

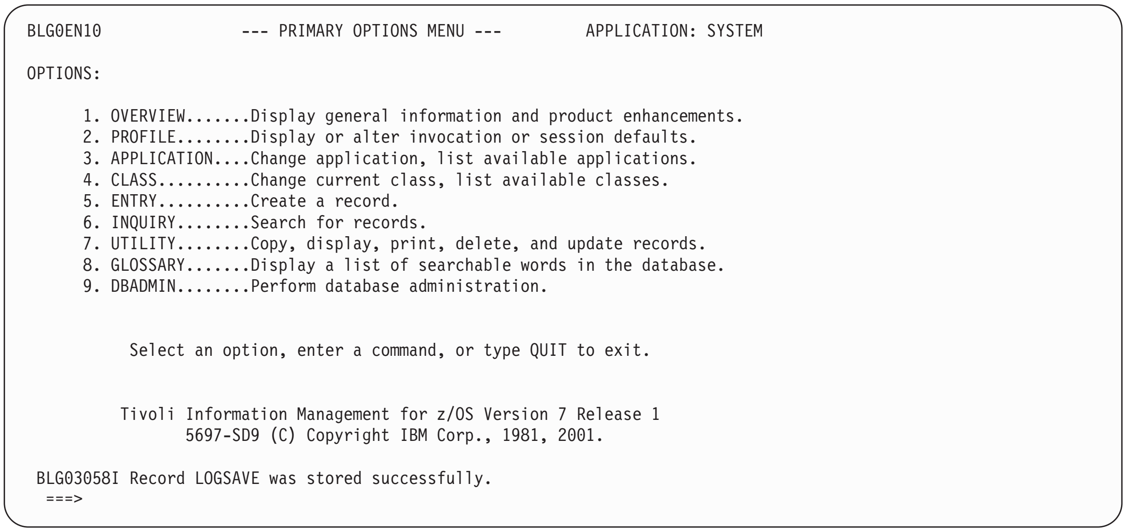
Now the **Status** is enabled. Send processing does not actually occur, however, until you start the BLGDUMP1 TSP to perform the Automatic Log Save Send function.

You have created the record for a send database. To file the record, type **9** and press **Enter**.



```
BLG0ZU50          AUTOMATIC LOG SAVE RECORD SUMMARY          RNID: LOGSAVE

 Data set name............. SEND.LOG
 Database type................ SEND      Date entered........ __________
 Time interval in minutes...... 0060      Time entered........ _____
 Synchronize Send and Receive?. YES       Date last altered... __________

 Receive filter prefix......... ______
 Receive filter data:
 ______________________________________________________________________
 SQL data set name.......... ____________________________________________

 Status........................ ENABLED

 Select one of the choices, or type END to save or CANCEL to discard changes.
    1. Automatic Log Save data set information.
    2. Automatic Log Save synchronize and filter information.
    3. DB2 Extract information.
    4. Toggle record status (ENABLED/DISABLED).

    9. File record.


===> 9
```

BLG0EN10, the Primary Options Menu, appears with a message that the LOGSAVE record was stored successfully. For future reference, you may want to print the LOGSAVE record.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.

 BLG03058I Record LOGSAVE was stored successfully.
  ===>
```

Creating a GDG

This section pertains to creating the generation data group (GDG) base for the Automatic Log Save send data set. Job BLMSASD or BLMSASDA does the actual backup to a generation of the GDG you create in this procedure.

When you create the GDG base, you define how many generations of the GDG to keep by the value you specify on the LIMIT parameter.

To create a GDG for the send data set, modify and submit JCL member BLGDG1 from the sample library, SBLMSAMP. Using uppercase text:

1. Change the job card to your local standards.
2. On the BLGDG DD card, change *userid.dsname.gdg* to the name you want to use for the generation data group for the send data set.
3. On the DEFINE statement, change *userid.dsname.gdg* to the name you want to use for the generation data group for the send data set. This name must match the name you used on the BLGDG card.
4. Change any other options in the DEFINE statement at your discretion. For example, you may want to increase or decrease the value for LIMIT.

As shipped, the DCB information in BLGDG1 matches the values primed into the LOGSAVE record on the send database when the LOGSAVE record is created.

Preparing the JCL

Copy and edit each of the following JCL members from the sample library, SBLMSAMP. Do not submit these JCL members after you update them. One of them, BLMSASD, you submit later to start the Automatic Log Save Facility. See “Directions for Updating JCL” on page 265 for general directions on updating JCL.

BLMSASD

Runs on the MVS image where the Tivoli Information Management for z/OS send database resides. You submit this job to initiate the Automatic Log Save Facility. This job does the following tasks:

- Starts TSP BLGDUMP1 using Tivoli Information Management for z/OS in batch mode.
- Sends a status message based on the success or failure of BLGDUMP1.
- Copies the Automatic Log Save send data set to a generation of a GDG.
- Prepares the send data set for inclusion in the receive job (BLMSARV).
- Builds the receive job (BLMSARV) containing the Automatic Log Save send data set as in-stream data.
- Routes job BLMSARV to the MVS image where the receive database resides for execution.

BLMSASDA

Runs on the MVS image where the Tivoli Information Management for z/OS send database resides. This job is submitted automatically by:

- BLMSARV following a successful receive operation
- BLMSASD or BLMSASDA if the send data set was empty.

Because this JCL can be submitted by BLMSARV, which runs on the MVS image where the receive database resides, you must copy this JCL to the MVS image where the receive database resides. You specify information to route this JCL to the MVS image where the send database resides.

This JCL member does the following tasks:

- Deletes the send data set created by the previous send operation.
- The tasks listed under BLMSASD.

BLMSARV

A partial job that is combined with the in-stream send data set and BLMSAL1 to create the job stream that is routed to the MVS image where the receive database resides for execution. It begins a step that copies the in-stream records from BLMSASD or BLMSASDA to a temporary data set.

BLMSAL1

A partial job that is appended to BLMSARV after the in-stream records from the Automatic Log Save send data set. It does the following tasks:

- Completes the step that copies in-stream records to a temporary data set.
- Prepares the in-stream data and performs TSO RECEIVE to receive the data into the receive data set.
- Starts TSP BLGLOAD1 using Tivoli Information Management for z/OS in batch mode.
- Sends a status message based on the success or failure of BLGLOAD1.
- If the receive is successful, submits job BLMSASDA, which is routed to the MVS image where the send database resides for execution.

BLMSASDE

Processes a previously created send data set that was not successfully received. This JCL must be manually submitted on the MVS image where the send database resides. It does the following tasks:

- Builds the job-stream (BLMSARV) containing the Automatic Log Save send data set as in-stream data.
- Routes job BLMSARV to the MVS image where the receive database resides for execution.

Directions for Updating JCL

1. Edit each member. Make your changes using uppercase text.
2. Follow the instructions in the prologue of each member and printed in this chapter, to find the specific job steps you need to update.
3. Verify the accuracy of each change you make against your LOGSAVE record, TSP, and other JCL members.
4. Be sure to change the job card to your local standards.
5. Save the updated member.

Preparing BLMSASD

1. Change the job card to your local standards.

2. Change the *node.userid* on the /*ROUTE PRINT card to the node and user ID to receive the job output.

3. In step OFFLD, change:

   a. The data set names for the STEPLIB, ISPPROF, ISPPLIB, ISPTLIB, ISPMLIB, ISPSLIB, BLGTRACE, ISPLOG, SYSPROC, and SYSTSPRT DDnames to correspond to the data set names at your installation.

   b. The value for SESS(*xx*) to the session-parameters member of your send database.

4. In step GOODOFF, change:

   a. *sample message* to your message for a successful completion (return code = 0000) of the SDLDS offload.

   b. *node.userid* to the destination for the message.

5. In step FAILOFF, change:

   a. *sample message* to your message for an unsuccessful offload of the SDLDS (return code ≠ 0000).

   b. *node.userid* to the destination for the message.

6. In step SAVESND, change:

   a. *userid.dsname1* on the SYSUT1 and SYSUT2 DDnames to the name of the send data set that is in the LOGSAVE record on the send database.

   b. *userid.gdgname1* to the name of the GDG for the send data set.

   c. The DCB attributes for the GDG generation to match those used for the send data set.

7. In step XMITSND, change *userid.dsname1* to the name of the send data set that is in the LOGSAVE record on the send database.

8. In step SUBSND, change the data set name to the name of your local data set that contains JCL BLMSASDA.

9. In step COPY1, change:

   a. The JCL data set name to the name of your local data set that contains JCL BLMSARV.

   b. The space DCB parameter on the SYSUT2 DD name to a value large enough to handle splitting each 80-byte record created from the transmit into two 80-byte records (the splitting occurs in step SPLIT).

10. In step COPY2, change the JCL data set name to the name of your local data set that contains BLMSAL1.

11. In step GOODSUB, change:

    a. *sample message* to your message for a successful submission of the receive JCL.

    b. *node.userid* to the destination for the message.

Preparing BLMSASDA

1. Change the job card to your local standards. This job is submitted on the MVS image where the send database resides. It is also submitted on the MVS image where the receive database resides and is routed to the MVS image where the send database resides for execution. You may need user and password information.

2. Change *node* on the /*ROUTE XEQ card to the node where the send database resides.

3. Change the *node.userid* on the /*ROUTE PRINT card to the node and user ID to receive the job output.
4. In step DELETE, change *userid.dsname1* to the name of the send data set that is specified in the LOGSAVE record on the send database.
5. The rest of this JCL member is the same as BLMSASD beginning with step 3 under Preparing BLMSASD. Therefore, go to “Preparing BLMSASD” on page 265 and make the changes listed there beginning at step 3.

Copy this JCL to a data set on the MVS image where the receive database resides.

Preparing BLMSARV

1. Change the job card to your local standards. This job is routed to the MVS image where the receive database resides; you may need user and password information.
2. Change *node* on the /*ROUTE XEQ card to the node where the job is to run.
3. Change *node.userid* on the /*ROUTE PRINT to the node and user ID that is to receive the job output.
4. In step DELETE, change *userid.dsname2* to the name of the receive data set that is specified in the LOGSAVE record on the receive database. This name must match the data set name in step RCVSND in JCL member BLMSAL1.

Preparing BLMSAL1

1. In DD card SYSUT2, which is part of the REBLK step started in BLMSARV, you may need to change the unit and space parameters.
2. In step RCVSND, change *userid.dsname2* to the name of the receive data set that is specified in the LOGSAVE record on the receive database. This name must match the data set name in step DELETE step in JCL member BLMSARV.
3. In step JOIN, you may need to change the unit and space parameters.
4. In step ONLOAD, change:
   a. The data set names for the STEPLIB, ISPPROF, ISPPLIB, ISPTLIB, ISPMLIB, ISPSLIB, BLGTRACE, ISPLLIB, SYSPROC, and SYSTSPRT DDnames to correspond to the data set names on the MVS image where the receive database resides.
   b. The value for SESS(*yy*) to the session-parameters member that has the receive database as database 5 with read/write access.
5. In step GOODON, change:
   a. *sample message* to your message for a successful completion (return code = 0000) of the onload.
   b. *node.userid* to the destination for the message.
6. In step FAILON, change:
   a. *sample message* to your message for an unsuccessful onload (return code ≠ 0000).
   b. *node.userid* to the destination for the message.
7. In step SUBSND, change the data set name to the name of your data set on the MVS image where the receive database resides that contains JCL BLMSASDA.
8. In step GOODSUB, change:

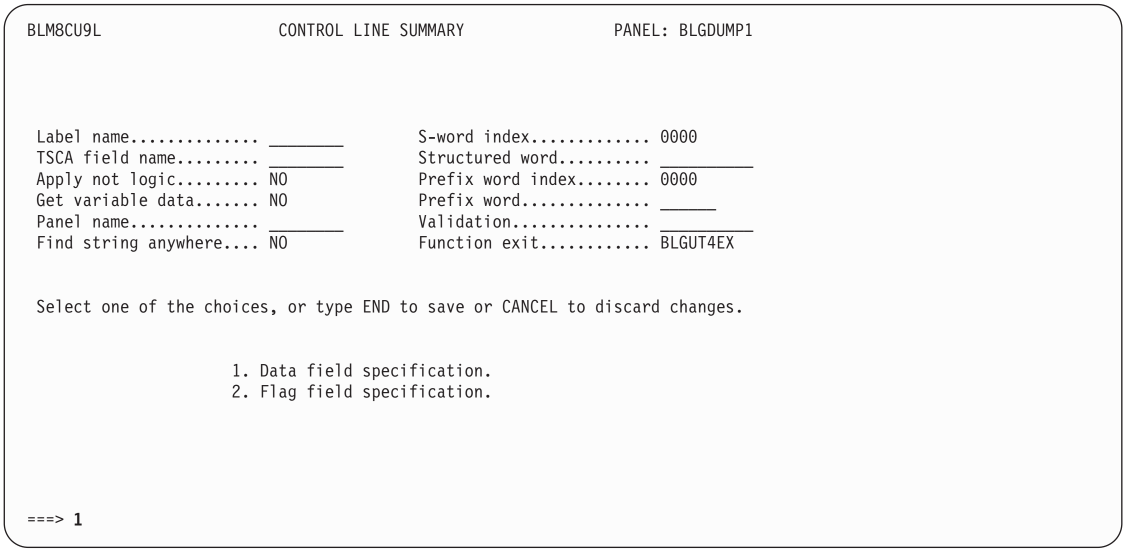
a. *sample message* to your message for a successful submission of the next send JCL.
b. *node.userid* to the destination for the message.

Modifying the BLGDUMP1 TSP

If you want to change the maximum number of records to offload each time an enqueue is obtained on the SDLDS, you can change the RECS= and %FULL= parameters in the control line that calls user exit BLGUT4EX. As shipped, the BLGDUMP1 tsp has RECS=20. See Data Propagation User Exits for more information about the parameters and the values they accept. The following example illustrates changing the RECS= parameter and adding the %FULL= parameter.

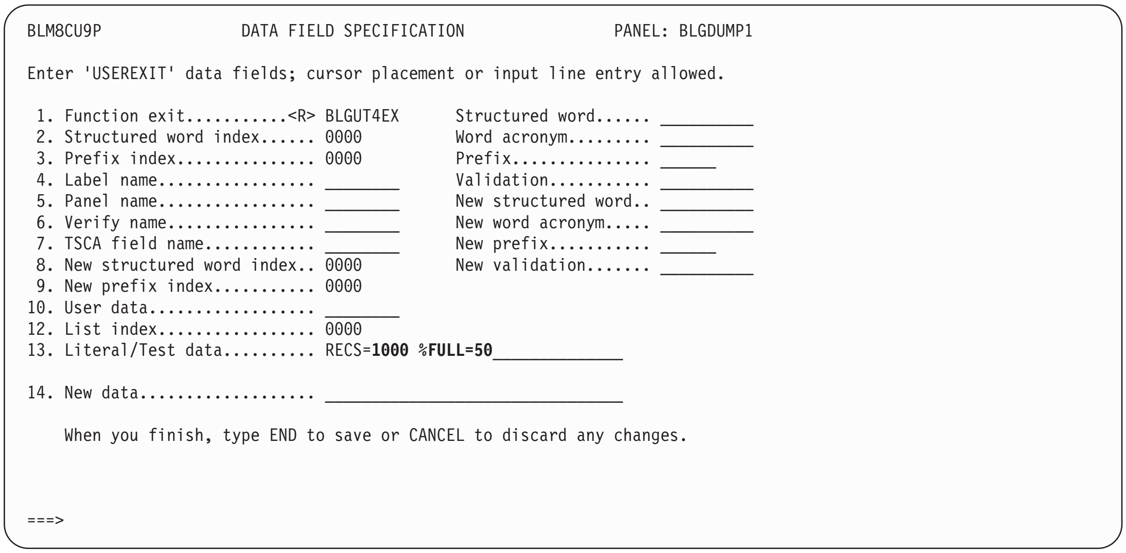
Using PMF, update the BLGDUMP1 tsp. Locate the control line for user exit BLGUT4EX. Update that line of the TSP.

On the Control Line Summary panel, select **Data field specification**. Type **1** and press **Enter**.



```
BLM8CU9L                 CONTROL LINE SUMMARY              PANEL: BLGDUMP1



 Label name.............. ________        S-word index............ 0000
 TSCA field name......... ________        Structured word.......... __________
 Apply not logic......... NO              Prefix word index........ 0000
 Get variable data....... NO              Prefix word.............. ______
 Panel name.............. ________        Validation............... __________
 Find string anywhere.... NO              Function exit............ BLGUT4EX


 Select one of the choices, or type END to save or CANCEL to discard changes.


                     1. Data field specification.
                     2. Flag field specification.




===> 1
```

You enter the parameters in the **Literal/Test data** field on the Data Field Specification panel. In this example, a value of 1000 for RECS= and a value of 50 for %FULL= have been entered.



```
BLM8CU9P              DATA FIELD SPECIFICATION               PANEL: BLGDUMP1

Enter 'USEREXIT' data fields; cursor placement or input line entry allowed.

 1. Function exit..........<R> BLGUT4EX      Structured word...... __________
 2. Structured word index...... 0000          Word acronym......... __________
 3. Prefix index.............. 0000          Prefix.............. ______
 4. Label name................ ________      Validation.......... __________
 5. Panel name................ ________      New structured word.. __________
 6. Verify name............... ________      New word acronym..... __________
 7. TSCA field name........... ________      New prefix.......... ______
 8. New structured word index.. 0000          New validation....... __________
 9. New prefix index.......... 0000
10. User data................. ________
12. List index................ 0000
13. Literal/Test data......... RECS=1000 %FULL=50______________

14. New data.................. ________________________________

    When you finish, type END to save or CANCEL to discard any changes.



===>
```

After you file the updated TSP, you must copy it from the write panel data set to the read panel data set so that it is available for use. Next, set up the receive database. After you do that, you are ready to start BLMSASD, the batch job that runs the TSP.

Setting Up the Automatic Log Save Receive Function

To set up the Automatic Log Save Receive Function, do the following tasks:

1. Define a receive database without an SDLDS in a session-parameters member. Define the receive database (SDDS and SDIDS) as a read-only database. You can do this in one of the following ways:

   - Define the SDDS and SDIDS for the receive database as database 5, and specify RDONLY=YES on the BLGCLDSN macro statements for the SDDS and SDIDS.
   - Define the SDDS and SDIDS for the receive database as database 4, 7, 8, or 9, so they cannot be written to. Specify a different SDDS and SDIDS for database 5.

   Use this session-parameters member for the applications that use the data in the receive database. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for information on defining session-parameters members.

2. Define a receive database without an SDLDS in a second session-parameters member. Define the receive database (SDDS and SDIDS) as database 5 with read/write access. This is the session-parameters member that the BLGLOAD1 tsp is run under. Therefore, do not create any records in this database. The program administrator needs read/write access to this database to update the LOGSAVE record. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for information on defining session-parameters members.

3. Copy the LOGSAVE record from the send database to the receive database. See "Copying the LOGSAVE Record into the Receive Database" on page 270 for more information on this task.

4. Optionally, copy other records from the send database. See "Copying the LOGSAVE Record into the Receive Database" for more information on this task.
5. Update the LOGSAVE record. See "Updating the LOGSAVE Record for Receive" on page 271 for more information on this task.
6. Copy JCL BLMSASDA from the MVS image where the send database resides to the MVS image where the receive database resides. See "Copying the JCL" on page 275 for more information on this task.

Copying the LOGSAVE Record into the Receive Database

The LOGSAVE record in the Tivoli Information Management for z/OS receive database must have the same VSAM key that the LOGSAVE record in the send database has. The following sections outline three methods you can use to copy the LOGSAVE record from the send database to the receive database.

Copying Entire Send Database without BLGUT1

This method uses IDCAMS REPRO to copy the entire SDDS and SDIDS from the send database to the receive database. You can use this method only if the receive database is empty.

1. On the send database, do the following tasks:
   a. Issue the BLX-SP FREE command on the send database's SDDS and SDIDS so that no records can be created on the send database.
   b. REPRO the send database's SDDS and SDIDS to a sequential data set using IDCAMS.
   c. Issue the BLX-SP REALLOC command on the send database's SDDS and SDIDS.
   d. Do not start send processing, but you may start using your send database for other functions.
2. On the receive database, REPRO the SDDS and SDIDS from the send database into the receive database using IDCAMS. You may need to issue the BLX-SP FREE command if the receive database is allocated to a BLX-SP. Then, after the REPRO, issue the BLX-SP REALLOC command.

Copying Entire Send Database with BLGUT1

This method uses IDCAMS REPRO to copy the entire SDDS from the send database to the receive database; then uses BLGUT1 to rebuild the SDIDS on the receive database. The receive database must be empty.

1. On the send database, do the following tasks:
   a. Issue the BLX-SP FREE command on the send database's SDDS and SDIDS so that no records can be created on the send database.
   b. REPRO the send database's SDDS to a sequential data set using IDCAMS.
   c. Issue the BLX-SP REALLOC command on the send database's SDDS and SDIDS.
   d. Do not start send processing, but you may start using your send database for other functions.
2. On the receive database, do the following tasks:

a. Using IDCAMS, REPRO the SDDS from the send database into the receive database. You may need to issue the BLX-SP FREE command if the receive database is allocated to a BLX-SP. Then, after the REPRO, issue the BLX-SP REALLOC command.

b. Run BLGUT1 to build the receive database's SDIDS. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for more information.

Copying Only the LOGSAVE Record from the Send Database

This method uses IDCAMS REPRO to copy just the LOGSAVE record from the SDDS of the send database to the SDDS of the receive database; then uses BLGUT1 to rebuild the SDIDS of the receive database. Use this method if you only want to put the LOGSAVE record from the send database in the receive database. The VSAM key for the LOGSAVE record must be available. (The LOGSAVE record on the receive database must have the same VSAM key as the LOGSAVE record on the send database.)

1. On the send database, do the following tasks:
   a. Issue the BLX-SP FREE command on the send database's SDDS so that no records can be created on the send database.
   b. REPRO just the LOGSAVE record from the send database's SDDS to a sequential data set using IDCAMS.
   c. Issue the BLX-SP REALLOC command on the send database's SDDS.
   d. Do not start send processing, but you may start using your send database for other functions.
2. On the receive database, do the following tasks:
   a. Delete and redefine the receive database's SDIDS.
      1) Issue the BLX-SP FREE command on the receive database's SDIDS.
      2) Use IDCAMS to delete the receive database's SDIDS.
      3) Use IDCAMS to define the receive database's SDIDS.
   b. Issue the BLX-SP FREE command on the receive database's SDDS.
   c. Using IDCAMS, REPRO just the LOGSAVE record from the send SDDS into the receive SDDS.
   d. Issue the BLX-SP REALLOC command on the receive database's SDDS and SDIDS.
   e. Run BLGUT1 to build the receive database's SDIDS.

Updating the LOGSAVE Record for Receive

When you update the LOGSAVE record that you copied from the send database, you must change the **Database type** field. You may also want to change the following fields:

- Data set name
- ABEND disposition
- Time interval in minutes

However, do not change the following fields:

- Logical record length
- Block size
- Synchronize Send and Receive

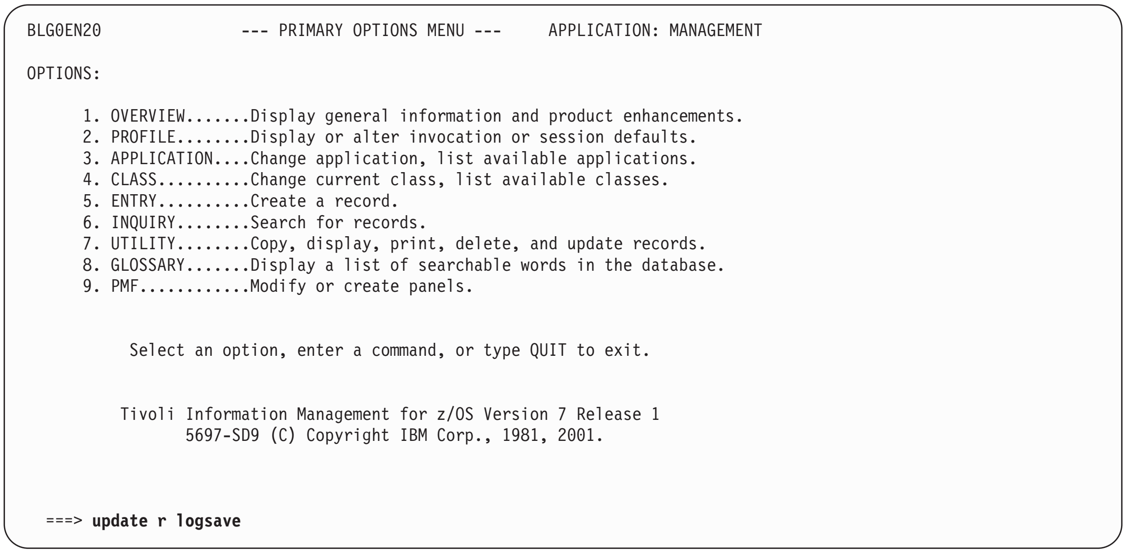
For receive processing, you may also specify values for the **Receive filter prefix** and **Receive filter data** fields. (The LOGSAVE record is automatically filtered out. So if you modify the LOGSAVE record on the send database, those changes are not propagated to the receive database.)

Before you file the record, you must be sure that the record is enabled. If you file the record with its status identified as disabled and then start the BLGLOAD1 TSP, no processing occurs.

To update the LOGSAVE record, start from the Primary Options Menu and type the following IRC:

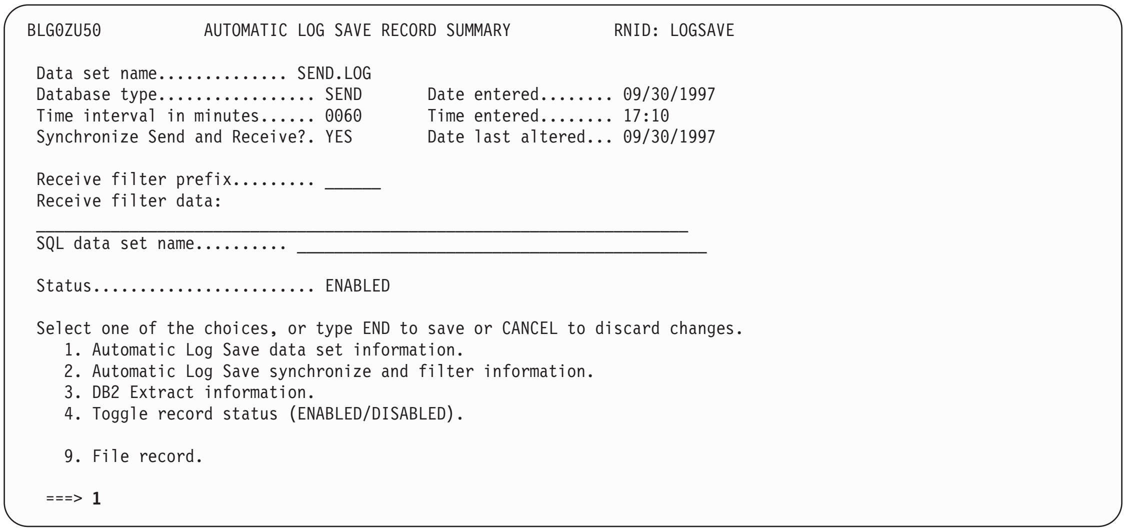
```
update r logsave
```

and press **Enter**.



```
BLG0EN20             --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.



  ===> update r logsave
```

On BLG0ZU50, the Automatic Log Save Record Summary panel, select **Automatic Log Save data set information**. Type **1** and press **Enter**.



```
BLG0ZU50          AUTOMATIC LOG SAVE RECORD SUMMARY          RNID: LOGSAVE

 Data set name............. SEND.LOG
 Database type................ SEND      Date entered........ 09/30/1997
 Time interval in minutes...... 0060      Time entered........ 17:10
 Synchronize Send and Receive?. YES       Date last altered... 09/30/1997

 Receive filter prefix......... ______
 Receive filter data:
 ______________________________________________________________________
 SQL data set name.......... ____________________________________________

 Status....................... ENABLED

 Select one of the choices, or type END to save or CANCEL to discard changes.
    1. Automatic Log Save data set information.
    2. Automatic Log Save synchronize and filter information.
    3. DB2 Extract information.
    4. Toggle record status (ENABLED/DISABLED).

    9. File record.

  ===> 1
```

On the Automatic Log Save Data Set Information Entry panel, you must change the **Database type** field to RECEIVE. You must not change the **Logical record length** or the **Block size** fields.

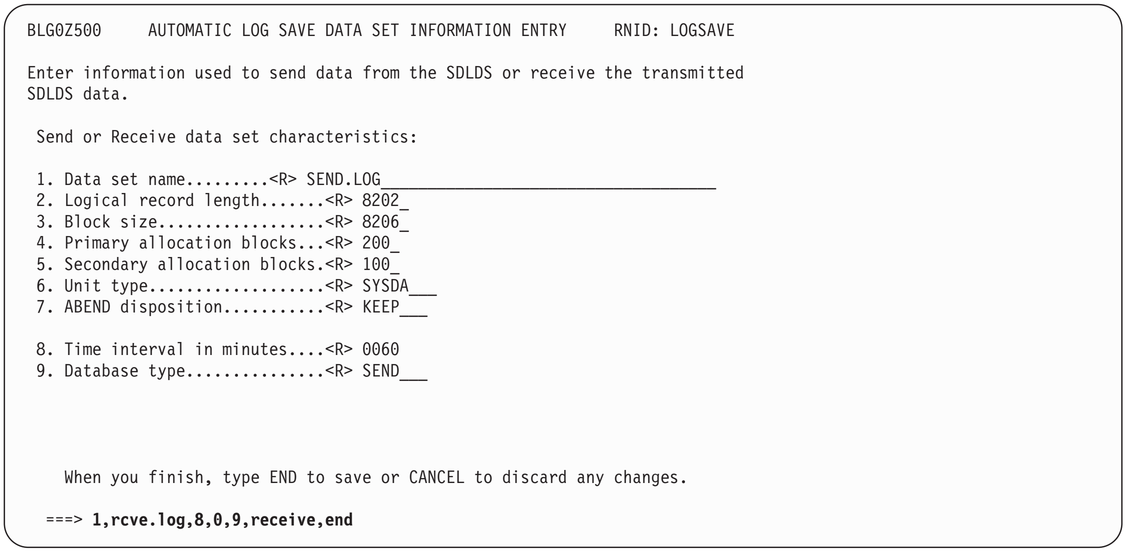
Even though the Automatic Log Save Receive function does not use the **Primary allocation blocks**, **Secondary allocation blocks**, and **Unit type** fields, the LOGSAVE record requires values in the fields.

You may change any of the remaining fields, including the **Data set name** field. The data set name for the receive data set does not have to match the data set name for the send data set; however, the receive data set name does have to match the name you specified in the DELETE step in BLMSARV and in the RCVSND step in BLMSAL1. (See “Preparing BLMSARV” on page 267 and “Preparing BLMSAL1” on page 267.)

The time interval you specify on the LOGSAVE record for the receive database tells user exit BLGUT3WT how long to wait before TSP BLGLOAD1 continues receive processing. If you want the records from the receive data set loaded into your receive database without a delay after BLGLOAD1 TSP starts, change the **Time interval in minutes** field to **0**. You can control how often the send and receive functions are performed by the time value you specify in the LOGSAVE record on the send database. For this example, name the data set **RCVE.LOG** and change the time interval to **0**. Type this IRC:

```
1,rcve.log,8,0,9,receive,end
```

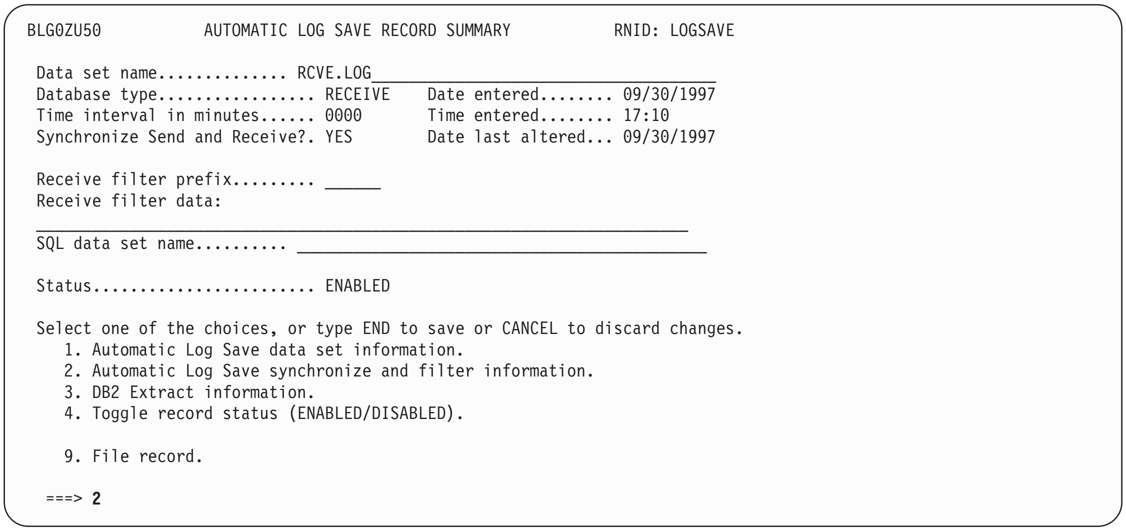
and press **Enter**.



```
BLG0Z500     AUTOMATIC LOG SAVE DATA SET INFORMATION ENTRY     RNID: LOGSAVE

Enter information used to send data from the SDLDS or receive the transmitted
SDLDS data.

 Send or Receive data set characteristics:

 1. Data set name.........<R> SEND.LOG____________________________________
 2. Logical record length.......<R> 8202_
 3. Block size..................<R> 8206_
 4. Primary allocation blocks...<R> 200_
 5. Secondary allocation blocks.<R> 100_
 6. Unit type...................<R> SYSDA___
 7. ABEND disposition...........<R> KEEP___

 8. Time interval in minutes....<R> 0060
 9. Database type...............<R> SEND___




    When you finish, type END to save or CANCEL to discard any changes.

  ===> 1,rcve.log,8,0,9,receive,end
```

The Automatic Log Save Record Summary appears. If you want to specify filtering information, select **Automatic Log Save synchronize and filter information**. Type **2** and press **Enter**.

See “Choosing the Data to Be Propagated” on page 234 for more information about filtering.



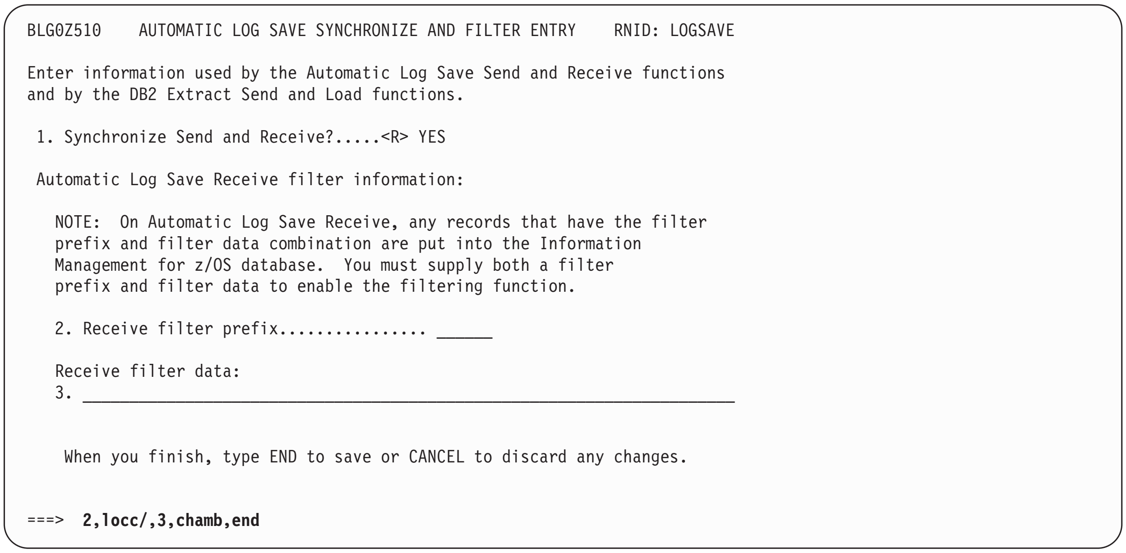
```
BLG0ZU50          AUTOMATIC LOG SAVE RECORD SUMMARY          RNID: LOGSAVE

 Data set name............. RCVE.LOG_______________________________
 Database type................. RECEIVE    Date entered........ 09/30/1997
 Time interval in minutes...... 0000       Time entered........ 17:10
 Synchronize Send and Receive?. YES        Date last altered... 09/30/1997

 Receive filter prefix......... ______
 Receive filter data:
 ______________________________________________________________________
 SQL data set name.......... ____________________________________________

 Status........................ ENABLED

 Select one of the choices, or type END to save or CANCEL to discard changes.
    1. Automatic Log Save data set information.
    2. Automatic Log Save synchronize and filter information.
    3. DB2 Extract information.
    4. Toggle record status (ENABLED/DISABLED).

    9. File record.

  ===> 2
```

Do not change the value in the **Synchronize Send and Receive?** field. The value in this field must be the same on both the send database and the receive database.

You do not want to receive all the records in the send data set. You want to receive only those records that have a location code of CHAMB. To do that, type:

```
2,locc/,3,chamb,end
```

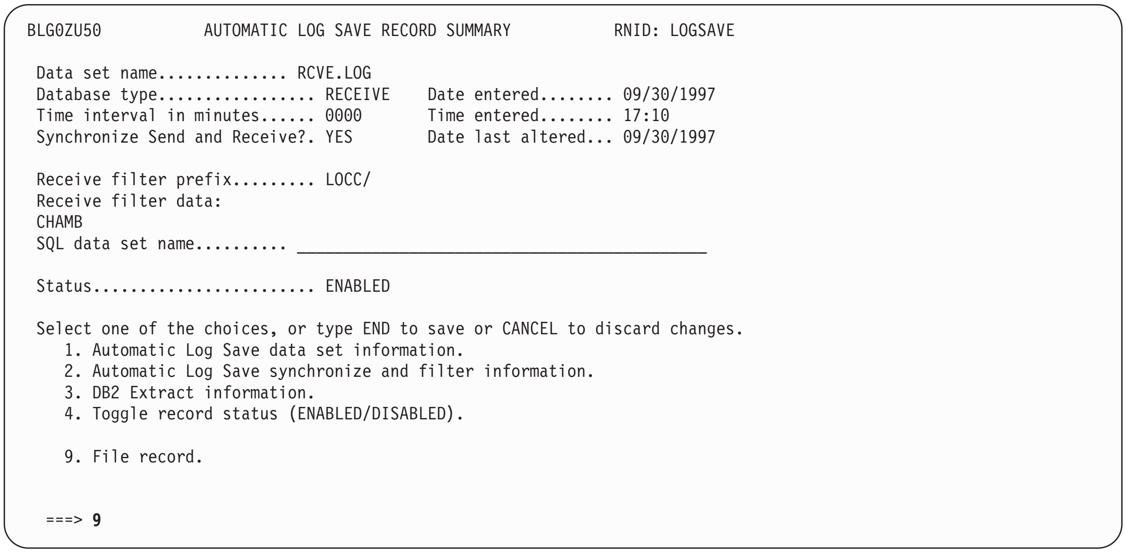
and press **Enter**.



```
BLG0Z510    AUTOMATIC LOG SAVE SYNCHRONIZE AND FILTER ENTRY    RNID: LOGSAVE

Enter information used by the Automatic Log Save Send and Receive functions
and by the DB2 Extract Send and Load functions.

 1. Synchronize Send and Receive?.....<R> YES

 Automatic Log Save Receive filter information:

   NOTE:  On Automatic Log Save Receive, any records that have the filter
   prefix and filter data combination are put into the Information
   Management for z/OS database.  You must supply both a filter
   prefix and filter data to enable the filtering function.

   2. Receive filter prefix................ ______

   Receive filter data:
   3. ______________________________________________________________________


    When you finish, type END to save or CANCEL to discard any changes.


===>  2,locc/,3,chamb,end
```

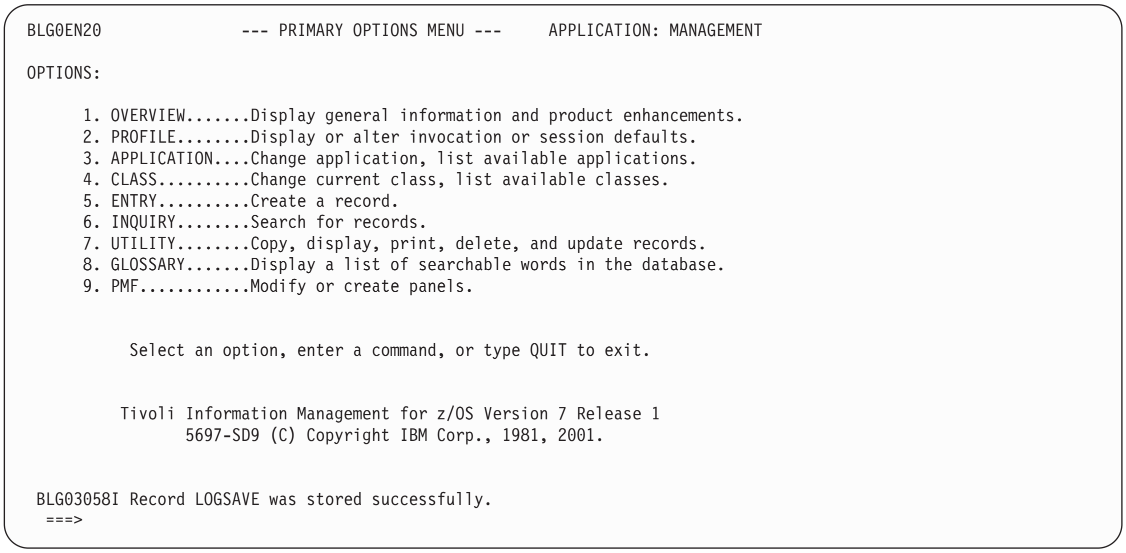
The Automatic Log Save Record Summary appears. The **Status** line says that the record is enabled.

You have finished updating the LOGSAVE record for a receive database. To file the record, type **9** and press **Enter**. Receive processing does not actually occur, however, until you start the BLGLOAD1 TSP to perform Automatic Log Save.



```
BLG0ZU50          AUTOMATIC LOG SAVE RECORD SUMMARY          RNID: LOGSAVE

 Data set name............. RCVE.LOG
 Database type................ RECEIVE    Date entered........ 09/30/1997
 Time interval in minutes...... 0000       Time entered........ 17:10
 Synchronize Send and Receive?. YES        Date last altered... 09/30/1997

 Receive filter prefix......... LOCC/
 Receive filter data:
 CHAMB
 SQL data set name......... ____________________________________________

 Status........................ ENABLED

 Select one of the choices, or type END to save or CANCEL to discard changes.
    1. Automatic Log Save data set information.
    2. Automatic Log Save synchronize and filter information.
    3. DB2 Extract information.
    4. Toggle record status (ENABLED/DISABLED).

    9. File record.


  ===> 9
```

BLG0EN20, the Primary Options Menu, appears with a message that the LOGSAVE record was stored successfully. For reference, print the LOGSAVE record.



```
BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


 BLG03058I Record LOGSAVE was stored successfully.
  ===>
```

Filtering Records

In this example, you specified LOCC/ in the **Receive filter prefix** field and CHAMB in the **Receive filter data** field. Therefore, only those records that contain the prefix and data combination LOCC/CHAMB are converted for storage in the receive Tivoli Information Management for z/OS database. The remaining records are left in the receive data set (RCVE.LOG in this example) where they are deleted during the next receive processing operation.

The LOGSAVE record is automatically filtered out. So if you modify the LOGSAVE record on the send database, those changes are not propagated to the receive database.

Copying the JCL

Copy JCL member BLMSASDA from the MVS image where the send database resides and store it in a data set referenced by step SUBSND in BLMSAL1. (See “Preparing BLMSAL1” on page 267.) You need not change BLMSASDA after you copy it. Do not submit the copied BLMSASDA; BLMSARV submits it automatically after a successful receive operation to start the next offload of the SDLDS.

Starting Normal Send and Receive Processing

To start normal Send and Receive processing, start the Automatic Log Save Facility by submitting JCL BLMSASD on the MVS image where the send database resides. Before you submit JCL BLMSASD, verify that:

- The set up for your send database is complete.
- The set up for your receive database is complete.
- The status of both the send and receive LOGSAVE records is enabled.

Stopping Send and Receive Processing

To stop Automatic Log Save Send and Receive processing, disable the LOGSAVE record on the send database. You can use the following IRC from the Primary Options Menu to toggle the status:

```
update r logsave,4,9
```

When you stop the send function, the receive function also stops. If a send is currently in progress (that is, BLMSASDA or BLMSASD has been started and TSP BLGDUMP1 has called user exit BLGUT4WT and the exit is waiting), the current send finishes waiting. Then the current send offloads the SDLDS and routes BLMSARV for execution. The receive (BLMSARV) processes and submits BLMSASDA for execution on the MVS image where the send database resides. BLMSASDA fails in step OFFLD because of a TSP error indicating that the status of the LOGSAVE record is disabled.

Do not disable the receive LOGSAVE record. If you do, the send occurs, but the send data set will not be successfully received, and you will have to reprocess the send data set.

Using the Automatic Log Save Facility for Automatic Backup

You can use the Automatic Log Save Facility to automatically offload the SDLDS to maintain a backup copy of your production database without propagating your production data to a receive database. Using the Automatic Log Save Facility in this way does not require you to set up a receive database.

To set up the Automatic Log Save Facility for automatic backup only, do the following tasks:

1. Verify that the production database is running with an SDLDS. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for more information.
2. Create a LOGSAVE record for a send database on the production database. See “Creating a LOGSAVE Record for a Send Database” on page 259 for information on this task.
3. Create a GDG for the send data set. See “Creating a GDG” on page 264 for more information on this task.
4. Prepare JCL members BLMSASD and BLMSASDA as described in “Preparing BLMSASD” on page 265 and “Preparing BLMSASDA” on page 266. Make the following additional changes to both JCL members:

   a. Delete steps XMITSND, SPLIT, BLDJOB, SUBRCV, and GOODSUB.
   b. Modify step SUBSND:
      1) On the COND parameter, delete **,(0,EQ,XMITSND),EVEN)**
      2) On the IEFRDER DDname, change the data set name that identifies the JCL member to be submitted if you have copied BLMSASDA to another data set and member.
5. Modify the BLGDUMP1 TSP that offloads the SDLDS if you want to restrict how long it can hold an enqueue on the SDLDS. See “Modifying the BLGDUMP1 TSP” on page 268 for more information on this task.
6. Submit BLMSASD. When it completes successfully, it submits BLMSASDA.

Disabling Automatically Starting the Next Send

If you want to disable automatically starting the next Automatic Log Save Send Function when the Automatic Log Save Receive Function completes successfully, you can delete the step that submits the JCL to start the next send from the JCL for the Automatic Log Save Receive Function. Then, in the JCL you use to start the Automatic Log Save Send Function (either BLMSASDA or BLMSASD) you reference the modified JCL for the Automatic Log Save Receive.

To implement the Automatic Log Save Facility so that it does just one send and one receive, then stops, modify the JCL as follows:

1. Copy or update JCL BLMSAL1. Delete steps SUBSND and GOODSUB.
2. Copy or update JCL BLMSASDA (or BLMSASD if you want to manually delete the send data set after each send and receive) as follows:
   a. Delete step SUBSND so that the next send is not submitted if the send data set is empty.
   b. Reference the copied or updated BLMSAL1 in step COPY2.

   You only need BLMSASDA on the MVS image where the send database resides because you have deleted the step to submit BLMSASDA from the receive JCL.

Each time you submit the modified BLMSASDA, the send data set is deleted and one Automatic Log Save Send Function and one Automatic Log Save Receive Function is performed. JCL to start another Automatic Log Save Send Function is *not* submitted.

Deleting a LOGSAVE Record

If you are propagating data to Tivoli Information Management for z/OS (that is, you are using both the Automatic Log Save Send and Automatic Log Save Receive Functions) and you need to delete the LOGSAVE record on either the send or the receive database, follow the procedures outlined below to preserve the VSAM keys for the LOGSAVE records.

If you are using only the Automatic Log Save Send Function, which is the case if you are propagating data only to DB2, you can delete the LOGSAVE record without regard for preserving the VSAM key.

Deleting a Send LOGSAVE Record

If you must delete the Send LOGSAVE record, decide whether you want to preserve the record's VSAM key. If you do not preserve the VSAM key and you are propagating data to a Tivoli Information Management for z/OS receive database, you will have to rebuild the receive database.

Preserving the LOGSAVE Record's VSAM Key

1. Stop send processing. See "Stopping Send and Receive Processing" on page 276 for details on how to do this.
2. Suspend normal database use.
3. Delete the LOGSAVE record.
4. Recreate the LOGSAVE record.
5. Resume normal database use.
6. Start normal send processing. See "Starting Normal Send and Receive Processing" on page 276.

Without Preserving the LOGSAVE Record's VSAM Key

1. Stop send processing. See "Stopping Send and Receive Processing" on page 276 for details on how to do this.
2. Stop receive processing. See "Stopping Send and Receive Processing" on page 276 for details on how to do this.
3. Delete the send LOGSAVE record.
4. Recreate the send LOGSAVE record.
5. Rebuild the receive database. Use one of the procedures in "Copying the LOGSAVE Record into the Receive Database" on page 270 to do this.
6. Update the LOGSAVE record on the receive database. See "Updating the LOGSAVE Record for Receive" on page 271 for details.
7. Start normal send and receive processing. See "Starting Normal Send and Receive Processing" on page 276.

Deleting a Receive LOGSAVE Record

If you must delete the Receive LOGSAVE record, do the following tasks:

1. Stop send processing. See "Stopping Send and Receive Processing" on page 276 for details on how to do this.
2. Stop receive processing. See "Stopping Send and Receive Processing" on page 276 for details on how to do this.
3. Delete the receive LOGSAVE record.
4. Rebuild the receive database. Use one of the procedures in "Copying the LOGSAVE Record into the Receive Database" on page 270 to do this.
5. Update the LOGSAVE record on the receive database. See "Updating the LOGSAVE Record for Receive" on page 271 for details.
6. Start normal send and receive processing. See "Starting Normal Send and Receive Processing" on page 276.

Recovering from Errors

This section includes the following procedures for error recovery:

- “Error during Send or Receive Processing” outlines a general procedure to follow when an error occurs during send or receive processing.
- “Restoring Synchronization” on page 280 outlines three procedures for restoring synchronization. You can use one of these procedures, or your site-specific procedure, to restore synchronization.

Error during Send or Receive Processing

If an error occurs during send or receive processing, do the following tasks:

1. Stop send and receive processing. See “Stopping Send and Receive Processing” on page 276 for details on how to do this.
2. Diagnose the problem:
   a. Examine the output from the BLMSASD, BLMSASDA, or BLMSARV job to determine the nature of the failure.

   b. Examine the SYSPRINT data set on the send database and on the receive database for any messages.

   c. Examine the return and reason codes from the user exit that failed.

   d. Look up the return and reason codes in the table for the user exit:
      - For user exit BLGUT3EX, see Table 37 on page 323.
      - For user exit BLGUT3wt, see Table 38 on page 325.
      - For user exit BLGUT4ex, see Table 39 on page 327.
      - For user exit BLGUT4wt, see Table 40 on page 328.

      The description column explains the source of the error and indicates what to do to correct the problem.
3. Correct the problem that caused the error.
4. If a send data set was created (the SDLDS was successfully offloaded) but the copy to the GDG failed:
   a. Manually copy the send data set to the GDG.
   b. Prepare and submit job BLMSASDE to process the send data set. See “Preparing BLMSASDE”. If the receive is successful, normal send and receive processing is resumed.
5. If a send data set was created (the SDLDS was successfully offloaded) and a receive job was submitted but did not process or if the onload was unsuccessful:
   a. Prepare and submit job BLMSASDE to process the send data set. See “Preparing BLMSASDE”. If the receive is successful, normal send and receive processing is resumed.

Preparing BLMSASDE

1. Change the job card to your local standards.
2. Change the *node.userid* on the /*ROUTE PRINT card to the node and user ID to receive the job output.
3. In step XMITSND, change *userid.dsname1* to the name of the send data set or generation of a GDG that you want to process.

4. In step COPY1, change:
   a. The JCL data set name to the name of your local data set that contains JCL BLMSARV.
   b. The space DCB parameter on the SYSUT2 DD name to a value large enough to handle splitting each 80-byte record created from the transmit into two 80-byte records (the splitting is performed in step SPLIT).
5. In step COPY2, change the JCL data set name to the name of your local data set that contains BLMSAL1.
6. In step GOODSUB, change:
   a. *sample message* to your message for a successful submission of the receive JCL.
   b. *node.userid* to the destination for the message.

Restoring Synchronization

Synchronization is the process by which the Automatic Log Save Facility ensures that the send data sets are loaded into the receive database in the same order that they were offloaded from the send database. Synchronization can provide additional protection against loading data out of order.

The Automatic Log Save Facility stores the sequence number of the send data set in the following places to assist you in recovering from errors:

- LAST_SDLDS_SENT key in the SDIDS of the send database
- LAST_SDLDS_SENT key in the SDIDS of the receive database
- LAST_SDLDS_SENT record in the send and receive data sets.

See “Maintaining Synchronization” on page 249 for more information about these records and how to determine their values.

When synchronization processing is on (**Synchronize Send and Receive?** field is **YES**), the following processing occurs before records are converted for storage in the receive database:

- The sequence number stored in the LAST_SDLDS_SENT record in the SDIDS of the receive database is read and incremented by 1.
- That this sequence number matches the sequence number in the LAST_SDLDS_SENT record in the receive data set is verified.

If the values do not match, send and receive processing are out of synchronization and receive processing stops.

If the Automatic Log Save Receive function stops because send and receive processing are out of synchronization, user exit BLGUT3EX writes a TSCAFRET of 8 and a TSCAFRES of 20 to the data set allocated to the SYSPRINT DDNAME on the receive database. Normally, this does not happen because the next send is only started after a successful receive. You could, however, inadvertently start another send before a previous send data set is received, for example, while recovering from a receive error. To restore synchronization, you can process the generations of your GDG that were either processed incorrectly or not at all. If you do not have GDGs, you can copy your entire send database over to your receive database and rebuild the SDIDS on the receive database. You can use BLGUT1 to rebuild the SDIDS on the receive database, or you can copy the entire SDIDS from the send database. Using BLGUT1 to rebuild the SDIDS requires copying only the LAST_SDLDS_SENT record from the send database, so your send database is out of production for less time if you use BLGUT1.

You can use one of the following procedures, or a procedure developed and tested at your site, to restore synchronization:

- "Restoring Synchronization with GDG Data Sets". Use this procedure if you are copying send data sets to generations of a GDG.
- "Restoring Synchronization Using BLGUT1" on page 283. Use this procedure if you are not copying the send data set to a generation of a GDG, or if the generations with the correct sequence numbers are not available and you want to minimize lost production time on your send database.
- "Restoring Synchronization without Using BLGUT1" on page 284. Use this procedure if you are not copying the send data set to a generation of a GDG, or if the generations with the correct sequence numbers are not available and you do not want to use BLGUT1.

All of the procedures require you to use IDCAMS. For information about IDCAMS, refer to the following publications:

- *MVS/DFP™ Access Method Services for the Integrated Catalog Facility*
- *MVS/DFP Macro Instructions for VSAM Data Sets*
- *MVS/DFP Using Data Sets*

After restoring synchronization and before resuming normal send and receive processing, determine why synchronization was lost.

Restoring Synchronization with GDG Data Sets

After you complete this procedure, the receive database contains the records from the send data sets that were created by the Automatic Log Save Send Function but that were not successfully received by the Automatic Log Save Receive Function. Also, the value stored in the LAST_SDLDS_SENT record in the receive database matches the value stored in the LAST_SDLDS_SENT record in the send database. Therefore, synchronization is restored.

To use this procedure, you must be saving copies of the send data set in a GDG and all the send data sets that were not successfully processed by the Automatic Log Save Receive function must be available.

Note: The program administrator working on the receive side needs two user IDs available that use the session-parameters member with read/write access to the receive database.

1. On the MVS image where the send database resides, stop processing. See "Stopping Send and Receive Processing" on page 276 for details on how to do this.
2. On the MVS image where the send database resides, check the SYSPRINT output to verify that user exit BLGUT4EX returned a TSCAFRES of 0 and a TSCAFRET of 0. (This verifies that the sequence number was incremented and is 1 greater than the number in the send data set. It also verifies that there was not a problem on the send database that you need to consider.)
3. From the Tivoli Information Management for z/OS Primary Options Menu on the send database, issue the IRC **glossary,,up max, find last** to determine the sequence number stored in the LAST_SDLDS_SENT record of the SDIDS.
4. From the Tivoli Information Management for z/OS Primary Options Menu on the receive database, issue the IRC **glossary,,up max, find last** to determine the sequence

number of the last receive data set that was received and processed by the receive database. This number is stored in the LAST_SDLDS_SENT record of the SDIDS.

5. To determine how many send data sets were not received, subtract the sequence number stored in the LAST_SDLDS_SENT record of the receive database's SDIDS, which you determined in step 4 on page 281, from the sequence number stored in the LAST_SDLDS_SENT record of the send database's SDIDS, which you determined in step 3 on page 281. The result is the number of data sets that have not been processed.

   Using the example in the first row of Table 36, where the sequence number from the send database is 12 and the sequence number from the receive database is 7, you determine that five data sets have not been received.

6. To determine which data sets have not been received, use the value for how many data sets have not been received and the sequence number from the receive database. Using the example in the first row of Table 36, you determine that data sets with sequence numbers 8, 9, 10, 11, and 12 are the five data sets that need to be sent and received to restore synchronization. After the data set with sequence number 12 is processed on the receive database, the sequence number in the receive database will be 12. At that point, the sequence number in the send database will match the sequence number in the receive database, and normal receive and send processing can be resumed. Table 36 gives additional examples of determining how many data sets were not received and their sequence numbers.

Table 36. Examples of Determining How Many Data Sets Were Not Received and Their Sequence Numbers

Sequence number in Send SDIDS	Sequence number in Receive SDIDS	Number of data sets not received	Sequence numbers to receive
12	7	5	8, 9, 10, 11, and 12
9	8	1	9
10	9	1	10
10	8	2	9 and 10

7. On the MVS image where the send database resides, find the GDG data sets that contain the sequence numbers that were not received. For the example in the first row of Table 36, you need to find the data sets that contain the sequence numbers 8, 9, 10, 11, and 12.

   a. Browse the most recently created GDG data set.

   b. Turn on the hex display.

   c. The first record in the data set contains the LAST_SDLDS_SENT record in hex. See "Maintaining Synchronization" on page 249 for details on reading this key.

   d. Convert the hex value to decimal.

8. After you locate the correct GDG data sets:
   a. Copy and modify JCL BLMSAL1 so that it does not submit BLMSASDA after the receive.
   b. Copy and modify JCL BLMSASDE. See "Preparing BLMSASDE" on page 279. Supplement those instructions with the following:

1) In step XMITSND, specify the entire data set name of the lowest sequence number of the GDG, for example, *userid.gdg*.G0001V00, on the TRANSMIT statement. For the example in the first row of Table 36 on page 282, this is the data set that contains the sequence number 8.

2) In step BLDJOB, replace BLMSAL1 with the name you saved JCL BLMSAL1 as in step 8a on page 282.

9. On the receive database, change the value in the **Time interval in minutes** field in the LOGSAVE record to 0, if it is not already 0.
10. Submit the modified version of BLMSASDE to process the specified data set and perform the receive function.
11. Check the job output for errors. If the job output contains no errors, receive processing for the first GDG data set is finished and the sequence number in the receive database was incremented.
12. Modify and submit BLMSASDE again. This time specify the next GDG data set. For the example in the first row of Table 36 on page 282, this is the data set that contains the sequence number 9.
13. Repeat steps 11 and 12 until all the GDG data sets needed to restore synchronization have been processed. For the example in the first row of Table 36 on page 282, this is 3 more times to process the data sets that contain sequence numbers 10, 11, and 12.
14. To verify that the LAST_SDLDS_SENT records on the receive database and the send database both contain the same value, issue the GLOSSARY command on both the send database and the receive database as you did in steps 3 and 4 on page 281. For the example in the first row of Table 36 on page 282, both records should now contain 12.
15. On the receive database, change the value in the **Time interval in minutes** field in the LOGSAVE record to the value for normal receive processing.
16. Start normal send and receive processing. See “Starting Normal Send and Receive Processing” on page 276.

Restoring Synchronization Using BLGUT1

After you complete this procedure the receive database contains copies of the records that were in the send database at the time the REPRO was run. Also, the value stored in the LAST_SDLDS_SENT record in the receive database matches the value stored in the LAST_SDLDS_SENT record in the send database. Therefore, synchronization is restored.

1. Stop send processing. See “Stopping Send and Receive Processing” on page 276 for details on how to do this.
2. On the receive database, print the LOGSAVE record. You will use this print out in step 4h on page 284 to recreate the LOGSAVE record.
3. On the MVS image where the send database resides, do the following tasks:
   a. Issue the BLX-SP FREE command on the send database’s SDDS and SDIDS so that no records can be created on the send database. Refer to the *Tivoli Information Management for z/OS Operation and Maintenance Reference* for information on the BLX-SP commands.
   b. REPRO the send database’s SDDS.
   c. REPRO the LAST_SDLDS_SENT record from the SDIDS. You can use the following IDCAMS control statements to copy the LAST_SDLDS_SENT record.

```
REPRO IDS('your.SDIDS') ODS('your.outrec') +
      FROMKEY(X'BAD3C1E2E36DE2C4D3C4E26DE2C5D5E35C') +
      COUNT(1)
```

d. Issue the BLX-SP REALLOC command on the send database’s SDDS and SDIDS.

e. Do not start send processing, but you may start using your send database for other functions.

4. On the MVS image where the receive database resides, do the following tasks:

a. Delete and redefine the receive database’s SDIDS and SDDS.
1) Issue the BLX-SP FREE command on the receive database’s SDDS and SDIDS.
2) Use IDCAMS to delete the receive database’s SDIDS and SDDS.
3) Use IDCAMS to define the receive database’s SDIDS and SDDS.

b. REPRO just the SDDS from the send database into the receive database.

c. Issue the BLX-SP REALLOC command on the receive database’s SDDS and SDIDS.

d. Run BLGUT1 on the receive database’s SDIDS. Refer to the *Tivoli Information Management for z/OS Operation and Maintenance Reference* for more information.

e. Issue the BLX-SP FREE command on the receive database’s SDIDS.

f. REPRO the LAST_SDLDS_SENT record from the send database into the receive database’s SDIDS. (This is the record you copied in step 3c on page 283.) You have to do this step because BLGUT1 does not create the LAST_SDLDS_SENT record and because you need the value stored in this record on the receive database to match the value stored in this record on the send database.

g. Issue the BLX-SP REALLOC command on the receive database’s SDIDS.

h. Update the receive database’s LOGSAVE record. Change the **Database type** field from **send** to **receive**. Refer to the print out of your LOGSAVE record, which you made in step 2 on page 283, to determine whether you need to make any other changes.

5. Start normal send and receive processing. See “Starting Normal Send and Receive Processing” on page 276.

Restoring Synchronization without Using BLGUT1

After you complete this procedure the receive database contains copies of the records that were in the send database at the time the REPRO was run. Also, the value stored in the LAST_SDLDS_SENT record in the receive database matches the value stored in the LAST_SDLDS_SENT record in the send database. Therefore, synchronization is restored.

To restore database synchronization, do the following

1. Stop send processing. See “Stopping Send and Receive Processing” on page 276 for details on how to do this.

2. On the receive database, print the LOGSAVE record. You will use this print out in step 4d on page 285 to recreate the LOGSAVE record.

3. On the MVS image where the send database resides, do the following tasks:

a. Issue the BLX-SP FREE command on the send database’s SDDS and the SDIDS so that no records can be created on the send database. Refer to the *Tivoli Information Management for z/OS Operation and Maintenance Reference* for information on the BLX-SP commands.

   b. REPRO the send database's SDDS and SDIDS.

   c. Issue the BLX-SP REALLOC command on the send database's SDDS and SDIDS.

   d. Do not start send processing, but you may start using your send database for other functions.

4. On the MVS image where the receive database resides, do the following tasks:

   a. Delete and redefine the receive database's SDIDS and SDDS.
      1) Issue the BLX-SP FREE command on the receive database's SDDS and SDIDS.
      2) Use IDCAMS to delete the receive database's SDIDS and SDDS.
      3) Use IDCAMS to define the receive database's SDIDS and SDDS.

   b. REPRO the SDDS and SDIDS from the send database into the receive database.

   c. Issue the BLX-SP REALLOC command on the receive database's SDDS and SDIDS.

   d. Update the receive database's LOGSAVE record. Change the **Database type** field from `send` to `receive`. Refer to the print out of your LOGSAVE record, which you made in step 2 on page 284, to determine whether you need to make any other changes.

5. Start normal send and receive processing. See "Starting Normal Send and Receive Processing" on page 276.

15

Propagating to a DB2 Database

This information is for the program administrator responsible for implementing the DB2 Extract Facility. You must understand Terminal Simulator Panels (TSPs), the Panel Modification Facility (PMF), and BLGUT8 to perform the tasks described in this chapter. For information about TSPs, refer to the *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference*. For information about PMF, refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide*. For information about BLGUT8, refer to the *Tivoli Information Management for z/OS Operation and Maintenance Reference* .

Some of the tasks described in this chapter also require a knowledge of DB2. Work with your DB2 database administrator to accomplish those tasks. You may need to refer to the following books:

- *IBM Database 2 Version 3 Administration Guide*
- *IBM Database 2 Version 3 SQL Reference*
- *IBM Database 2 Version 3 Messages and Codes*
- *IBM Database 2 Version 3 Command and Utility Reference*

Implementing the DB2 Extract Facility

Before you implement the DB2 Extract Facility, you must understand the process and perform the following tasks:

1. Read “Propagating Tivoli Information Management for z/OS Data” on page 231 for a discussion of data propagation and an overview of the LOGSAVE record.
2. Read this chapter for an overview of DB2 Extract Processing and the implementation tasks.
3. Decide how your installation will use the DB2 Extract Facility:
   - How often do you want to offload the SDLDS of the send database?
   - Do you want to change how long an enqueue on the SDLDS is held?
   - Which Tivoli Information Management for z/OS records and fields will you propagate to DB2?
   - Which user IDs will receive status messages indicating whether the update step of the DB2 Update Utility processed successfully?
   - Which user ID will the job output from the DB2 Update Utility be routed to?
   - How often will you review the status messages from DB2 and how will you handle error situations?
   - Will you maintain copies of the offloaded SDLDS (send data set) in a generation data group (GDG)?

- Will you maintain copies of the SQL data set in a GDG?
- If you maintain GDGs, how many backup levels will you keep to ensure adequate data integrity? Consider the frequency of send processing in setting this value.
- Do you want to synchronize the DB2 Extract Send and Update Utilities?
- Do you want to disable automatically starting the next DB2 Extract Send operation when the DB2 Update completes successfully?
- What procedures will you use for backup and disaster recovery?
- How will the automatic offload of the SDLDS fit in with your current database recovery procedures?
- Do you want to propagate Tivoli Information Management for z/OS data to both DB2 and Tivoli Information Management for z/OS? If so, how will your Automatic Log Save processing fit in with your DB2 Extract processing?

4. Develop and test your backup and disaster recovery procedures.
5. Develop procedures for monitoring the operation of the DB2 Extract Facility.

Understanding DB2 Extract Send Processing

The DB2 Extract Send Utility reads the LOGSAVE record to obtain the following information:

- Sequential data set characteristics for the Automatic Log Save send data set and for the SQL data set, including:
  - Data set name
  - Logical record length
  - Block size
  - Primary allocation blocks
  - Secondary allocation blocks
  - Unit type
  - ABEND disposition
- Database type
- How long to wait between send processing operations
- Whether to tag the offload data set with a sequence number so that the DB2 Extract Send and Update Utilities can be synchronized
- Authorization ID of the DB2 tables
- Names of the RDMTs that map the Tivoli Information Management for z/OS records to DB2
- Whether send processing is enabled or disabled.

The DB2 Extract Send Utility is started by JCL BLMSDSD, which does the following tasks:

1. Starts Tivoli Information Management for z/OS in batch mode and starts TSP BLGTDBXM, which does the following tasks. See “Data Propagation User Exits” on page 323 for more information about the user exits.
   a. Calls user exit BLMSSGEN to setup the environment and read the RDMTs that were created by BLGUT8 and stored in a data set in the BLGRFT concatenation.
   b. Calls user exit BLGUT4WT to:

1) Read the LOGSAVE record.
2) Verify that the LOGSAVE record is for a send database.
3) Verify that the status of the LOGSAVE record is enabled.
4) Put selected information from the LOGSAVE record into the TSCA variable data area.
5) Wait the length of time specified in the LOGSAVE record; then return to the TSP.

c. Call user exit BLGUT4EX to:
   1) Read information from the TSCA variable data area.
   2) Allocate the send data set.
   3) If you are synchronizing the send and receive functions:
      a) Obtain the sequence number from the SDIDS.
      b) Add 1 to the sequence number.
      c) Write the sequence number as the first record in the send data set.
   4) Move records from the SDLDS into the send data set and empty the SDLDS.
   5) If you are synchronizing the Send and Update Utilities, store the sequence number in the SDIDS.

d. Calls user exit BLMSSGEN to do the following until all the records in the send data set have been read:

   1) Read a record from the send data set.

   2) Match a record type s-word from an RDMT to the record. Use the first RDMT for which the record type s-word is found in the Tivoli Information Management for z/OS record.

   3) If no RDMT is found that maps the record, go to the next record.

   4) If a match, build SQL statements for the record using the RDMT to determine which fields to extract and the information that is needed to construct the SQL statement.

   5) Write the SQL statements to the SQL data set.

e. If no SQL was built and no errors occurred, submits JCL BLMSDSD to begin another send.

f. Submits JCL BLMSSND, which builds and submits jobstream BLMSRCV by concatenating the following pieces:
   1) JCL member BLMSRCV
   2) The SQL statements from the SQL data set as in stream data
   3) JCL member BLMSL1.

   The resultant job (BLMSRCV) is submitted to the internal reader and routed to the MVS image specified on the /*ROUTE XEQ card, where it runs the SQL in batch mode to update the DB2 database. See “Understanding DB2 Update Processing” on page 290 for a discussion of the processing done by BLMSRCV.

g. Calls user exit BLMSSGEN to cleanup the environment.

h. Submits JCL BLMSDCS to copy the send data set to a GDG based upon the success or failure of step 1c.

i. Send is complete.

2. Sends a status message about the success or failure of the TSP to the user ID specified in the JCL.

Understanding DB2 Update Processing

The DB2 Update Utility uses BLMSRCV, the JCL generated and submitted by the DB2 Extract Send Utility. This JCL does the following tasks:

1. Starts DSNTEP2, a PL/I program provided by DB2. This program:
   - Runs the in-stream SQL statements in batch mode to update the data in the DB2 database.
   - If you leave SYSPRINT DD SYSOUT=* in the JCL, messages are written to SYSPRINT on the MVS image where DB2 resides. If you specified a data set for this DDname, writes messages to that data set on the MVS image where DB2 resides. Refer to the prologue of DSNTEP2 for a list of the messages it returns.
2. Sends a status message to the user ID specified in the JCL.
3. Submits JCL BLMSDSD to begin the next send. This job is routed to the MVS image where the send database resides for execution.
4. Routes the job output to the user ID specified on the /*ROUTE PRINT card in the JCL.

Setting Up the DB2 Extract Facility

To propagate Tivoli Information Management for z/OS data only to a DB2 database, do the following tasks. You can do these tasks in any order you choose. You must do all of them, however, before you start TSP BLGTDBXM.

1. Be sure the Tivoli Information Management for z/OS database is running with an SDLDS.
2. Build relational data mapping tables (RDMTs) to map Tivoli Information Management for z/OS records to DB2. See "Building Relational Data Mapping Tables" on page 291 for information on this task.
3. Define the LOGSAVE record for DB2 Extract processing. See "Defining a LOGSAVE Record for DB2 Extract Processing" on page 298 for information on this task.
4. Establish the DB2 environment. See "Establishing the DB2 Environment" on page 301 for information on this task.
5. On the MVS image where DB2 resides, locate and modify your local copy of DSNTEP2. See "Modifying DSNTEP2" on page 304 for information on this task.
6. Create GDGs. See "Creating GDGs" on page 305 for information on this task.
7. Prepare the JCL that is used for DB2 Extract processing. See "Preparing the JCL" on page 305 for information on this task.
8. Modify the TSP (BLGTDBXM) that performs DB2 Extract processing. See "Modifying the DB2 Extract TSP" on page 310 for information on this task.
9. Submit JCL BLMSDSD to initiate data propagation to DB2. See "Starting Send and Update Processing" on page 312 for more information on this task.

If you are propagating data to both Tivoli Information Management for z/OS and DB2, you must make some additional changes to TSP BLGTDBXM and to the JCL for the Automatic Log Save Facility and the DB2 Extract Facility. Those changes are described in "Propagating Data to Both Tivoli Information Management for z/OS and DB2" on page 316.

Building Relational Data Mapping Tables

Use BLGUT8 to build tables that map Tivoli Information Management for z/OS panels and fields to relational database tables and columns. These tables are called *relational data mapping tables* (RDMTs). The DB2 Extract Send Utility uses RDMTs to construct the SQL statements for propagating Tivoli Information Management for z/OS data to DB2.

BLGUT8 is used to build tables for both the DB2 Extract Facility and the Tivoli Information Management for z/OS application program interfaces (APIs). If you are building an RDMT, any keywords that apply only to low-level application program interface (LLAPI) program interface data tables (PIDTs) are ignored. If you are building a PIDT, any DB2 Extract Facility related keywords are ignored. Therefore, if you are using both the DB2 Extract Facility and one of the APIs, you can use the same input statements to build both PIDTs and RDMTs. Refer to the *Tivoli Information Management for z/OS Application Program Interface Guide* for details on using BLGUT8 and the syntax for creating PIDTs. The BLGUT8 syntax described here applies only to building RDMTs.

If you add or delete fields, or change their lengths, in the Tivoli Information Management for z/OS record after you create an RDMT for it, consider whether you need to rerun BLGUT8 to update the Tivoli Information Management for z/OS to DB2 map. If the map changes, you need to consider altering, or deleting and re-creating, the DB2 tables to conform to the new format for the Tivoli Information Management for z/OS record.

BLGUT8 Syntax

The TABLE statement accepts SQLMAP as a value for USE. Specifying USE(SQLMAP) tells BLGUT8 to generate an RDMT for each TABLE and ETABLE pair you specify as input.

The FIELD statement accepts the keywords SQLTAB and SQLCOL. These keywords are ignored if USE(SQLMAP) is not specified. If USE(SQLMAP) is specified, keywords that are not used, such as NOTLOGIC and REQUIRED, are ignored. This enables you to use the same statements to create either PIDTs for the API or RDMTs for the DB2 Extract Facility. Refer to the *Tivoli Information Management for z/OS Application Program Interface Guide* for information on creating PIDTs.

To create an RDMT, specify USE(SQLMAP) on the TABLE statement. Specify the name of the DB2 table with the SQLTAB keyword on the FIELD statement. Specify the DB2 column with the SQLCOL keyword on the FIELD statement.

TABLE/ETABLE Statement

You can specify multiple RDMTs in one input stream by using the TABLE and ETABLE combination. SQLMAP statements make up a record and specify the mapping of the record to tables and columns in a relational database.

Keywords on the TABLE statement specify table attributes, and the TABLE and ETABLE combination defines the boundaries of a table specification. Each table specification corresponds to a record or search argument definition and generates one data table and one validation table. All other statements must lie within a TABLE and ETABLE combination.

The syntax of the TABLE statement is:

```
[ label: ]     TAble  Name(name)  Use {(Inquiry)  }
                                      {(Retrieve) }
                                      {(Create)   }
                                      {(Update)   }
                                      {(Add)      }
                                      {(ALias)    }
                                      {(Header)   }
                                      {(Sqlmap)   }
          .
          .
          .
[label:]       Etable
```

Name
Identifies the name to be given to the table. This name also serves as the PDS member name. You can use up to 8 characters for the table name when the value specified with Use is ALias or Sqlmap. You can use up to 7 characters for all other tables. Two member names are determined by this specification. The PIDT is named with the value for this keyword, and the PIPT name is this value with a P appended. This keyword determines PIDTNAME, PIDTPTNM, and PIPTNAME.

Use
Identifies the function for which the table is used. The value specified with this keyword determines how the APIs set PIDTUSEF for all USE keyword values except ALias and Sqlmap. Use(ALias) causes the Table Build Utility to process an alias table only. Use(Sqlmap) causes the Table Build Utility to process a relational data mapping table only. Use(Header) causes the Table Build Utility to process header rows only.

Inquiry
For fields to include in a search argument.

Retrieve
For fields to retrieve from an existing record.

Create
For fields to include in a new record.

Update
For fields to update in an existing record.

Add
For record relation fields to add to an existing record.

ALias
For fields to collect for an alias table. You can use only ALias statements when you specify Use(ALias).

Header
For fields to create a model PIDT containing only header rows. This PIDT is for use in generating dynamic PIDTs on the retrieve transaction (T100). A FIELD statement specified with Use(Header) is ignored.

Sqlmap
For fields to map to relational database tables and columns.

FIELD Statement

The FIELD statement defines the Tivoli Information Management for z/OS fields that are to be extracted from the log data set (the SDLDS) and mapped to DB2 tables and columns. A field is any attribute that can be assigned to a record type and is characterized by an s-word or a p-word, or both. Examples are reporter name, record ID, freeform text description, and date entered.

For a Use (Sqlmap table, do not include a FIELD statement for the RNID. The DB2 Send Utility automatically generates SQL statements for this field.

The syntax of the FIELD statement as used by the DB2 Extract Utility is:

```
[label:] Field  Panel(panel)  Index(dictionary index key)

             [SQLTAB(table-name)] [SQLCOL(column-name)]

             [RCdsword {(Yes)}]   [List {(Yes)}] [Text {(Yes)}]
             [         {(No) }]   [     {(No) }] [     {(No) }]
```

If you omit a keyword from the FIELD statement, the default for the keyword is **NO**. However, if you include the keyword in your statement, you *must* also enter **YES** or **NO**.

Panel

Identifies the panel in which the Tivoli Information Management for z/OS field is located. This keyword is required.

panel

An 8-character alphanumeric string. The first character must be alphabetic (A-Z). This string names a member of a panel data set.

Index

Identifies the Tivoli Information Management for z/OS field. This keyword is required.

dictionary index key

The character P or S followed by 4 hexadecimal characters, which identify a p-word or an s-word, respectively, in the dictionary data set or specified panel. This s-word index or p-word index must be in the panel named by the PANEL keyword unless you specify an assisted-entry panel with its **Collect From Caller** field set to **YES**. In this case, the s-word is obtained from the dictionary.

For control panels, the DB2 Extract Facility uses only ADD control lines. Within these control lines, it uses only p-words with literal p-word validation data (pfx/<data> or <data>).

When you specify an s-word index key and control panel, the DB2 Extract Facility retrieves corresponding literal prefix validation data, if available, along with the s-word. If you specify an s-word index key and an assisted-entry panel, the DB2 Extract Facility uses the first p-word that appears in the validation section of the specified assisted-entry panel to define the field.

For a Use (Sqlmap table, BLGUT8 suppresses any entry that it determines does not have actual data. Two examples of this are:

1. An s-word that defines a visible phrase from a selection or data-entry panel
2. An s-word that is added by a control panel with no corresponding data.

Other cases may cause this situation. These entries are not listed in the report and no SQL is generated by the DB2 Extract Send Utility for them. All entries put into the RDMT will have SQL generated for them. If a MAXL of 0 appears in the output, BLGUT8 did not have enough information to determine what the maximum length for the field is. You must manually determine what the maximum length of the field is by looking at the panel that collects the data.

The DB2 Extract Send Utility tries to find the data for an RDMT entry in the Tivoli Information Management for z/OS record. If it does not find any data for the entry, it uses the MAXL field to determine how many blanks to use as the data. If MAXL is 0, it supplies 1 blank as the data for the field.

You can only specify a p-word index key (Pxxxx) for an assisted-entry or control panel. Specify a p-word key only if you are describing a field that does not have an s-word.

Specify each index keyword value only once within a TABLE and ETABLE section.

SQLTAB

Identifies the DB2 table in which the Tivoli Information Management for z/OS field is to be placed. If this keyword is omitted and the LIST(Y) or TEXT(Y) is not specified, the DB2 table to which the field is mapped defaults to the value of NAME given on the TABLE statement. If this keyword is omitted and the LIST(Y) or TEXT(Y) is specified, the table name defaults to the value of NAME given on the TABLE statement concatenated with an underscore character followed by the dictionary index key.

table-name

This string names the DB2 table in which this Tivoli Information Management for z/OS field is to be placed. This can be from 1 to 18 characters long. The data that you enter in this field is not validated. Ensure that this field contains a valid DB2 table name. If the name is not valid, the DB2 Update Utility processes with errors.

SQLCOL

Identifies the DB2 column in which the Tivoli Information Management for z/OS field is to be placed. If this keyword is omitted, the column to which the field is mapped defaults to the dictionary index key.

column-name

This string names the DB2 column in which this Tivoli Information Management for z/OS field is to be placed. This can be from 1 to 18 characters long. The data that you enter in this field is not validated. Ensure that this field contains a valid DB2 column name. If the name is not valid, the DB2 Update Utility processes with errors. You must map each list and text entry to a unique table.

List

Identifies list processing for the field.

Yes

Indicates a list field. Specify **List(Y)** only when the Panel keyword value specifies an assisted-entry panel and the **I**ndex keyword value is a s-word index (Sxxxx) that refers to an s-word of no more than 8 characters. **List(Y**) is not allowed with **Text(Y)** or **RCsword(Y**).

No (default)

Indicates a nonlist field.

If you omit this keyword, the default is NO. If you include the keyword, specify either **YES** or **NO**.

Text

Identifies a freeform text field.

Yes

Indicates a freeform text field. Specify **Text(Y)** only when the Panel keyword value names a selection, option, control, or data-entry panel. **Text(Y)** is not valid when the Index keyword value represents a p-word index (Pxxxx). Do not use **Text(Y)** with **List(Y)** or **RCdsword(Y)**.

No (default)

Indicates a field that does not identify freeform text.

If you omit this keyword, the default is NO. If you include the keyword, specify either **YES** or **NO**.

RCdsword

Identifies the field that defines the record type.

Yes

Indicates a record type field. Specify **RCdsword(Y)** only when the Panel keyword value names a selection, option, data-entry, or control panel. **RCdsword(Y)** is not valid when the Index keyword value represents a p-word index (Pxxxx).

You must specify **RCdsword(Y)** for one and only one Sqlmap statement within a table specification. **RCdsword(Y)** is not allowed with **List(Y(** or **Text(Y)**.

No (default)

Indicates a field that does not define the record type.

If you omit this keyword, the default is NO. If you include the keyword, specify either **YES** or **NO**.

RDMT Build Job Stream Example

The following example shows how to use BLGUT8 to build an RDMT named MYMAP. In this example, the RDMT is stored in a partitioned data set named XXX.INFOV6.SBLMFMT.PDS referenced by the DDname BLGRFT. The actual DDname to use is the one your session-parameters member uses for the RFT data set. Only a portion of the problem record type fields are included.

In the job stream, DDname BLGPNLS references the panel data set that was created by using BLGUT6 to copy the panels into a partitioned data set. DDname BLGDICT references the dictionary data set.

The sample library SBLMSAMP contains input statements that build PIDTs. You can modify these statements to build RDMTs.

```
//TABLEO   JOB
//STEP1    EXEC PGM=BLGUT8,REGION=2048K
//STEPLIB  DISP=SHR,DSN=BLM.SBLMMOD1
//BLGDICT  DISP=SHR,DSN=BLM.DICT
//BLGPNLS  DISP=SHR,DSN=BLM.SBLMPNLS
//BLGRFT   DISP=OLD,DSN=BLM.TABLES.PDS
//SYSPRINT DD SYSOUT=A
//* 'DLM=%%' CHANGES END-OF-STREAM DELIMITER FROM '/*' to '%%'
//* OTHERWISE, COMMENT STARTED IN FIRST COLUMN TERMINATES INPUT STREAM
//SYSIN    DD *,DLM='%%'
TABLE NAME(MYMAP) USE(SQLMAP);

FIELD PANEL(BLG00000) INDEX(S0032)  /*  PROBLEM RECORD TYPE        */
      RCDSWORD(Y);

FIELD PANEL(BLG6REQN) INDEX(S0B59)  /*  REPORTER NAME              */
      SQLTAB(PROBLEM_DATA) SQLCOL(REPORTER_NAME);

FIELD PANEL(BLG6PTYP) INDEX(S0C09)  /*  PROBLEM TYPE               */
      SQLTAB(PROBLEM_DATA) SQLCOL(PROBLEM_TYPE);
```

```
FIELD PANEL(BLG6RQDP) INDEX(S0B9B)  /*  REPORTER DEPARTMENT        */
      SQLTAB(PROBLEM_DATA) SQLCOL(REPORTER_DEPT);

FIELD PANEL(BLG6STAT) INDEX(S0BEE)  /*  REPORTER STATUS            */
      SQLTAB(PROBLEM_DATA) SQLCOL(REPORTER_STAT);

FIELD PANEL(BLG6PHON) INDEX(S0B2D)  /*  REPORTER PHONE             */
      SQLTAB(PROBLEM_DATA) SQLCOL(REPORTER_PHONE);

FIELD PANEL(BLG6PRII) INDEX(S0BE6)  /*  INITIAL PRIORITY           */
      SQLTAB(PROBLEM_DATA) SQLCOL(INITIAL_PRIORITY);

FIELD PANEL(BLG6DSAB) INDEX(S0E0F)  /*  DESCRIPTION                */
      SQLTAB(PROBLEM_DATA) SQLCOL(DESCRIPTION);

FIELD PANEL(BLG0B010) INDEX(S0E01)  /*  TEXT, DESCRIPTION          */
      SQLTAB(PROBLEM_TEXT) SQLCOL(TEXT)
      TEXT(Y);

ETABLE;                             /*  END TABLE                  */
%%
//
```

RDMT Job Stream Sample Output

The following example shows the output that results from a successful run of BLGUT8. Notice that for this example there is no output entry for the FIELD statement defining the record type s-word. This is because the field is a visible phrase type field and does not have any data; therefore, the field is not mapped to DB2. All entries in the output report will have SQL generated for them. In addition, SQL will be generated for the RNID. See “Example of Propagating Problem Data to DB2” on page 313 for examples of the DB2 tables this RDMT maps to.

```
NO ERRORS DETECTED DURING TABLE BUILD PROCESSING FOR DATA TABLE: MYMAP AND PATTERN TABLE: MYMAPP
DATE: 10/27/1999  TIME: 16:17:49                               PROGRAM INTERFACE TABLE BUILD UTILITY

FIELD SPECIFICATION SUMMARY FOR RELATIONAL DATA MAPPING TABLE (RDMT): MYMAP
   FIELDS DEFINED: 9  RECORD TYPE SWORD: Z//S/TXS
HEADING DESCRIPTIONS: F=FIELD TYPE C=SEARCHABLE L=TABLE LIST ITEM R=RECORD TYPE SWORD D=DATE T=PANEL TYPE J
                       S=STRING DATA  MAXL=MAX RESPONSE LENGTH  X=DATA TYPE  M=REPLACEABLE
INDEX   SWORD     PANEL         SQLTAB             SQLCOL       T PREFIX X F MAXL D C J L R S M
_____________________________________________________________________________________________________
S0B59 XIM0I0PS00 BLG6REQN PROBLEM_DATA       REPORTER_NAME      A PERS/  M R   15 . Y . . . . Y
S0C09 XIMGCT0000 BLG6PTYP PROBLEM_DATA       PROBLEM_TYPE       A TYPE/  M R    8 . Y O . . . Y
S0B9B XIM0I0GS00 BLG6RQDP PROBLEM_DATA       REPORTER_DEPT      A GROS/  M R   11 . Y . . . . Y
S0BEE XIMS0SSC00 BLG6STAT PROBLEM_DATA       REPORTER_STAT      A STAC/  M R    7 . Y O . . . Y
S0B2D XIM00NPS00 BLG6PHON PROBLEM_DATA       REPORTER_PHONE     A PH/    M R   13 . Y . . . . Y
S0BE6 XIMS0SPI00 BLG6PRII PROBLEM_DATA       INITIAL_PRIORITY   A PRII/  M R    2 . Y O . . . Y
S0E0F XIM0TXCA00 BLG6DSAB PROBLEM_DATA       DESCRIPTION        A        M R   45 . Y . . . Y Y
S0E01 XIM0TX0S00 BLG0B010 PROBLEM_TEXT       TEXT               S        . X  250 . Y . . . . .
BLG21900I Mapping table MYMAP was successfully written to table data set XXX.INFOV6.SBLMFMT.PDS
BLGUT8 - PROGRAM INTERFACE TABLE BUILD UTILITY COMPLETED
```

Using the RDMT Output

Use the RDMT output when you create the DB2 tables. Use:

- The table names listed in the SQLTAB column to name the DB2 tables when you create them.
- The column names listed in the SQLCOL column to name the columns in the DB2 tables when you create them.
- The lengths listed in the MAXL column to determine the lengths for the DB2 columns.
- Use the data types listed in the column labeled X to determine the data types for the DB2 columns. Possible data types are as follows:

  D Double-byte character set (DBCS)
  M Mixed data
  S Single byte character set (SBCS)

  Use this information to determine the data type for the DB2 column to which this field is mapped. If this column contains a D, map the field to a DB2 column with a data type for double-byte characters. If this column contains an M, consider whether to map the field to a DB2 column with a data type for double-byte characters or single-byte characters. (The M does not mean that the field contains mixed data, only that it can.) Refer to the *IBM Database 2 Version 3 SQL Reference* for information on DB2 data types.

The DB2 table and column names from this report must match the table and column names in the DB2 database. A mismatch will cause the DB2 Update Utility to fail.

“Create Tables to Receive Tivoli Information Management for z/OS Data” on page 302 contains more information about how to use this report.

DBCS and Mixed Data Considerations

Tivoli Information Management for z/OS and DB2 both support DBCS data. DSNTEP2, as it is supplied by DB2, does not. If your Tivoli Information Management for z/OS data contains DBCS data, you must modify DSNTEP2 so that it can handle DBCS data.

Defining a LOGSAVE Record for DB2 Extract Processing

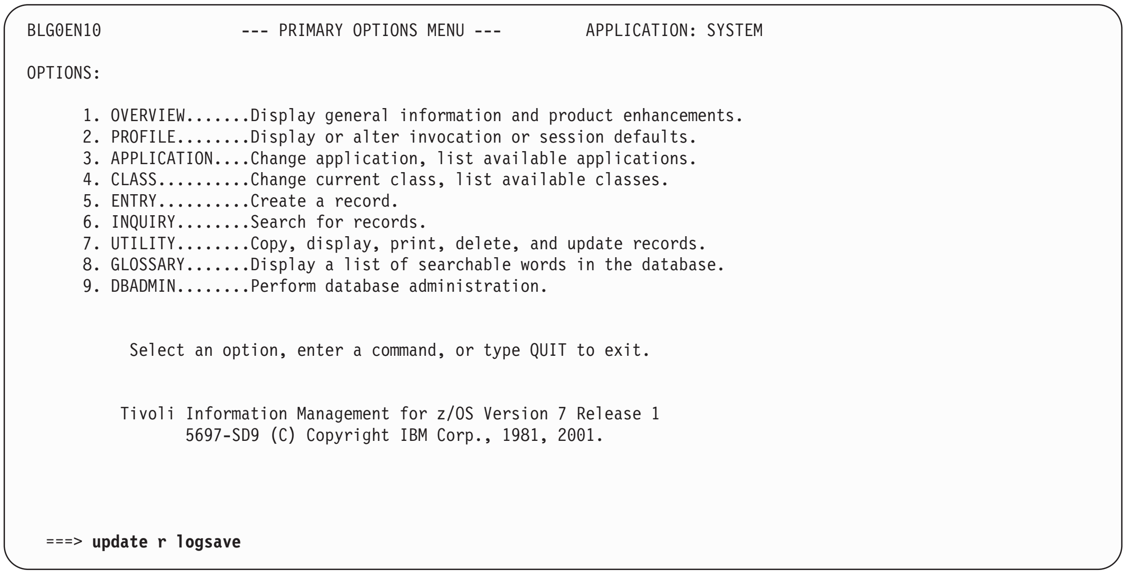
This scenario assumes that you have a LOGSAVE record already defined for the Automatic Log Save Send function (the **Database type** field contains *send*). If you do not, start at "Creating a LOGSAVE Record for a Send Database" on page 259 and create one; then return here.

Note: You cannot propagate Tivoli Information Management for z/OS data from a receive database; you can only propagate Tivoli Information Management for z/OS data from a send database.

To define the LOGSAVE record for DB2 Extract processing, start at the Primary Options Menu and type the following IRC:

```
update r logsave
```

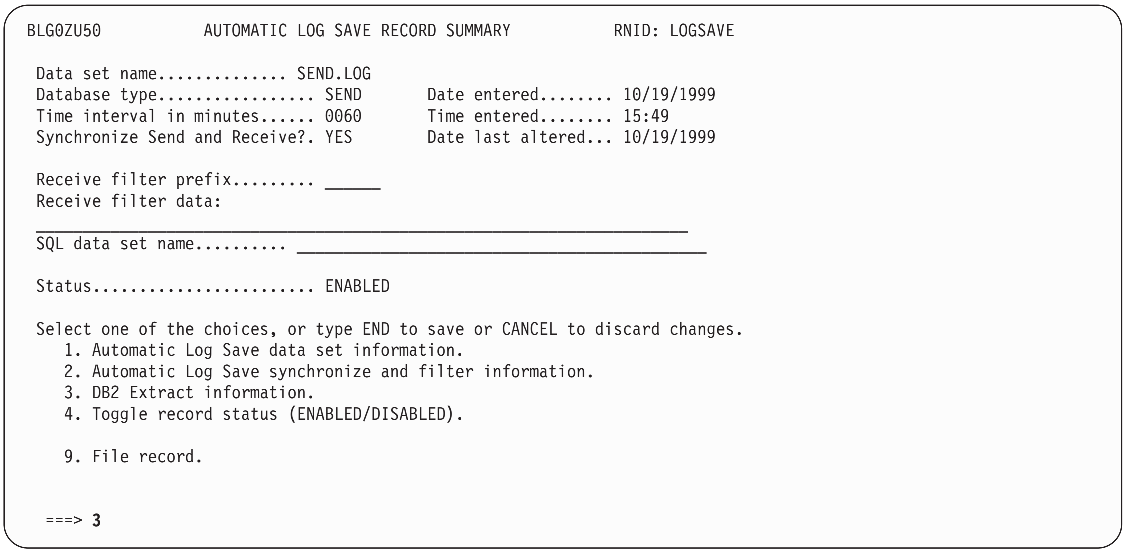
and press **Enter**.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.



  ===> update r logsave
```

On BLG0ZU50, the Automatic Log Save Record Summary panel, select **DB2 Extract information**. and press **Enter**.

Type **3** and press **Enter**.

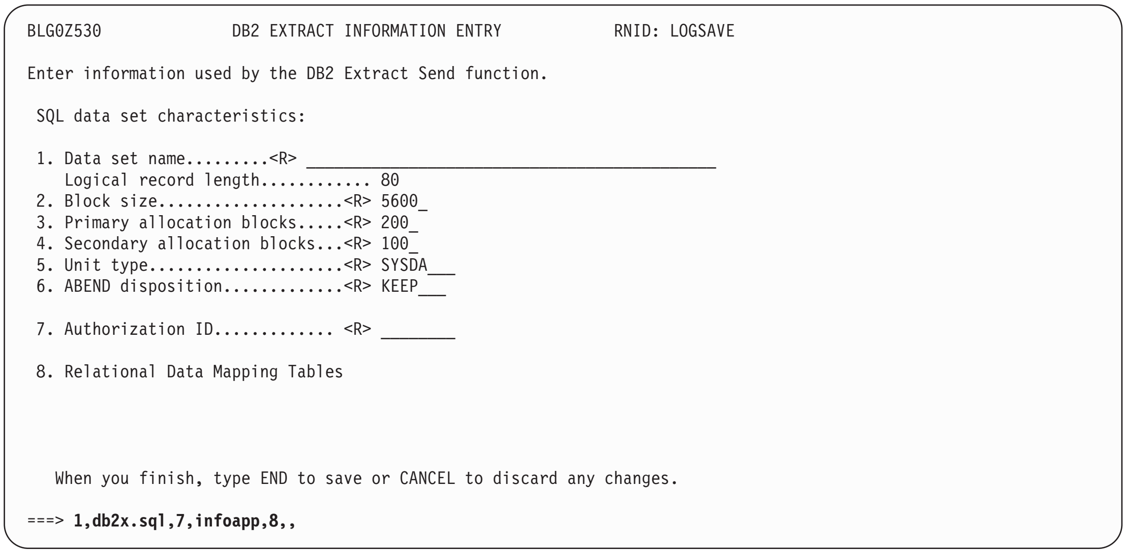


```
BLG0ZU50          AUTOMATIC LOG SAVE RECORD SUMMARY          RNID: LOGSAVE

 Data set name............. SEND.LOG
 Database type................ SEND      Date entered........ 10/19/1999
 Time interval in minutes...... 0060      Time entered........ 15:49
 Synchronize Send and Receive?. YES       Date last altered... 10/19/1999

 Receive filter prefix......... ______
 Receive filter data:
 ______________________________________________________________________
 SQL data set name......... ____________________________________________

 Status........................ ENABLED

 Select one of the choices, or type END to save or CANCEL to discard changes.
    1. Automatic Log Save data set information.
    2. Automatic Log Save synchronize and filter information.
    3. DB2 Extract information.
    4. Toggle record status (ENABLED/DISABLED).

    9. File record.


  ===> 3
```

The DB2 Extract Information Entry panel contains several required fields, some of which are primed. You can accept these values or change them. You must complete the **Data set name** field, the **Authorization ID** field, and select **Relational Data Mapping Tables**. For this example, name the data set DB2X.SQL and specify **infoapp** as the authorization ID. Type this IRC:

```
1,db2x.sql,7,infoapp,8,,
```

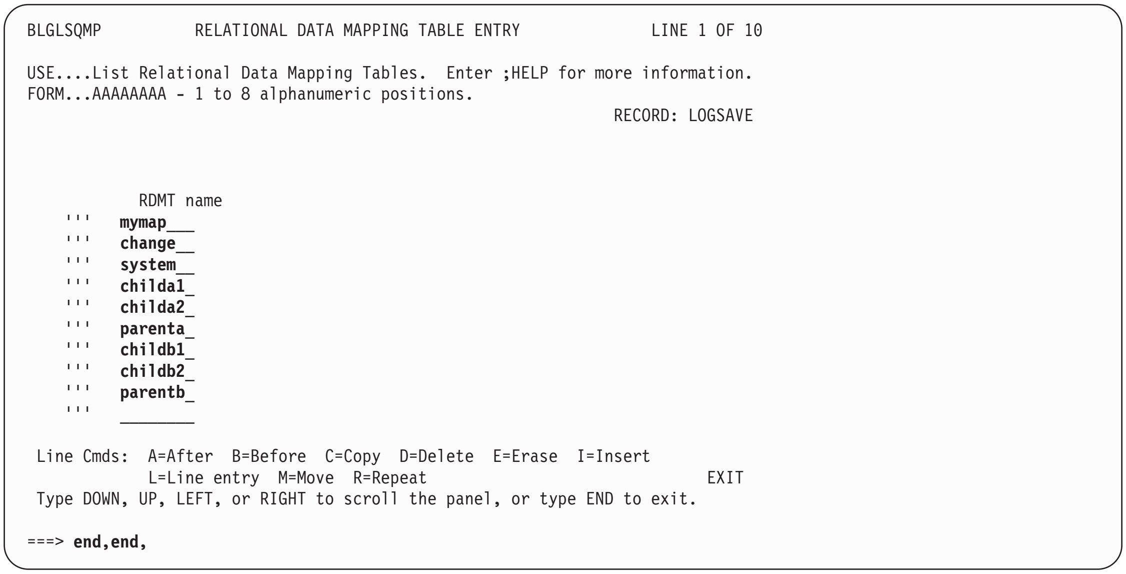
and press **Enter**.



```
BLG0Z530             DB2 EXTRACT INFORMATION ENTRY           RNID: LOGSAVE

Enter information used by the DB2 Extract Send function.

 SQL data set characteristics:

 1. Data set name.........<R> ____________________________________________
    Logical record length........... 80
 2. Block size...................<R> 5600_
 3. Primary allocation blocks.....<R> 200_
 4. Secondary allocation blocks...<R> 100_
 5. Unit type....................<R> SYSDA___
 6. ABEND disposition............<R> KEEP___

 7. Authorization ID............ <R> ________

 8. Relational Data Mapping Tables




   When you finish, type END to save or CANCEL to discard any changes.

===> 1,db2x.sql,7,infoapp,8,,
```

The Relational Data Mapping Table Entry panel appears. Starting at the top and leaving no blanks, list the names of all the RDMTs you want the DB2 Extract Facility to use. (The DB2 Extract Facility stops reading the entries when it detects a blank field.) The RDMTs you list must be in a data set in your BLGRFT concatenation. When you list RDMTs for parent and child records, list child records before their parent records, because a child record typically contains the same s-word as its parent record. If the parent record is listed before the child record, DB2 Extract would find the s-word in the RDMT for the parent before it found the RDMT for the child record and stop looking. Be sure the names you type here match the names you gave to the RDMTs on the BLGUT8 TABLE statements.

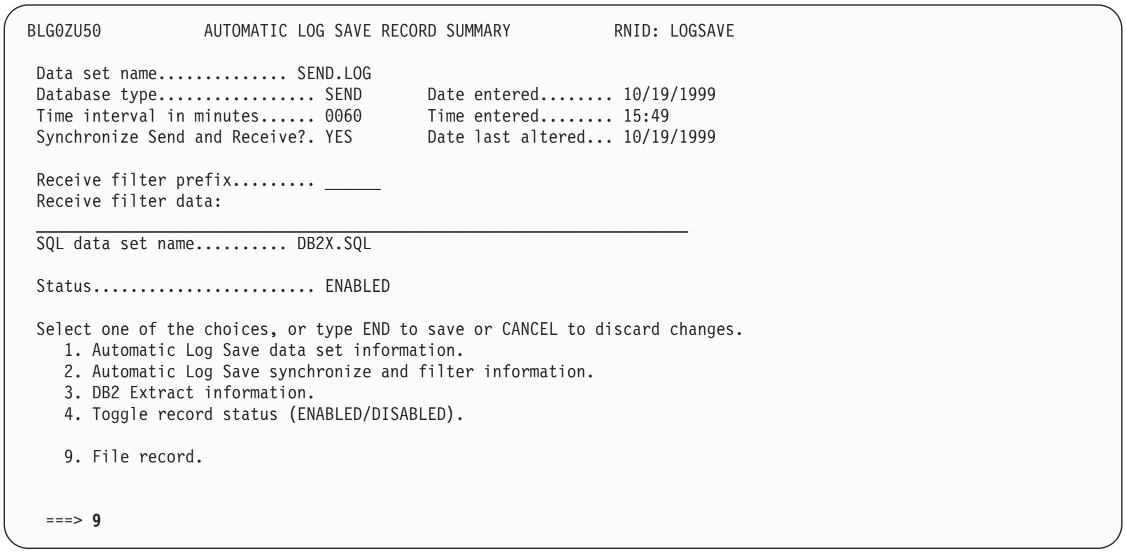
After you list all the RDMTs you want to use, type **end,end,** and press **Enter**.



```
BLGLSQMP         RELATIONAL DATA MAPPING TABLE ENTRY             LINE 1 OF 10

USE....List Relational Data Mapping Tables.  Enter ;HELP for more information.
FORM...AAAAAAAA - 1 to 8 alphanumeric positions.
                                                              RECORD: LOGSAVE



           RDMT name
    '''   mymap___
    '''   change__
    '''   system__
    '''   childa1_
    '''   childa2_
    '''   parenta_
    '''   childb1_
    '''   childb2_
    '''   parentb_
    '''   ________

 Line Cmds:  A=After  B=Before  C=Copy  D=Delete  E=Erase  I=Insert
            L=Line entry  M=Move  R=Repeat                           EXIT
 Type DOWN, UP, LEFT, or RIGHT to scroll the panel, or type END to exit.

===> end,end,
```

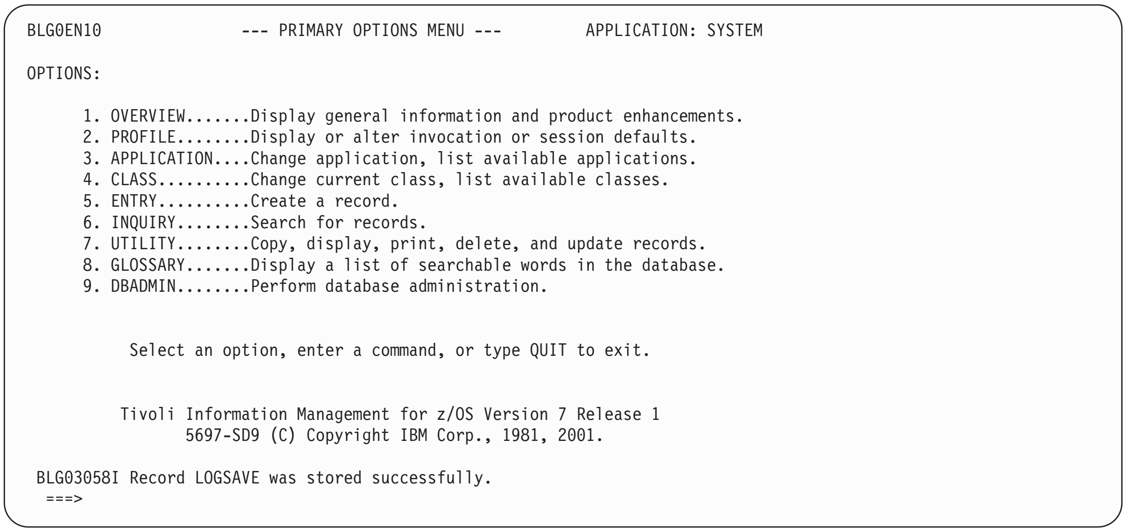
The Automatic Log Save Record Summary panel appears. You have finished updating the LOGSAVE record. To file the record, type **9** and press **Enter**.

DB2 Extract processing does not actually occur, however, until you start the BLGTDBXM TSP.



```
BLG0ZU50          AUTOMATIC LOG SAVE RECORD SUMMARY          RNID: LOGSAVE

 Data set name............. SEND.LOG
 Database type................ SEND      Date entered........ 10/19/1999
 Time interval in minutes...... 0060      Time entered........ 15:49
 Synchronize Send and Receive?. YES       Date last altered... 10/19/1999

 Receive filter prefix......... ______
 Receive filter data:
 ______________________________________________________________________
 SQL data set name.......... DB2X.SQL

 Status....................... ENABLED

 Select one of the choices, or type END to save or CANCEL to discard changes.
    1. Automatic Log Save data set information.
    2. Automatic Log Save synchronize and filter information.
    3. DB2 Extract information.
    4. Toggle record status (ENABLED/DISABLED).

    9. File record.


  ===> 9
```

BLG0EN10, the Primary Options Menu, appears with a message that the LOGSAVE record was stored successfully.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.

 BLG03058I Record LOGSAVE was stored successfully.
  ===>
```

For future reference, you may want to print the LOGSAVE record.

Establishing the DB2 Environment

The DB2 Extract Facility depends on the existence of the tables you specified when you built the RDMTs. Work with your DB2 database administrator to establish the DB2 environment.

DB2 tables exist in a tablespace; tablespaces, in turn, exist in a database. The first step, therefore, in creating a DB2 table, is to create a DB2 database; the second step is to create a tablespace; and the third step is to create a table and any associated indexes. You can modify the following SQL statement to create a database:

CREATE DATABASE *database_name*
STOGROUP *storage_group*
BUFFERPOOL *buffer_pool*

You can modify the following SQL statement to create a tablespace:

CREATE TABLESPACE *tablespace_name*
IN *database_name*
BUFFERPOOL *buffer_pool*;
LOCKSIZE PAGE
CLOSE YES
DSETPASS *data set_password*
PCTFREE 10;

Create the Synchronization Table

To maintain synchronization, the DB2 Extract Utility maintains a sequence number in a DB2 table. If the DB2 Extract Facility is to run with synchronization (the LOGSAVE record has YES in the **Synchronize Send and Receive?** field), you must create this table before you start the facility. See “Maintaining Synchronization” on page 249 for a complete discussion of how synchronization is maintained.

You can modify and submit the following SQL statement to create the synchronization table:

```
CREATE TABLE authorization_id.SYNCH_TABLE
 (SYNCH_NUMBER INT)
 IN database_name.tablespace_name;
```

If you have been running the Automatic Log Save Facility with synchronization, look in the glossary on the send database for the LAST_SDLDS_SENT key to determine the correct sequence number to enter in SYNCH_TABLE. (See “Maintaining Synchronization” on page 249 for more information.) If you have not been using the Automatic Log Save Facility with synchronization, initialize the sequence number to 0. You can use, or modify, this SQL statement to set the sequence number:

```
INSERT INTO authorization_id.SYNCH_TABLE(SYNCH_NUMBER)
 VALUES(00000000);
```

You can use, or modify, this SQL statement to verify that the sequence number is set to the correct value:

```
SELECT * FROM authorization_id.SYNCH_TABLE;
```

Note: SYNCH_TABLE must never contain more than one row. If it has more than one row, DSNTEP2 will fail. Therefore, after you initialize SYNCH_TABLE, you must not INSERT another value into this table. (The DB2 Update Utility uses the UPDATE statement to increment the value.)

Create Tables to Receive Tivoli Information Management for z/OS Data

You can use the output report from BLGUT8 to create the DB2 tables to receive the data from each Tivoli Information Management for z/OS record type you mapped. The table must contain a column for each Tivoli Information Management for z/OS field you mapped to it, as listed in the output report. It must also contain a column named RNID, which is not listed in the report.

If you are mapping freeform text, list, or nonreplaceable data, you must create a separate table for each s-word. This table must contain three columns: RNID, SEQ_NUM, and a third column to receive the text, list processor, or nonreplaceable data.

Use the following information about how Tivoli Information Management for z/OS data is mapped to DB2 to construct your SQL CREATE TABLE statements:

- The RNID of the Tivoli Information Management for z/OS record is expected to be the primary key to all DB2 tables. (You do not have to define the RNID column as the primary key to your DB2 table; however, this column must be unique.) Do not specify the RNID when you create an RDMT with BLGUT8. The user exit BLMSSGEN assumes that RNID is a column in each table and automatically includes it with each SQL statement. Therefore, your DB2 table must contain a column named RNID even though this column is not listed on the RDMT output from BLGUT8. The RNID column must have a length of 8. The RNID can contain mixed character data; therefore, you must decide whether to define this column for SBCS or DBCS data. Consider the information in “DBCS and Mixed Data Considerations” on page 298 when making this decision.
- The sequence number is used as a secondary key for list, freeform text, and nonreplaceable data. A column named SEQ_NUM is mapped to the sequence number. Do not specify SEQ_NUM when you create an RDMT with BLGUT8. The user exit

BLMSSGEN assumes that SEQ_NUM is a character column in each table that is mapped to a record with list, freeform text, and nonreplaceable data and automatically includes it with each SQL statement it creates for list, freeform text, and nonreplaceable data. Therefore, your DB2 table must also contain a column named SEQ_NUM with a length of 8 and a data type of CHAR even though this column is not listed on the RDMT output from BLGUT8. (Mixed data and DBCS data are not possible for this column.) The table must have a compound key of RNID/SEQ_NUM.

The sequence number referred to here is not a VSAM sequence number, but a sequence number generated by Tivoli Information Management for z/OS for repeatable constructs, such as an instance of freeform text or the row number of an instance of list processor data.

- The column length for the column mapped to freeform text must be 250 or the maximum length defined by your Tivoli Information Management for z/OS installation. You must specify a maximum length for the column when you create the DB2 table. See “Example of Propagating Problem Data to DB2” on page 313 for an example.
- If a specific instance of freeform text, list, or nonreplaceable data is less than the column length and the column is defined as data type CHAR, DB2 pads it with blanks.
- If a Tivoli Information Management for z/OS field is defined in an RDMT, but no data is found in the record for that field, the SQL statement generated for that field is blanks with a data length of that defined in the MAXL field of the RDMT. If MAXL is 0, data of 1 blank is used.
- For lists, the DB2 Extract Facility stores data and row occurrences, not every blank field.
- The DB2 Extract Facility is not sensitive to DB2 indexing. You are responsible for ensuring that, if a column is designated to be unique in DB2 by a unique index, the data coming from Tivoli Information Management for z/OS is unique. The DB2 Extract Facility does not check for a value’s existence in a column before putting a record into DB2.
- Create all the DB2 tables with the same authorization ID you specified on the LOGSAVE record, for example INFOAPP.
- You can specify the p-word or the s-word index associated with the Tivoli Information Management for z/OS data item to be mapped to DB2. If you specify a p-word and the panel for the field has an s-word, the s-word is used to identify the data. If the field is replaceable Tivoli Information Management for z/OS data, the first occurrence of the data is extracted from the Tivoli Information Management for z/OS record.
- You cannot map the same s-word/p-word combination to multiple DB2 tables. This means that the same s-word/p-word combination cannot be used as a foreign key in multiple DB2 tables.
- You cannot mix different record types in the same DB2 table. For example, you cannot map some data from the change record and some data from a problem record to the same DB2 table.
- SRC records are not supported. User exit BLMSSGEN skips them.
- For purged records, DELETE statements are generated for each DB2 table identified in the RDMT.

- Each Tivoli Information Management for z/OS record is mapped to one or more DB2 tables. A DELETE statement and an INSERT statement are created for each DB2 table to which data is propagated. So the existing row of data is deleted, then the new one is inserted.
- A COMMIT statement is generated after the statements for each RNID. If the record maps to more than one DB2 table, the COMMIT occurs after all SQL statements for the RNID are complete.
- You can put a Tivoli Information Management for z/OS time field in a DB2 column with a TIME data type.
- Dates in the normal Tivoli Information Management for z/OS format of yy/mm/dd do not go into DB2 if you define the DB2 column with a DATE data type. If you want to use the DATE data type for your DB2 column, you need to use an exit routine that converts the date coming from Tivoli Information Management for z/OS into a valid format for DB2 dates. Refer to the *IBM Database 2 Version 3 Administration Guide* for information on writing and using a date/time exit routine.

Modifying DSNTEP2

On the MVS image where DB2 resides, modify DSNTEP2. DSNTEP2 is a DB2 sample application that is provided with the DB2 product. As DSNTEP2 is shipped, it stops after 10 errors. To enable synchronization, you must modify DSNTEP2 so that it stops processing after only 1 error. Even if you do not synchronize send and receive processing, you may want receive processing to stop after only 1 error. This program may require additional modifications to accommodate your data propagation needs. Refer to the *IBM Database 2 Version 2 Administration Guide* for more information on this program. Consult with your DB2 database administrator before attempting any modifications.

To modify DSNTEP2, do the following steps:

1. Locate your locally installed version of DSNTEP2, which was shipped with the current version of DB2. Modify it as follows:
   a. Copy the code to a source library under your control.
   b. Edit the code and change the INIT value on the DECLARE statement for MAXERROR from 10 to 0.
   c. Edit the comment in the PROGRAM SIZES section of the prologue stating the new MAXERROR value.
   d. Make any other changes to the customizable variables listed in the PROGRAM SIZES section of the prologue according to your local installation preferences.
2. Modify the JCL member BLMSTEP2 in the sample library SBLMSAMP according to the prologue. (Observe the directions in “Directions for Updating JCL” on page 307.) “Preparing the JCL” on page 305 and “Preparing BLMSTEP2” on page 310 contain more information about JCL member BLMSTEP2.
3. Submit the modified JCL to compile, link, and bind a new plan for DSNTEP2.
4. The load module you just created is the one you must specify in the JOBLIB statement of JCL member BLMSRCV for *userid.loadlib*.

   Note: Plans beginning with DSN are usually generated by DB2; so it is recommended that you use a different plan name, perhaps LDSNTEP2 for Local . Specify this name for *plan_name* in JCL members BLMSL1 and BLMSL2.

Creating GDGs

This section pertains to creating the generation data group (GDG) bases for the Automatic Log Save send data set and the SQL data set. Jobs BLMSSND and BLMSDCS do the actual backups to the GDGs you create in this procedure.

When you create the GDG bases, you define how many generations of the GDG to keep by the value you specify on the LIMIT parameter.

To create a GDG base for the Automatic Log Save send data set, modify and submit JCL member BLGDG1 from the sample library, SBLMSAMP. Follow the steps outlined in "Directions for Updating JCL" on page 307 when updating this member in addition to the following steps:

1. Change the job card to your local standards.
2. On the BLGDG card, change *userid.dsname.gdg* to the name you want to use for the generation data group for the send data set. Change **SPACE** and **DCB** attributes to match those of the send data set.
3. On the DEFINE statement, change *userid.dsname.gdg* to the name you want to use for the generation data group for the send data set. This name must match the name you used on the BLGDG card.
4. Change any other options in the DEFINE statement at your discretion. For example, you may want to increase or decrease the value for LIMIT.

The name that you give to the GDG for the Automatic Log Save send data set must match the name that you give to *userid.gdgname1* in BLMSDCS. The DCBDSNTEP2 information must match the DCB information for your Automatic Log Save send data set, which is derived from the DCB information for your SDLDS.

To create a GDG base for the SQL data set, modify and run BLGDG1 as follows:

1. On the BLGDG card, change *userid.dsname.gdg* to the name you want to use for the generation data group for the SQL data set.
2. On the BLGDG card, change the DCB values as follows:

   ```
   DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
   ```

   Note: Use the appropriate BLKSIZE for your installation.
3. On the DEFINE statement, change *userid.dsname.gdg* to the name you want to use for the generation data group for the SQL data set. This name must match the name you used on the BLGDG card.
4. Change any other options in the DEFINE statement at your discretion. For example, you may want to increase or decrease the value for LIMIT.

The name that you give to the GDG for the SQL data set must match the name that you give to *userid.gdgname2* in BLMSSND.

Preparing the JCL

Copy and edit each of the following JCL members from the sample library, SBLMSAMP. Do not submit these JCL members after you update them. Some of these members are used by TSP BLGTDBXM and TSP BLGTDBX1. One of the JCL members, BLMSTEP2, you submit on the MVS image where DB2 resides to compile, link, and bind a new plan for DSNTEP2. See “Directions for Updating JCL” on page 307 for general directions on updating JCL.

BLMSDSD

Runs on the MVS image where the Tivoli Information Management for z/OS send database resides. This job is submitted:

- By you to initiate the DB2 Extract Facility
- By TSP BLDTDBXM if it completes successfully but does not build an SQL data set
- By JCL BLMSRCV if it updates the DB2 database successfully

This JCL does the following tasks:

- Starts TSP BLGTDBXM using Tivoli Information Management for z/OS in batch mode.
- Sends a status message based on the success or failure of TSP BLGTDBXM.

BLMSDCS

Runs on the MVS image where the Tivoli Information Management for z/OS send database resides. It is submitted by TSP BLGTDBXM and does the following tasks:

- Copies the Automatic Log Save send data set to a generation of a GDG.
- Deletes the Automatic Log Save send data set.

This job must complete before the next offload of the SDLDS. User exit BLGUT3EX allocates the send data set as a new data set. If the send data set already exists, BLGUT3EX encounters an allocation error.

Note: The deletion of the SQL data set and the Automatic Log Save send data set must occur before the next iteration of the DB2 Extract Send Utility.

BLMSSND

Runs on the MVS image where the Tivoli Information Management for z/OS send database resides. It does the following tasks:

- Builds the job-stream (BLMSRCV) containing the SQL statements to be processed on the MVS image where DB2 resides. (The SQL statements were built by user exit BLMSSGEN from the data in the Automatic Log Save send data set.)
- Submits job BLMSRCV to the internal reader for routing to the MVS image where DB2 resides.
- Copies the SQL data set to a generation of a GDG and deletes the SQL data set.

BLMSRCV

A partial job that is combined with the in-stream SQL statements from BLMSSND and BLMSL1 to create the job-stream for submission to the internal reader. It does the following tasks:

- Deletes the SQL data set from the previous run.
- Adds the in-stream SQL statements from BLMSSND.
- Starts copying the in-stream SQL statements into a new data set for use by the DSNTEP2 program.

BLMSL1

A partial job that is appended to BLMSRCV after the in-stream SQL statements. It does the following tasks:

- Finishes copying the in-stream SQL statements into a new data set for use by the DSNTEP2 program.
- Runs the PL/I program DSNTEP2 to process the SQL statements in batch.
- Transmits a message to the system and user ID specified. The message that is sent is based on the return code from DSNTEP2 as follows:
  0000 Good message
  0004 Warning message
  >0004 Failure message
- If DSNTEP2 returns a code of 0, submits job BLMSDSD for routing to the MVS image where the send database resides for execution.

BLMSL2

For recovery when errors occur after an SQL data set was successfully generated. This JCL runs on the MVS image where the DB2 database resides. For this job, you specify the input data set instead of using one that is automatically sent from the MVS system where Tivoli Information Management for z/OS resides. It performs the following tasks:

- Runs the PL/I program DSNTEP2 to process the SQL statements in batch.
- Transmits a message to the system and user ID specified. The message that is sent is based on the return code from DSNTEP2 as follows:
  0000 Good message
  0004 Warning message
  >0004 Failure message

BLMSTEP2

For submission on the MVS image where DB2 resides to compile, link, and bind a new plan for DSNTEP2.

Directions for Updating JCL

1. Edit each member. Make your changes using uppercase text.
2. Follow the instructions in the prologue of each member, and printed in this chapter, to find the specific job steps you need to update.
3. Verify the accuracy of each change you make against your LOGSAVE record, TSP, and other JCL members.
4. Be sure to change the job card to your local standards.
5. Save the updated member.

Preparing BLMSDSD

1. Change the job card to your local standards. This job is routed to the MVS image where the Tivoli Information Management for z/OS send database resides; you may need user and password information.
2. Change *node* on the /*ROUTE XEQ card to the node for the MVS image where the job is to run.

3. Change *node.userid* on the /*ROUTE PRINT card to the node and user ID that is to receive the job output.
4. In step SQLGEN, change:
   1. The data set names for the STEPLIB, ISPPROF, ISPPLIB, ISPTLIB, ISPMLIB, ISPSLIB, BLGTRACE, ISPLLIB, SYSPROC, and SYSTSPRT DDnames to correspond to the data set names at your installation. Verify that the report format table concatenation contains the RDMTs needed by the DB2 Extract Send Utility.
   2. The value for SESS(*xx*) to the session-parameters member of your send database.
5. In step GOODGEN, change:
   1. *sample message* to your message for a successful completion (return code = 0) of TSP BLGTDBXM.
   2. *node.userid* to the destination for the message.
6. In step FAILGEN, change:
   1. *sample message* to your message for an unsuccessful completion of TSP BLGTDBXM (return code ≠ 0).
   2. *node.userid* to the destination for the message.

Copy this JCL to a data set on the MVS image where the DB2 database resides.

Preparing BLMSDCS

1. Change the job card to your local standards.
2. In step SAVESEND, change:
   1. All occurrences of *userid.dsname1* to the name of the Automatic Log Save send data set that is in the LOGSAVE record.
   2. *userid.gdgname1* to the name of the GDG for the send data set.
   3. The DCB parameters as necessary to match those of the Automatic Log Save send data set.

Preparing BLMSSND

1. Change the job card to your local standards.
2. In steps BLDJOB, SAVESQL, and SAVEinfo, change *userid.dsname2* to the SQL data set name specified in the LOGSAVE record.
3. On the SYSUT1 DD CARD, you may need to change the names of the two other data sets to the names of your local data sets.
4. In step SAVESQL, change *userid.dsname2* to the SQL data set name specified in the LOGSAVE record. Put this same name in the DSNAME parameter of the DCB statement.
5. In step SAVESQL, change *userid.gdgname2* to the name of the GDG for the SQL data set.

Preparing BLMSRCV

1. Change the job card to your local standards. This job is routed to the MVS image where the DB2 database resides; you may need user and password information.

2. Change *node* on the /*ROUTE XEQ card to the node for the MVS image where the job is to run.
3. Change *node.userid* on the /*ROUTE PRINT to the node and user ID that is to receive the job output.
4. Change *userid.loadlib* to the load library that contains DSNTEP2. If necessary, specify the PL/I system libraries.
5. In step DELETE, change *userid.dsname2* to a name for the SQL data set that will be created on the MVS image where DB2 resides. This name must match the data set name on the SYSUT2 card in BLMSL1.

Preparing BLMSL1

1. In step SYSUT2, change *userid.dsname2* to a name for the SQL data set. If the DB2 database is on the same MVS image as the Tivoli Information Management for z/OS send database, this name must be different from the name you specified in the LOGSAVE record for the SQL data set. If they are the same, data will be overwritten.
2. You may need to change the UNIT and SPACE parameters according to your local standards and the volume of data being processed.
3. In step RUNSQL, change:
   a. *userid.dsname2* to the SQL data set name you specified in the previous step.
   b. *prefix_name* to your prefix, the high-level qualifier for system resource data sets.
   c. *db2_id* to your DB2 subsystem ID.
   d. *plan_name* to your plan name for DSNTEP2.
   e. *userid.loadlib* to the load library that contains the executable load module, DSNTEP2.
4. In step GOODMSG, change:

   a. *this is a sample message...* to your message for a successful completion (return code = 0000) of the RUNSQL step.

   b. *node.userid* to the destination for this message.
5. In step WARNMSG, change:
   a. *this is a sample message...* to your message for a return code of 0004 from the RUNSQL step.
   b. *node.userid* to the destination for this message.
6. In step FAILMSG, change:

   a. *this is a sample message...* to your message for an unsuccessful completion (return code > 0004) of the RUNSQL step.

   b. *node.userid* to the destination for this message.
7. In step SUBDBXG, change *userid.jcl*(BLMSDSD) to the name of the data set that you copied BLMSDSD into on the MVS image where the DB2 database resides.

Preparing BLMSL2

1. In step RUNSQL, change:

   a. *userid.dsname2* to the name of the data set that contains the SQL statements you want to process.

   b. *prefix_name* to your prefix, the high-level qualifier for system resource data sets.

   c. *db2_id* to your DB2 subsystem ID.

   d. *plan_name* to your plan name for DSNTEP2.

   e. *userid.loadlib* to the load library that contains the executable load module, DSNTEP2.

2. In step GOODMSG, change:

   a. *this is a sample message*... to your message for a successful completion (return code = 0000) of the RUNSQL step.

   b. *node.userid* to the destination for this message.

3. In step WARNMSG, change:

   a. *this is a sample message*... to your message for a return code of 0004 from the RUNSQL step.

   b. *node.userid* to the destination for this message.

4. In step FAILMSG, change:

   a. *this is a sample message*... to your message for an unsuccessful completion (return code > 0004) of the RUNSQL step.

   b. *node.userid* to the destination for this message.

Preparing BLMSTEP2

1. Change the job card to your local standards.
2. In step JOBLIB CARD, add the PL/I system libraries, if necessary.
3. In step PPLI.SYSIN, change *userid.dsnsamp* to the data set that contains the DSNTEP2 program.
4. In step PC.DBRMLIB, change *userid.dbrmlib* to the name of your DBRM library.
5. In step PC.SYSLIB, change *userid.srclib* to the name of the data set for the modified source.
6. In step LKED. SYSLMOD, change *userid.runlib.load* to the name of the target load library.
7. In step PH01PS02, change:
   a. *userid.dbrmlib* to the name of your DBRM library.
   b. *plan_name* to your plan name.
   c. *userid.runlib.load* to your target load library.

Modifying the DB2 Extract TSP

Before you modify TSP BLGTDBXM, understand its logic. The logic of TSP BLGTDBXM is as follows:

1. Perform BLMSSGEN setup (passing a **1** in the **Literal/Test data** field). If an error occurs, issue a message and exit the TSP.

2. Set TSCAUFLD field to blanks. TSCAUFLD is used later to determine whether to submit JCL to copy the Automatic Log Save send data set to a generation of a GDG.

3. Call user exit BLGUT4WT to:
   a. Read information from the LOGSAVE record.
   b. Put the information into the TSCA variable data area.
   c. Wait for the time interval specified in the LOGSAVE record.

4. If BLGUT4WT returns an error:
   a. Issue a message.

   b. Perform BLMSSGEN cleanup (passing a **3** in the **Literal/Test data** field).
   c. Exit the TSP.

5. If BLGUT4WT finishes without an error, call user exit BLGUT4EX to offload the SDLDS to the Automatic Log Save send data set.

6. If BLGUT4EX returns an error:
   a. Issue a message.
   b. Perform BLMSSGEN cleanup (passing a **3** in the **Literal/Test data** field).
   c. Exit the TSP.

7. If BLGUT4EX finishes without an error:
   a. Set TSCAUFLD to OFFLDOK
   b. Perform BLMSSGEN extract (passing a **2** in the **Literal/Test data** field).

8. If BLMSSGEN finishes with a TSCAFRET that is not 0 or a TSCAFRES of 12:
   a. Issue a message.
   b. Perform BLMSSGEN cleanup (passing a **3** in the **Literal/Test data** field).
   c. Submit JCL BLMSDCS to copy the Automatic Log Save send data set to a generation of a GDG.
   d. Exit the TSP.

9. If BLMSSGEN finishes with a TSCAFRES of 4 or 8, no SQL data set was built; therefore, the JCL to process the SQL is not submitted.
   a. Submit JCL BLMSDSD to start the next send.
   b. Issue a message indicating the DB2 Extract Send Utility was successful.
   c. Perform BLMSSGEN cleanup (passing a 3 in the **Literal/Test data** field).
   d. Submit JCL BLMSDCS to copy the Automatic Log Save send data set to a generation of a GDG.
   e. Exit the TSP.

10. If BLMSSGEN finishes without an error and SQL was generated, submit JCL BLMSSND to process the SQL data set.

    The JCL builds a job stream to load the SQL data into DB2 and routes it to the MVS image where the DB2 database resides. It also copies the SQL data set to a member of a GDG and deletes the data set. The SQL data set must be deleted before the next time interval elapses because BLMSSGEN attempts to create the data set; therefore, the data set must not exist.

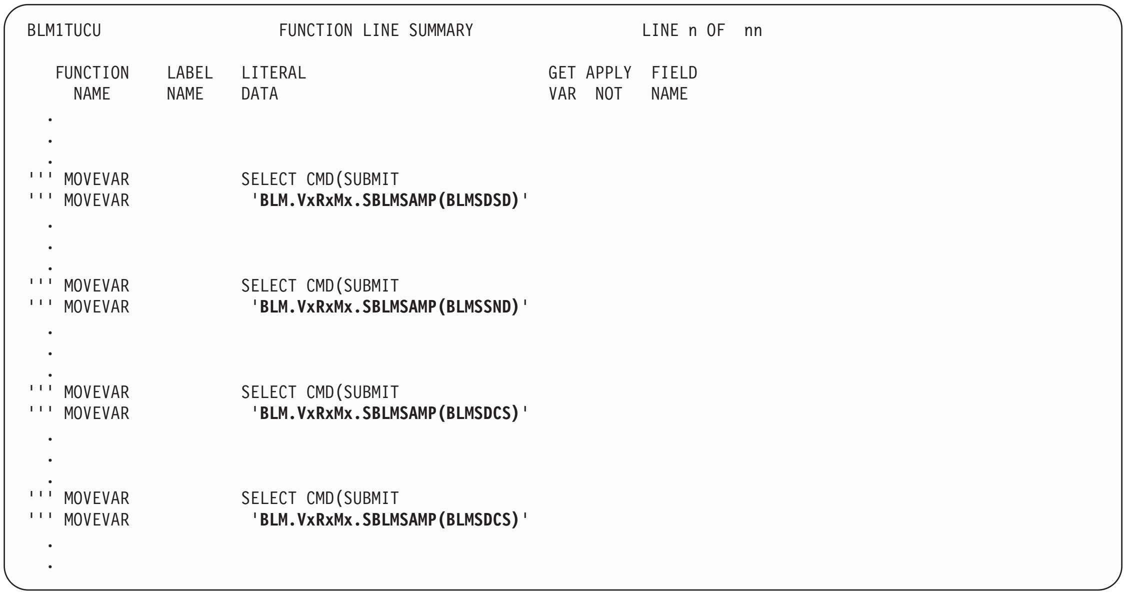
11. If there is an error submitting the JCL:
    a. Issue a message.
    b. Perform BLMSSGEN cleanup (passing a **3** in the **Literal/Test data** field).
    c. Submit JCL BLMSDCS to copy the Automatic Log Save send data set to a generation of a GDG.
    d. Exit the TSP.

12. If there is no error submitting the JCL:
    a. Issue a message indicating the DB2 Extract Send Utility was successful.
    b. Perform BLMSSGEN cleanup (passing a 3 in the **Literal/Test data** field).
    c. Submit JCL BLMSDCS to copy the Automatic Log Save send data set to a generation of a GDG.
    d. Exit the TSP.

Using PMF, update the following statements in the BLGTDBXM TSP as follows:

1. Change BLM.VxRxMx.SBLMSAMP(BLMSDSD) to the name of your library that contains the JCL you modified in “Preparing BLMSDSD” on page 307. (This library is on the MVS image where the send database resides.)
2. Change BLM.VxRxMx.SBLMSAMP(BLMSSND) to the name of your library that contains the JCL you modified in “Preparing BLMSSND” on page 308. (This library is on the MVS image where the send database resides.)
3. Change BLM.VxRxMx.SBLMSAMP(BLMSDCS) to the name of your library that contains the JCL you modified in “Preparing BLMSDCS” on page 308. (This library is on the MVS image where the send database resides.)

This panel shows only the lines in the BLGTDBXM TSP that you change to propagate data to DB2.



```
BLM1TUCU                FUNCTION LINE SUMMARY                 LINE n OF  nn

   FUNCTION    LABEL   LITERAL                          GET APPLY  FIELD
     NAME      NAME    DATA                             VAR  NOT   NAME
  .
  .
  .
''' MOVEVAR            SELECT CMD(SUBMIT
''' MOVEVAR             'BLM.VxRxMx.SBLMSAMP(BLMSDSD)'
  .
  .
  .
''' MOVEVAR            SELECT CMD(SUBMIT
''' MOVEVAR             'BLM.VxRxMx.SBLMSAMP(BLMSSND)'
  .
  .
  .
''' MOVEVAR            SELECT CMD(SUBMIT
''' MOVEVAR             'BLM.VxRxMx.SBLMSAMP(BLMSDCS)'
  .
  .
  .
''' MOVEVAR            SELECT CMD(SUBMIT
''' MOVEVAR             'BLM.VxRxMx.SBLMSAMP(BLMSDCS)'
  .
  .
```

If you want to change the maximum number of records to offload each time an enqueue is obtained on the SDLDS, you can change the **RECS=** and **%FULL=** parameters in the control line that calls user exit BLGUT4EX. As shipped, the BLGTDBXM TSP has RECS=20. See “Modifying the BLGDUMP1 TSP” on page 268 for an example of how to change the control line.

File the updated TSP. After you file the updated TSP, you must copy it from the write panel data set to a read panel data set so it is available for use. After you do that, you are ready to start the batch job that runs the TSP if you have completed all the other setup steps.

Starting Send and Update Processing

To start normal Send and Update processing, start the DB2 Extract Facility by submitting JCL BLMSDSD on the MVS image where the send database resides. Before you submit JCL BLMSDSD, verify that:

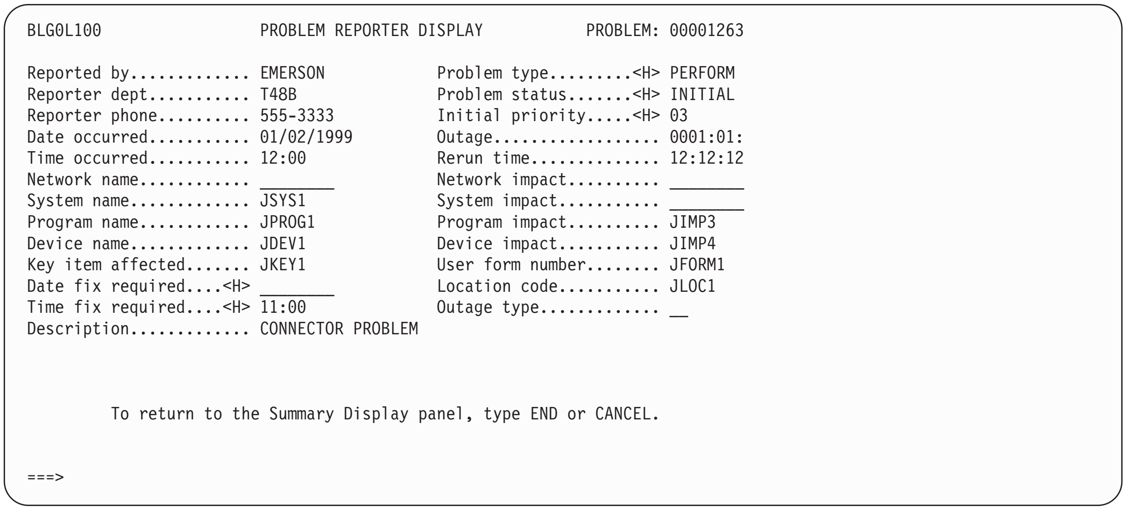
- The set up for your send database is complete.
- The set up for the DB2 database is complete.
- The status of the send LOGSAVE record is enabled.

Example of Propagating Problem Data to DB2

This section illustrates Tivoli Information Management for z/OS data propagated to DB2.

This is one panel of a Tivoli Information Management for z/OS problem record. The example on page 297 maps some of the fields on this panel to DB2 tables PROBLEM_DATA and PROBLEM_TEXT.

The DB2 table in Figure 17 shows data from this record propagated to DB2 table PROBLEM_DATA. It shows two sample records with 8 fields. The ninth field is a freeform text field, so it is mapped to another table, illustrated in Figure 18 on page 314.



```
BLG0L100               PROBLEM REPORTER DISPLAY          PROBLEM: 00001263

Reported by............ EMERSON           Problem type.........<H> PERFORM
Reporter dept.......... T48B              Problem status.......<H> INITIAL
Reporter phone......... 555-3333          Initial priority.....<H> 03
Date occurred.......... 01/02/1999        Outage.................. 0001:01:
Time occurred.......... 12:00             Rerun time.............. 12:12:12
Network name........... ________          Network impact.......... ________
System name............ JSYS1             System impact........... ________
Program name........... JPROG1            Program impact.......... JIMP3
Device name............ JDEV1             Device impact........... JIMP4
Key item affected...... JKEY1             User form number........ JFORM1
Date fix required....<H> ________         Location code........... JLOC1
Time fix required....<H> 11:00            Outage type............. __
Description............ CONNECTOR PROBLEM



        To return to the Summary Display panel, type END or CANCEL.


===>
```

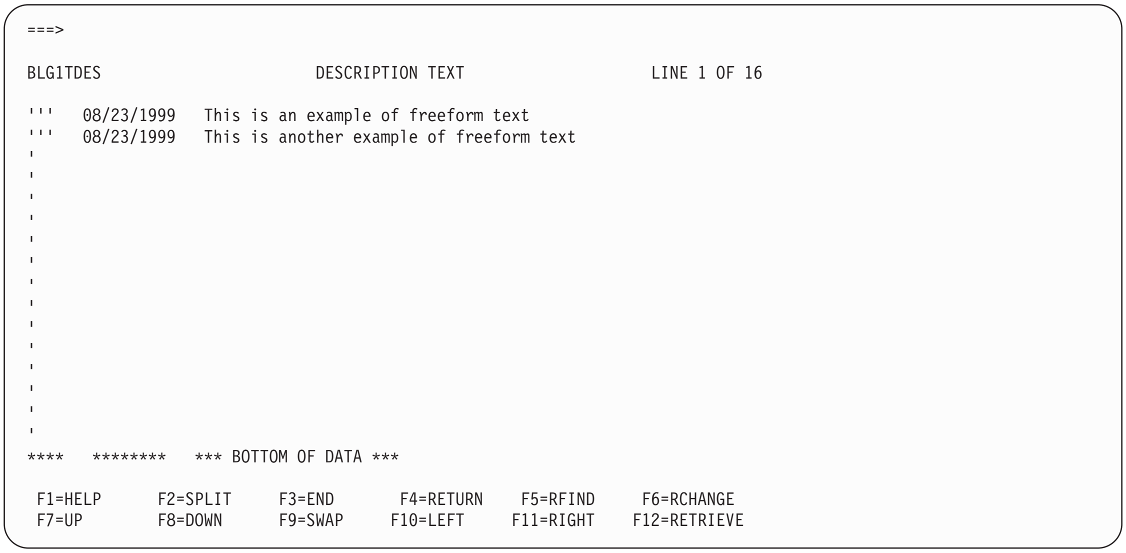
```
RNID     REPORTER_NAME   PROBLEM_TYPE   REPORTER_DEPT  REPORTER_STAT REPORTER_PHONE   INITIAL_PRIORITY   DESCRIPTION
-------- --------------- --------       ------------   -------       -----------       --                -----------
00001263 EMERSON         PERFORM        T48B           INITIAL        555-3333         03                CONNECTOR PROBLEM
00000090 THOREAU         HARDWARE       T44A           OPEN          7579              02                SEND
```

Figure 17. Propagated Data Retrieved from DB2

You might have used these SQL statements to create the DB2 table and index:

```
CREATE TABLE authorization_id.PROBLEM_DATA
 (RNID                       CHAR(08) NOT NULL         ,
   REPORTER_NAME             CHAR(15)                  ,
   PROBLEM_TYPE              CHAR(08)                  ,
   REPORTER_DEPT             CHAR(11)                  ,
   REPORTER_STAT             CHAR(07)                  ,
   REPORTER_PHONE            CHAR(13)                  ,
   INITIAL_PRIORITY          CHAR(02)                  ,
   DESCRIPTION               CHAR(45)                  ,
   PRIMARY KEY (RNID) )
  IN database_name.tablespace_name;
 CREATE UNIQUE INDEX authorization_id.index_name
  ON authorization_id.PROBLEM_DATA (RNID);
```

This example shows freeform text in Tivoli Information Management for z/OS. The RDMT in the example on page 297 maps it to DB2 table PROBLEM_TEXT. Figure 18 on page 314 shows how this data could be propagated to DB2.



```
===>

BLG1TDES                      DESCRIPTION TEXT                  LINE 1 OF 16

'''   08/23/1999   This is an example of freeform text
'''   08/23/1999   This is another example of freeform text
'
'
'
'
'
'
'
'
'
'
'
'
'
'
****   ********   *** BOTTOM OF DATA ***

 F1=HELP      F2=SPLIT     F3=END       F4=RETURN    F5=RFIND     F6=RCHANGE
 F7=UP        F8=DOWN      F9=SWAP     F10=LEFT     F11=RIGHT    F12=RETRIEVE
```

```
RNID     SEQ_NUM  TEXT
-------- -------- -----------------------------------
00001263 00000001 This is an example of freeform text
00001263 00000002 This is another example of freeform text
```

Figure 18. Propagated Freeform Text Retrieved from DB2

You might have used these SQL statements to create the DB2 table and index:

```
CREATE TABLE authorization_id.PROBLEM_TEXT
 (RNID       CHAR(08) NOT NULL        ,
  SEQ_NUM    CHAR(08)                 ,
  TEXT       CHAR(250)                ,
 PRIMARY KEY (RNID,SEQ_NUM)              )
 IN database_name.tablespace_name;
CREATE UNIQUE INDEX authorization_id.index_name
 ON authorization_id.PROBLEM_TEXT (RNID,SEQ_NUM);
```

Stopping the DB2 Extract Facility

To stop the DB2 Extract Facility, toggle the status of the LOGSAVE record from *enabled* to *disabled*. You can use the following IRC from the Primary Options Menu to toggle the status:

```
update r logsave,4,9
```

When you stop the send utility, the update utility also stops. If a send is currently in progress (that is BLMSDSD has been started and TSP BLGTDBXM has called user exit BLGUT4WT and the exit is waiting), the current send finishes waiting. Then the current send offloads the SDLDS, builds the SQL data set, and routes BLMSRCV for execution. The update utility (BLMSRCV) executes and submits BLMSDSD on the MVS image where the send database resides. BLMSDSD fails in step OFFLD because of a TSP error indicating that the status of the LOGSAVE record is disabled.

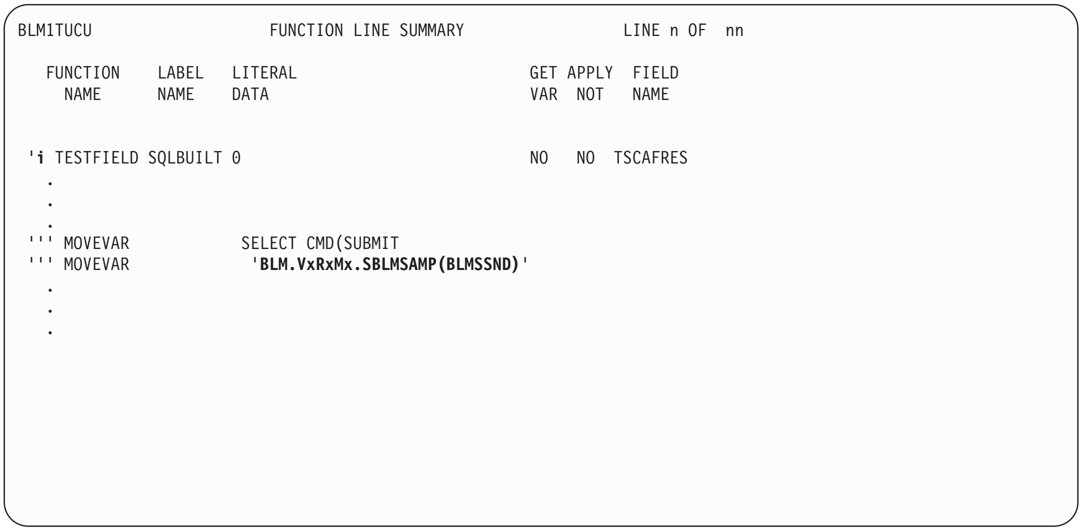
Disabling Automatically Starting the Next Send

If you want to disable automatically starting the next send after a successful update, delete the step that submits the JCL to start the next send from the JCL for the DB2 Update Utility. You also need to modify the TSP so that it does not submit the JCL to start a send if no SQL is built.

To implement the DB2 Extract Facility so that it does just one send and one update, then stops, modify the TSP and JCL as follows:

1. Using PMF, copy and update TSP BLGTDBXM as follows:
   a. Insert a control line to branch to label ISSUGMSG after the control line that tests whether an SQL data set was built. This disables the TSP control lines that submit JCL to perform the next send if no SQL is built.
   b. Change the line of the TSP that submits JCL BLMSSND to submit the new version of BLMSSND that you copy or update according to step 2c.

   This panel shows where you need to insert the control line to branch to label ISSUGMSG and change the line that submits the JCL.



```
BLM1TUCU                  FUNCTION LINE SUMMARY                LINE n OF  nn

   FUNCTION    LABEL   LITERAL                        GET APPLY  FIELD
     NAME      NAME    DATA                           VAR  NOT   NAME


 'i TESTFIELD SQLBUILT 0                              NO   NO  TSCAFRES
   .
   .
   .
 ''' MOVEVAR            SELECT CMD(SUBMIT
 ''' MOVEVAR             'BLM.VxRxMx.SBLMSAMP(BLMSSND)'
   .
   .
   .
```

2. Copy and update the following JCL members:
   a. BLMSDSD. In step SQLGEN, change TSP BLGTDBXM to the name you used for the TSP you modified according to step 1.
   b. BLMSL1. Delete step SUBDBXG.
   c. BLMSSND. In step BLDJOB, change BLMSL1 to the name of the copy of BLMSL1 you modified in step 2b.

Each time you submit the modified BLMSDSD, one send and one update utility is performed. JCL to start another DB2 Send Utility is *not* submitted.

Propagating Data to Both Tivoli Information Management for z/OS and DB2

If you want to propagate data to both Tivoli Information Management for z/OS and DB2, set up the Automatic Log Save Facility as described in “Setting Up the Automatic Log Save Send Function” on page 257 for data propagation to both Tivoli Information Management for z/OS and DB2. You also need to modify TSP BLGTDBXM and the JCL for both the Automatic Log Save Facility and the DB2 Extract Facility as described in this section.

The changes alter the JCL for the Automatic Log Save Facility to use the DB2 Extract TSP so that both a send data set and an SQL data set are created. Because the JCL for the Automatic Log Save Receive Function submits JCL to start the next send after a successful receive, modify the JCL for the DB2 Update Utility so that it does not submit JCL to start a send after a successful update.

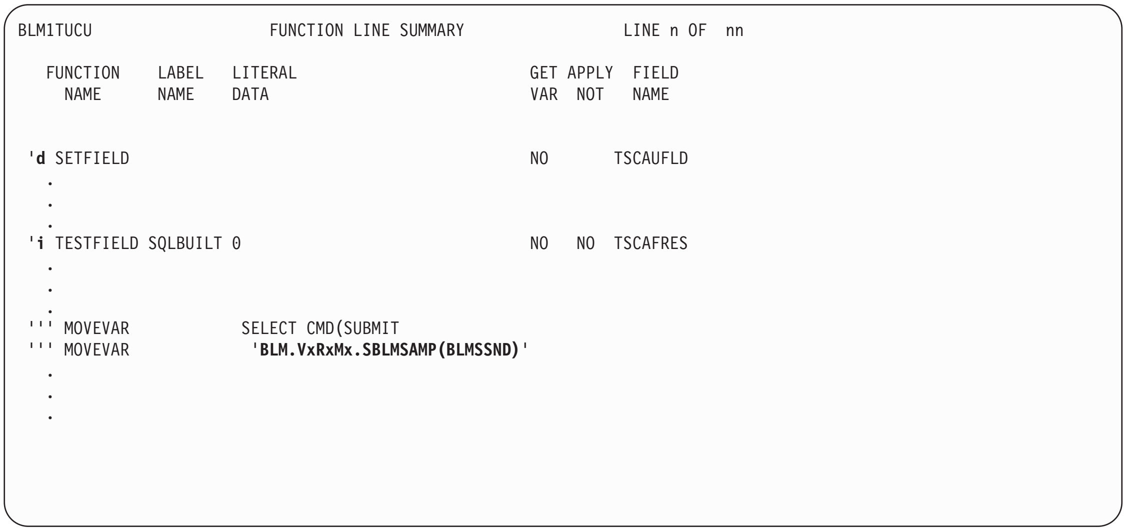
When you use the Automatic Log Save and DB2 Extract Facilities with these changes, the Automatic Log Save Facility drives the process, and an Automatic Log Save Receive failure causes both facilities to stop. After the send and SQL data sets are created, JCL is submitted by TSP BLGTDBXM to process the SQL data set and start the DB2 Update Utility. If the DB2 Update Utility fails, the failure has no effect on Automatic Log Save Send and DB2 Extract Send processing.

Changing the TSP

Change TSP BLGTDBXM as described in “Modifying the DB2 Extract TSP” on page 310. In addition, make the following changes:

1. Delete the control line that does a SETFIELD of TSCAUFLD to OFFLDOK. This disables the submission of the JCL to copy the send data set to a GDG because BLMSASD or BLMSASDA will do the copying.
2. Insert a control line to branch to label ISSUGMSG after the control line shown in this panel. This disables the TSP control lines that submit JCL to perform the next send if no SQL is built.
3. Change the line that submits JCL BLMSSND to submit the new version of BLMSSND that you copy and modify according to step 2 under “Changing the JCL for the DB2 Extract Facility” on page 317.

This panel shows the line you need to delete. This panel also shows where you need to insert the control line to branch to label ISSUGMSG and change the line that submits the JCL.



```
BLM1TUCU                   FUNCTION LINE SUMMARY                 LINE n OF  nn

   FUNCTION    LABEL   LITERAL                         GET APPLY  FIELD
     NAME      NAME    DATA                            VAR  NOT   NAME


 'd SETFIELD                                           NO       TSCAUFLD
   .
   .
   .
 'i TESTFIELD SQLBUILT 0                               NO   NO  TSCAFRES
   .
   .
   .
 ''' MOVEVAR            SELECT CMD(SUBMIT
 ''' MOVEVAR             'BLM.VxRxMx.SBLMSAMP(BLMSSND)'
   .
   .
   .
```

Changing the JCL for the Automatic Log Save Facility

To enable data propagation to both a Tivoli Information Management for z/OS receive database and a DB2 database, make the following additional modifications to the JCL members for the Automatic Log Save Facility:

1. BLMSAL1. In step SUBSND, refer to the version of JCL BLMSASDA that you modified according to step 2 in this section and that is on the MVS image where the receive database resides.
2. BLMSASD and BLMSASDA
   a. In step OFFLD, change TSP(BLGDUMP1) to refer to the version of TSP BLGTDBXM you copied and modified according to "Changing the TSP" on page 316. Modify data set names as required; for example, verify that the RFT data set concatenation contains the RDMTs required for the DB2 Extract Send Utility.
   b. In step GOODOFF, change the GOODOFF DD contents if necessary. This message is sent for a successful completion (return code = 0000) from step OFFLD.
   c. In step FAILOFF, change the FAILOFF DD contents if necessary. This message is sent for an unsuccessful completion (return code ≠ 0000) from step OFFLD.
   d. In step SUBSND, change the data set name on IEFRDER to refer to the copied and modified JCL BLMSASDA.
   e. In step COPY2, change the data set name to refer to the version of JCL BLMSAL1 you modified in step 1 in this section.

Changing the JCL for the DB2 Extract Facility

Make the following additional modifications to the JCL members for the DB2 Extract Facility:

1. BLMSL1. Delete step SUBDBXG so that the JCL for the next send is not submitted.
2. BLMSSND. In step BLDJOB, change BLMSL1 to refer to the version of BLMSL1 you modified in step 1 of this section.

Starting Propagation to Both

To start data propagation to both a Tivoli Information Management for z/OS receive database and a DB2 database, on the MVS image where the send database resides, submit the version of BLMSASD you modified in step 2 on page 317 under “Changing the JCL for the DB2 Extract Facility” on page 317.

DSNTEP2 Messages

DSNTEP2 writes messages to SYSPRINT. When the RUNSQL step of job BLMSRCV finishes, the job output, which includes the messages from DSNTEP2 if SYSPRINT is not allocated to a data set, is sent to the user ID specified on the ROUTE PRINT card in the JCL.

Refer to the prologue of the DSNTEP2 program for a list of the messages it returns.

Recovering from Errors

The DB2 Extract Facility provides the following tools to assist you in recovering from errors:

- TSP BLGTDBX1 parses an Automatic Log Save send data set into SQL when the send data set was left unparsed because the TSP BLGTDBXM stopped. If the send data set was successfully copied to a GDG and the send data set was deleted, you must manually create the send data set and copy the most recent generation of the GDG into the send data set before running TSP BLGTDBX1.
- The Automatic Log Save send data set and the SQL data set are copied to GDGs on the MVS image where the Tivoli Information Management for z/OS send database resides.
- A synchronization table in DB2 and a synchronization counter in the Tivoli Information Management for z/OS SDIDS are provided to store the sequence number. The first SQL statement in the SQL data set checks the sequence number and issues a 0 divide error if the number is not the one that was expected. DB2 Update Utility returns an error message and does not submit the JCL to start the next DB2 Send Utility. If the sequence number in the DB2 SYNCH_TABLE is the one expected, the rest of the SQL statements are processed. The last SQL statement in the SQL data set updates the SYNCH_TABLE sequence number. Then the DB2 Update Utility submits the JCL to start the next DB2 Send Utility. See “Maintaining Synchronization” on page 249 for more information about the synchronization counters and how to determine their values.
- BLMSL2 is a batch job that processes the SQL statements in the SQL data set. When the DB2 Update Utility stops, the MVS image where the Tivoli Information Management for z/OS send database resides may have one or more SQL data sets saved in GDGs. Under normal processing, there is only one unprocessed SQL data set. BLMSL2 is a batch interface to DSNTEP2. It runs on the MVS image where DB2 resides to allow these data sets to be processed into DB2. After the data sets have been processed, you can restart the DB2 Extract Send Utility.

Recovering from a BLGTDBXM Failure

When TSP BLGTDBXM fails, follow this procedure to recover:

1. Refer to the messages written to SYSPRINT on the MVS image where Tivoli Information Management for z/OS resides and to the output from BLMRSCV to diagnose problem:

a. Examine the return and reason codes from the user exit that failed.

b. Look up the return and reason codes in the table for the user exit:
   - For user exit BLGUT4EX, see Table 39 on page 327.
   - For user exit BLGUT4WT, see Table 40 on page 328.
   - For user exit BLMSSGEN, see Table 41 on page 330.

   The description column explains the source of the error and indicates what to do to correct the problem.

c. Examine the job output. Consider these questions:
   - Was the job submission to the internal reader on the send database successful? Check step SUBJOB in BLMSSND.
   - Was the job routed to the correct MVS image where your DB2 database resides? Check the /*ROUTE XEQ card in BLMSRCV.
   - Are the attributes of the SQL data set correct in the LOGSAVE record on the send database?

2. If the SQL data set was created but not processed:

   a. Manually copy it to a GDG.

   b. Transmit the SQL data set to the MVS image where the DB2 database resides and receive the data set.

   c. Prepare and submit BLMSL2 as described in "Preparing BLMSL2" on page 309 to process the SQL data set to update DB2 in batch.

3. If the SQL data set was not created:

   a. Determine why it was not created and fix the problem.

   b. Change the **Time interval in minutes** field in the LOGSAVE record to 0.

   c. If the send data set was copied to a GDG and the send data set was deleted, create the send data set and copy the most recent generation of the send GDG into it.

   d. Run TSP BLGTDBX1 to reprocess the Automatic Log Save send data set.

   e. After BLGTDBXM1 completes successfully, change the **Time interval in minutes** field in the LOGSAVE record back to its previous value.

   f. Delete the send data set if you had to create it.

4. Delete the SQL data set and restart the DB2 Extract Facility. See "Starting Send and Update Processing" on page 312.

Recovering When There Is a Receive Error

If the SQL data set was generated, but the receiving system encountered an error, or the receive JCL was not submitted, do the following tasks:

1. Determine the cause of the problem. Consider the questions in step 1c in "Recovering from a BLGTDBXM Failure" on page 318 plus the following questions:
   - Do you have the DB2 tables built with the same authorization ID that is specified in the LOGSAVE record on the send database?
   - Is your DB2 environment active?

2. If the problem occurred with DB2, for example the system was down, you can resubmit the SQL data sets with BLMSL2 after the problem has been fixed.
3. If the error was caused by an incorrect RDMT:
   a. Use BLGUT8 to rebuild the RDMT.
   b. Use TSP BLGTDBX1 to regenerate the SQL data set. Copy the generation of the send data set to be processed into the send data set and run BLGTDBX1 to regenerate the SQL data set.
   c. Prepare and submit BLMSL2 as described in “Preparing BLMSL2” on page 309 to process the SQL data set to update DB2 in batch.
4. Restart the DB2 Extract Facility. See “Starting Send and Update Processing” on page 312.

Recovering When the SDLDS Is Full

When the SDLDS is full, you cannot update the LOGSAVE record. If the information in the LOGSAVE record (for example, the **Time interval in minutes** is too long, or the **Status** is *disabled*) does not allow for immediate offload using TSP BLGTDBXM, you can use the following steps to manually start DB2 Extract Send and Update processing:

1. Manually allocate the send data set using the name and data set characteristics specified in the LOGSAVE record.
2. Use BLGUT4 to offload the SDLDS to the send data set allocated in step 1.
3. Run TSP BLGTDBX1 to generate the SQL and start DB2 Update processing.
4. Submit JCL BLMSDCS to copy the send data set to a GDG and delete the send data set.

If you are also using the Automatic Log Save Facility to propagate data to a receive database, use BLGUT3 to load the records from the send data set into the receive database. Refer to the *Tivoli Information Management for z/OS Operation and Maintenance Reference* for information on using BLGUT3.

Resetting Synchronization

If you encounter a synchronization error from the DB2 Update Utility, one or more SQL data sets have not been processed. This could happen if:

- SQL is successfully generated, but it is not used to update DB2 because of an error.
- You start the DB2 Extract Send Utility again without first successfully processing the first SQL data set.
- The synchronization error occurs and you have two SQL data set to process.

Reset Synchronization Scenario 1

Synchronization is lost. Copies of the data sets that have not been received are available.

1. Determine what caused the loss of synchronization.
2. If the problem occurred with DB2, for example the system was down, you can resubmit the SQL data sets with BLMSL2 after the problem is fixed. See “Preparing BLMSL2” on page 309 for more information about BLMSL2.

Reset Synchronization Scenario 2

The scenario in “Reset Synchronization Scenario 1” on page 320 depends on recovering the Automatic Log Save send data sets in order by the sequence number so that you can resubmit them to DB2. If the data sets are not available:

1. Rebuild the DB2 database using the procedure currently in place at your data center for loading Tivoli Information Management for z/OS data into DB2.
2. Determine the sequence number stored in the SDIDS of the Tivoli Information Management for z/OS database. See “Maintaining Synchronization” on page 249 for information on how to do this.
3. Reset the synchronization count in DB2 table SYNCH_TABLE to equal the one in the Tivoli Information Management for z/OS send database’s SDIDS. You can modify and use the following SQL statement to do this:

   ```
   UPDATE authorization_id.SYNCH_TABLE
      SET SYNCH_NUM = new_value
      WHERE SYNCH_NUM = current_value;
   ```

   Do not use the INSERT statement because it would add another row to the table. This table must have only one row.

16

Data Propagation User Exits

The user exits described in this section are intended for use by the Automatic Log Save Facility and DB2 Extract Facility. After you understand these exits and how they work together, you can use them for your own purposes.

User exits BLGUT3EX and BLGUT4EX receive their input from BLGUT3WT and BLGUT4WT respectively. User exits BLGUT3WT and BLGUT4WT receive their input from the LOGSAVE record in the Tivoli Information Management for z/OS database. User exit BLMSSGEN receives its input from the LOGSAVE record, relational data mapping tables (RDMTs), and BLGUT4EX. The data is passed through the TSCA variable data area. Refer to the *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference* for information on the TSCA variable data area.

Note: User exits which require that information be specified on the TSP USEREXIT control line cannot be run from a TSX. Refer to the Environment section of each of the user exits in this chapter to determine whether or not the user exit can be run from a TSX.

User Exits

BLGUT3EX–Online Recovery Utility

Loads the Tivoli Information Management for z/OS database with records that were offloaded by the BLGUT4EX user exit. The BLGLOAD1 TSP contains an example of using this user exit.

Input TSCA variable data area output by BLGUT3WT.

Send data set that was output by BLGUT4EX and received as the receive data set.

Output

Records from the receive data set converted to SDDS records.

Environment

TSP and TSX.

A return code and reason code as listed in Table 37. The user exit writes informational and error messages to SYSPRINT. If you allocate SYSPRINT to a data set, the record format must be Variable Blocked ANSI (VBA).

Table 37. BLGUT3EX Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.

Table 37. BLGUT3EX Return and Reason Codes (continued)

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
4	2	Error allocating the receive data set. Check the data set name. Make sure the data set was received. Send it again.
4	11	Logical record skipped because the data is not valid. Message BLG03906 was written. Update the record on the send database so that it will be sent again.
4	12	Freeform text from one or more records cannot be processed. Message BLG21305 was written. Update the record on the send database so that it will be sent again.
4	17	Records are not in sequential order.
8	1	Internal control blocks were not found.
8	3	Error opening the receive data set. Look for other messages.
8	4	Error reading records from the receive data set. Look for other messages.
8	9	Error opening the SDIDS. Look for other messages.
8	10	Error closing the SDIDS. Look for other messages.
8	13	Error updating the SDIDS. Look for other messages.
8	14	Error reading the SDIDS. Look for other messages.
8	15	Error closing the receive data set. Look for other messages.
8	16	Error freeing the receive data set. Look for other messages.
8	18	Error cognizing the records added to the SDDS.
8	20	The sequence number in the send data set does not match the sequence number in the receive database's SDIDS. Synchronization is lost. See “Restoring Synchronization” on page 280 for information on how to restore synchronization.

BLGUT3WT–Initialize for Receive

Initializes the TSCA variable data area and waits for the interval specified in the LOGSAVE record. The BLGLOAD1 TSP contains an example of using this user exit.

Input LOGSAVE record for a receive database.

Output
TSCA variable data area.

Environment
TSP and TSX.

A return code and reason code as listed in Table 38 on page 325. The user exit writes informational and error messages to SYSPRINT. If you allocate SYSPRINT to a data set, the record format must be Variable Blocked ANSI (VBA).

Table 38. BLGUT3WT Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
4	4	A database error occurred reading the LOGSAVE record.
4	8	The LOGSAVE record was not found. Make sure it exists.
4	12	An enqueue error occurred reading the LOGSAVE record. Try the process again.
4	16	The database contains more than one LOGSAVE record. Delete one.
4	20	Not enough storage to read the LOGSAVE record. Check the region size.
4	24	The LOGSAVE record is busy. Try again, or run DBCLEANUP.
4	28	Cannot read the LOGSAVE record. The record is damaged. Delete it and recreate it.
8	4	Internal control blocks were not found.
8	21	The status of the LOGSAVE record is disabled. Enable it.
12	4	Cannot locate database 5. Check that the correct session-parameters member is being used.
16	0	Record with RNID LOGSAVE is not LOGSAVE type. Delete it and create the LOGSAVE record using the procedures described in Propagating to a Tivoli Information Management for z/OS Database.
20	8	The **Data set name** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.
20	12	The **Block size** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.
20	16	The **Logical record length** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.
20	20	The **Primary allocation blocks** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.
20	24	The **Secondary allocation blocks** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.

Table 38. BLGUT3WT Return and Reason Codes (continued)

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
20	36	The **Unit type** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.
20	56	The **Synchronize Send and Receive?** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.
20	60	The **ABEND disposition** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.
20	68	The **Receive filter data** field was not found in the LOGSAVE record when a **Receive filter prefix** field was. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged.
24	4	A numeric field is not all numbers.
24	8	The LOGSAVE record has *send* in the **Database type** field and this user exit performs receive processing. Correct the record.

BLGUT4EX–Offload a Recovery Data Set

Offloads the recovery data set (SDLDS) into a send data set. The BLGDUMP1, BLGTDBXM, and BLGTDBXM1 TSPs contain examples of using this user exit.

Input TSCA variable data area output by BLGUT4WT.

The SDLDS.

Optional parameters RECS= and %FULL = in the **Literal/Test data** field on the Data Field Specification panel of the calling USEREXIT control line.

RECS=*nn*
An optional parameter that specifies the number of physical records to be processed each time an enqueue is obtained on the SDLDS before releasing the enqueue. *nn* can range from 0 to 99 999 999. A value of 0 indicates that the SDLDS enqueue is not to be released until the offload is complete. The default value is 0.

%FULL=*xxx*
An optional parameter that specifies if, during offload processing, the SDLDS becomes more than *xxx* percent full, processing is to continue without releasing the enqueue on the SDLDS until the offload is complete. *xxx* can range from 0 to 100. The default value is 80. If RECS=*nn* is omitted or has a value of 0, then %FULL= is ignored.

Output
A send data set.

Environment

TSP only.

A return code and reason code as listed in Table 39. The user exit writes informational and error messages to SYSPRINT. If you allocate SYSPRINT to a data set, the record format must be Variable Blocked ANSI (VBA).

Table 39. BLGUT4EX Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion
8	1	Internal control blocks were not found.
8	2	Error opening the recovery data set (SDLDS). Look for other messages.
8	3	Error allocating the send data set. Look for other messages.
8	4	Error opening the send data set. Look for other messages.
8	5	Error performing VSAM set on the SDLDS. Look for other messages.
8	6	Error getting an SDLDS record. Look for other messages.
8	7	SDLDS record is not valid. Look for other messages.
8	8	Error writing to the send data set. Look for other messages.
8	9	Error updating the SDLDS control record. Look for other messages.
8	10	Error reading the SDLDS control record. Look for other messages.
8	11	Error closing the SDLDS. Look for other messages.
8	12	Error closing the send data set. Look for other messages.
8	13	Error freeing the send data set. Look for other messages.
8	14	No SDLDS found. Look for other messages. Check whether you are using the correct session-parameters member.
8	15	Error opening the SDIDS. Look for other messages.
8	16	Error closing the SDIDS. Look for other messages.
8	17	Error reading the SDIDS. Look for other messages.
8	18	Error writing to the SDIDS. Look for other messages.
8	19	Error performing VSAM set on the SDIDS. Look for other messages.
8	20	Error processing optional parameters. Look for other messages.

BLGUT4WT–Initialize for Send

Initializes the TSCA variable data area and waits for the interval specified in the LOGSAVE record. The BLGDUMP1 and BLGTDBXM TSPs contain examples of using this user exit.

Input LOGSAVE record for a send database.

Output
TSCA variable data area.

Environment
TSP and TSX.

A return code and reason code as listed in Table 40. The user exit writes informational and error messages to SYSPRINT. If you allocate SYSPRINT to a data set, the record format must be Variable Blocked ANSI (VBA).

Table 40. BLGUT4WT Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
4	4	An error occurred reading the LOGSAVE record.
4	8	The LOGSAVE record was not found. Make sure it exists.
4	12	An enqueue error occurred reading the LOGSAVE record. Try the process again.
4	16	The database contains more than one LOGSAVE record. Delete one.
4	20	Not enough storage to read the LOGSAVE record. Check the region size.
4	24	The LOGSAVE record is busy. Try again or run DBCLEANUP.
4	28	Cannot read the LOGSAVE record. The record is damaged. Delete the record and recreate it.
8	4	Internal control blocks were not found.
8	20	The status of the LOGSAVE record is disabled. Enable it.
12	4	Cannot locate database 5. Check that the correct session-parameters member is being used.
16	0	Record with RNID LOGSAVE is not LOGSAVE type.
20	12	The **Block size** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.
20	16	The **Logical record length** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.
20	20	The **Primary allocation blocks** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.

Table 40. BLGUT4WT Return and Reason Codes (continued)

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
20	24	The **Secondary allocation blocks** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.
20	36	The **Unit type** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.
20	56	The **Synchronize Send and Receive?** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.
20	60	The **ABEND disposition** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.
20	64	The **Data set name** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged. Verify and correct the missing information.
24	4	A numeric field is not all numbers.
24	8	The LOGSAVE record has *receive* in the **Database type** field and this user exit performs send processing.

BLMSSGEN–SQL Setup, Extract, and Cleanup

Converts Tivoli Information Management for z/OS records in an Automatic Log Save send data set format to SQL statements for propagating the Tivoli Information Management for z/OS data to a DB2 database. The BLGTDBXM and BLGTDBXM1 TSPs contain examples of using this user exit.

The user exit has three modes: setup, extract, and cleanup. For maximum performance, do setup and cleanup once for multiple calls to BLMSSGEN extract. The user exit writes informational and error messages to SYSPRINT. If you allocate SYSPRINT to a data set, the record format must be Variable Blocked ANSI (VBA).

Setup Reads the LOGSAVE record and the RDMTs that are identified in the LOGSAVE record and stored in a data set in the report format table concatenation. Allocates storage for extract processing.

Input A value of **1** in the **Literal/Test data** field on the Data Field Specification panel of the calling USEREXIT control line.

LOGSAVE record defined for DB2 Extract processing.

Report format table data set concatenation containing the RDMTs specified in the LOGSAVE record.

Output

BLMSSGEN environment established.

Environment

TSP only.

If an error occurs reading or validating an RDMT, TSCAUFLD is set to the name of the RDMT in error.

A return and reason code as listed in Table 41.

Extract

Reads Tivoli Information Management for z/OS records from the send data set identified in the LOGSAVE record and generated by user exit BLGUT4EX.

For each record read from the send data set, attempts to find the record type s-word identified in each RDMT. Uses the first RDMT that matches.

If no RDMTs match the Tivoli Information Management for z/OS record, continues to the next record.

For each field contained in the RDMT that matches a field in the Tivoli Information Management for z/OS record, takes the field data and generates SQL statements in the SQL data set.

Input A value of **2** in the **Literal/Test data** field on the Data Field Specification panel of the calling USEREXIT control line.

TSCAVDAP pointing to data created by user exit BLGUT4WT.

Environment set up by BLMSSGEN setup.

LOGSAVE record defined for DB2 Extract processing.

Send data set that was output by BLGUT4EX.

Output

SQL data set.

Environment

TSP only.

A return and reason code as listed in Table 41.

Cleanup

Frees storage obtained during BLMSSGEN setup.

Input A value of **3** in the **Literal/Test data** field on the Data Field Specification panel of the calling USEREXIT control line.

Environment set up by BLMSSGEN setup.

Output

BLMSSGEN environment freed.

A return and reason code as listed in Table 41.

Environment

TSP only.

Table 41. BLMSSGEN Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion. SQL statements generated.

Table 41. BLMSSGEN Return and Reason Codes (continued)

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	4	No records were in the send data set. No SQL statements generated.
0	8	No SQL statements generated, including the UPDATE statements for synchronization. All records skipped. No processing errors.
0	12	At least one record skipped because of processing errors. SQL statements generated unless all records were skipped.
4	4	Incorrect or null **Literal/Test data** field. One character must be specified in the **Literal/Test data** field of the calling USEREXIT control line and it must be a value of 1, 2, or 3.
4	8	Performing extract processing and TSCAVDAL is not equal to the length set by user exit BLGUT4WT.
8	4	Performing setup processing and the environment is already established.
8	8	BLMSSGEN environment not established. Perform setup before performing extract or cleanup.
8	12	The API is active. BLMSSGEN cannot be called as part of an API transaction.
12	4	Out of storage.
16	4	Internal control blocks were not found. Contact Tivoli.
16	8	Internal error detected. Contact Tivoli.
16	16	BLMSSGEN environment damaged. Contact Tivoli.
20	4	RFT data set not specified in the session-parameters member.
20	8	Error allocating the RFT data set. Look for other messages.
20	12	Error opening the RFT data set. Look for other messages.
20	16	Error closing the RFT data set. Look for other messages.
20	20	Record format of RFT data set is not fixed.
20	24	RDMT was not found in the RFT data set. TSCAUFLD contains the name of the RDMT.
20	28	Error reading RDMT from RFT data set. TSCAUFLD contains the name of the RDMT.

Table 41. BLMSSGEN Return and Reason Codes (continued)

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
20	32	RDMT is not valid or is not of correct type. For example, it is not an RDMT, the name stored in the RDMT header does not match the name of the RFT data set member, or the RDMT is truncated. TSCAUFLD contains the name of the RDMT. Verify that name of the RFT data set member matches the name you used on the TABLE statement when you ran BLGUT8 to build the RDMT. Verify that the name you put in the **RDMT name** field of the LOGSAVE record is correct.
20	36	No record type s-word found in the RDMT. TSCAUFLD contains the name of the RDMT.
20	48	Logical record length of RFT data set is not 80.
20	52	Unknown error allocating the RFT data set, or an unknown error reading the RDMT. Verify that your session-parameters member allocates an RFT data set that contains RDMTs. Verify that the names in the **RDMT name** field of the LOGSAVE record are typed correctly and that they are in the RFT data set allocated to your session-parameters member.
20	56	Error freeing the RFT data set.
24	4	A database error occurred reading the LOGSAVE record.
24	8	The LOGSAVE record was not found.
24	12	An enqueue error occurred reading the LOGSAVE record.
24	16	The database contains more than one LOGSAVE record.
24	20	Not enough storage to read the LOGSAVE record.
24	24	The LOGSAVE record is busy.
24	28	Cannot read the LOGSAVE record. The record is damaged. Delete the record and recreate it.
24	32	Record with RNID LOGSAVE is not LOGSAVE record type. Delete it and create the LOGSAVE record using the procedures described in Propagating to a Tivoli Information Management for z/OS Database and Propagating to a DB2 Database.
24	36	No RDMTs specified in the **Mapping** field in the LOGSAVE record. BLMSSGEN stops at the first empty slot in the list of RDMTs. Verify that the first slot is not empty.

Table 41. BLMSSGEN Return and Reason Codes (continued)

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
24	40	The send **Data set name** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged.
24	44	The **SQL data set name** field was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged.
24	48	The **Logical record length** field for the SQL data set was not found in the LOGSAVE record, or the length is 0. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged.
24	52	The **Block size** field for the SQL data set was not found in the LOGSAVE record, or the size is 0. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged.
24	56	The **Primary allocation blocks** field for the SQL data set was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged.
24	64	The **Unit type** field for the SQL data set was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged.
24	68	The **ABEND disposition** field for the SQL data was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged.
24	72	Cannot read the LOGSAVE record. Unknown reason.
24	76	The **Authorization ID** field for the DB2 table was not found in the LOGSAVE record. Either the information is not in the LOGSAVE record, or the TSCA variable data area is damaged.
28	8	Error allocating the send data set. Look for other messages.
28	12	Error opening the send data set. Look for other messages.
28	16	Error closing the send data set. Look for other messages.
28	28	Error reading the send data set. Look for other messages.
28	56	Error freeing the send data set.
32	8	Error allocating the SQL data set. Look for other messages.
32	12	Error opening the SQL data set. Look for other messages.

Table 41. BLMSSGEN Return and Reason Codes (continued)

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
32	16	Error closing the SQL data set. Look for other messages.
32	20	Error writing the SQL data set. Look for other messages.
32	56	Error freeing the SQL data set.
40	4	Cannot locate database 5. Verify that you specified the correct session-parameters member.

17

Remote Data Resources in Tivoli Information Management for z/OS

A *remote data resource* is a means by which data can be passed between TSXs running in different Tivoli Information Management for z/OS address spaces. You use the PUTRDATA and GETRDATA TSX control lines to move data to and from a remote data resource. You use other TSX control lines or the RDR BLX-SP command to monitor or modify the state of a remote data resource. If you are using the message queueing described in "Sending E-mail Messages" on page 95, you can also use the BLGTNPQM TSX to put message information on a remote data resource and the BLGTNRQM TSX to query the queues to retrieve the messages.

A remote data resource is identified by a unique name. Any number of remote data resources can be defined, and each resource can contain any number of items (character strings), limited by the amount of storage available to the BLX-SP address space. The number of items that a resource can contain can be limited to a specified count, thus ensuring that available storage is not exhausted. The length of each item is limited to no more than 512 bytes because the TSCAVDA variable data area is used to return items to a TSX.

Any number of TSXs can put items in a resource at the same time; however, only one TSX can get items out of a resource at a time. Because of this limitation, the GETRDATA TSX control line obtains and releases exclusive control of an ENQ for the resource for each get request. In addition, if you want your TSX to wait for an item when a resource is empty, that TSX must explicitly open the resource. This gives that TSX "ownership" of the resource, and only that TSX can issue GETRDATA requests with the WAIT or time interval parameter. Any TSX can issue the GETRDATA request against any resource, but if there is another TSX with an active GETRDATA request for the same resource, the second GETRDATA request is rejected.

When your TSX finishes all processing for a resource, the resource can be left in one of two states: it can be CLOSED or it can be RELEASED. You use the CLOSERRES FLUSH TSX control line or the RDR FLUSH BLX-SP command to change the state of a resource to closed. You use the RELEASERRES TSX control line or the RDR RELEASE BLX-SP command to change the state of a resource to released. When a resource is FLUSHED, all items remaining in the resource are discarded, and any active GET requests and any subsequent GET requests are rejected. When a resource is RELEASED, existing items are not discarded; however, if there is an active GET request, it is rejected. If a TSX that obtained ownership of a resource terminates without issuing the CLOSERRES TSX control line and the resource is not closed, the resource is flushed and its state is set to CLOSED.

Both the Tivoli Service Desk Bridge (the link between Tivoli Information Management for z/OS and Tivoli Service Desk) and the Tivoli Information Management for z/OS interface to

Tivoli Software Distribution utilize remote data resources; both of these interfaces are described in the *Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications*.

Functional Overview

TSX control lines that manipulate the remote data resources are described in the *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference*. The RDR command used to monitor or modify the state of a remote data resource is described in the *Tivoli Information Management for z/OS Operation and Maintenance Reference*.

This is an overview:

Creating a remote data resource
A resource is explicitly created with specific processing limits via the OPENRRES TSX control line. A resource is implicitly created with default processing limits when an attempt is made to remove data from a new Resource via the GETRDATA TSX control line.

Setting Processing Limits for a remote data resource
Processing limit values can be set for a remote data resource via the SETRRES TSX control line or the BLX-SP RDR SET command. Limits that can be set include the maximum number of items that can be added to a resource, or a value for triggering a warning message when an item count threshold has been reached.

Adding an Item to a remote data resource
Items can be added to a remote data resource by using the PUTRDATA TSX control line. The string specified as the data parameter to this control line is placed in the remote data resource.

Removing an Item from a remote data resource
Items can be removed from a resource by using the GETRDATA TSX control line. The removed item is placed in the TSCA variable data area. Execution of a TSX can be suspended either indefinitely or for a specified period of time until data is put into a resource by using the WAIT parameter.

Modifying the State of a remote data resource
The state of a remote data resource can be changed by using either the CLOSERRES FLUSH/DRAIN or RELEASERRES TSX control lines or the RDR FLUSH/DRAIN/RELEASE BLX-SP commands. FLUSH causes all items in a resource to be discarded, terminate the active GETRDATA request (if any), and the state of the resource to be set to CLOSED. DRAIN prohibits new items from being added to a resource and sets the state of the resource to DRAINING. RELEASE terminates an active GETRDATA request and sets the state of the resource to RELEASED. If your TSX issues the OPENRRES TSX control line and terminates without issuing either the CLOSERRES or RELEASERRES TSX control lines, the resource is flushed and its state is set to CLOSED.

Querying a remote data resource
Both the RDR QUERY BLX-SP command and the QUERYRRES TSX control line can be used to display or retrieve the processing limits, resource state, and related data about a resource.

18

The Tivoli Information Management for z/OS Archiver

The Archiver is a Tivoli Information Management for z/OS utility which allows you to manage records on your Tivoli Information Management for z/OS databases. The following functions are provided:

Archive
Move records from one database to another. Refer to “Sample 1” on page 349 for sample BLGARCP specifications.

Copy Copy records from one database to another (refer to “Sample 2” on page 350) or copy records from one database back to the same database or to another database with a new RNID (refer to “Sample 3” on page 351).

Purge Delete records from a database. Refer to “Sample 4” on page 351 for sample BLGARCP specifications.

Note: The term “live” database refers to the existing, or “from” database, and the “archive” database refers to the new, or “to” database.

The advantage of using the Archiver instead of other methods of manipulating records on a database is that you can run the Archiver against a database while other users are interactively using Tivoli Information Management for z/OS on the same database. You are not required to run BLGUT1 or any other Tivoli Information Management for z/OS utilities in conjunction with running the Archiver.

In addition, the Archiver provides ample logging functions, including:

- The parameters that you specified when you invoked the Archiver
- Messages related to each record processed
- A completion summary of what the Archiver has successfully done
- Full LLAPI (Low-Level Application Programming Interface) logging
- Internal analysis of each record processed with details of all errors encountered

Functional Overview of the Archiver

The Archiver is a LLAPI job that makes use of dynamic program interface data tables (PIDTs). You can run the Archiver while other users are interactively accessing the databases. The “archive” database must be accessed as database 5, read/write. By default, the “live” database is also accessed as database 5, read/write, but it may be accessed as read-only if the DBID= parameter is specified during FUNCTION=COPY.

The Archiver's functions are controlled by parameters passed to it at invocation. The sample JCL shown in Figure 19 should be modified to fit your environment. Once modified, it can be used to invoke the Archiver.

```
//BLGARC JOB
//*                                                                    *
//ARCHIVE EXEC PGM=BLGARC,REGION=6M
//*
//* Do not forget to STEPLIB to your session load library if needed.
//*
//STEPLIB DD DSN=BLM.SBLMMOD1,DISP=SHR
//*
//* BLGARCP DD is the input parameters data set and is required.
//*
//BLGARCP DD DSN=????????.BLGARCP,DISP=SHR
//*
//* BLGINREC DD is needed only if INPUT=EXTERNAL is coded in the
//* data set defined by the BLGARCP DD.
//*
//BLGINREC DD DSN=????????.BLGINREC,DISP=SHR
//*
//* BLGOPREC DD is the data set where the RNIDs of the
//* records archived (created) are stored.
//* This data set can be used an input to another archive job.
//*
//BLGOPREC DD DSN=????????.BLGOPREC,DISP=OLD
//*
//*RFTDD DD HAS YOUR PIDT EXECUTABLE TABLES
//*
//RFTDD DD DSN=BLM.SBLMFMT,DISP=SHR
//APIPRINT DD SYSOUT=*
//*
//* BLGOUT DD is the data set where Archiver messages are routed.
//*
//BLGOUT   DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
```

Figure 19. Sample JCL to Invoke the Archiver. The sample JCL can be found in the SBLMSAMP data set as member BLGARCJ.

The parameters enable you to initialize and control the use of the Program Interface Communications Area (PICA), which in turn controls the functions available within the LLAPI. For example, the parameters you specify in the BLGARCP data set define such items as the number of records to process and the search argument to be used for inquiry, as well as events such as the transfer of text and journal data in the records. (All of the parameters and their values are described in "Input Parameters" on page 340.)

The Archiver automatically updates both the SDDS (structured description data set) and SDIDS (structured description index data set) so there is no need to run any additional utilities, such as BLGUT1.

The Archiver uses several PIDTs to control inquiry, retrieve, and create functions. The sample BLGARCP shown in Figure 21 on page 350 uses the inquiry PIDT for problem records, BLGYPRI, which is a member of the SBLMFMT data set. The source code for this data table (PIDT) and other PIDTs can be found in the SBLMSAMP library. The SBLMFMT and SBLMSAMP libraries are described in the *Tivoli Information Management for z/OS Application Program Interface Guide*. You should review the sample PIDTs to ensure that they contain the values that you need. If they do not, you should create your own, modeled on the samples. More information on modifying PIDTs can be found in the *Tivoli Information Management for z/OS Application Program Interface Guide*.

The inquiry PIDT can contain a high level inquiry only, such as the record type. The specific search argument can be passed to the Archiver as a parameter list to form a freeform search. The retrieve and create PIDTs are built dynamically by the LLAPI, and automatically represent an exact data map of the retrieved record. Once retrieved, each record can be compressed and entries removed or uncognized prior to the create transaction. In addition, the create transaction can be requested to always issue a new system-assigned RNID.

To determine the records to be processed, the Archiver can use either specific search arguments or an external data set.

- Using search arguments, an inquiry is performed using the inquiry PIDT in conjunction with the freeform argument specified by the ARG= parameters in the parameter data set BLGARCP. The values supplied by the ARG= parameters form the freeform search argument for the inquiry. An inquiry PIDT must contain the s-word (structured word) of the record type to be processed.
- The external data set utilizes a “header” line and explicit record identifiers. The data set contains a sequential list of Tivoli Information Management for z/OS record number identifiers (RNIDs). This data set can be created “manually” or it can be created with an RFT (report format table) which could perform all the necessary logic searching, especially when processing complex parent/child relationships. Refer to Figure 29 on page 355 for sample RFT code to build the BLGINREC data set.

The Archiver logs the startup parameters that you specified, messages related to each record processed, and a completion summary. Logging is controlled by the LOGGING= parameter specified in the BLGARCP parameter data set. Archiver messages are written to the data set defined by the BLGOUT DD statement and LLAPI messages are written to the data set defined by the APIPRINT DD statement, assuming that these data sets were allocated and LOGGING=Y was specified in the BLGARCP parameter data set.

The Archiver creates the requested records in the “archive” database and automatically checks them in on both databases when CHECKIO=Y.

Input Data Sets

DDNAME
  Purpose

BLGARCP
  Input parameter data set; it may be specified in-stream if desired. See the example in Figure 21 on page 350. It should have the following characteristics:

```
RECFM=FB     LRECL=80
```

BLGINREC
  Input data set containing the RNIDs to be processed when INPUT=EXTERNAL is specified. The first line in the data set, beginning in column 1, should contain the word *ARCHIVE* or *COPY* or *PURGE* matching the FUNCTION= value in the BLGARCP DD statement. The input may be specified in-stream if desired. It should have the following characteristics:

```
RECFM=FB     LRECL=80
```

RFTDD
Input data set containing the PIDTs.

Output Data Sets

DDNAME
Purpose

BLGOPREC
Output data set containing the RNIDs copied as part of user-defined relationship processing. The first record written to the BLGOPREC data set is always a *PURGE* entry. This data set should be used as the BLGINREC data set for a subsequent *PURGE*. You should allocate the BLGOPREC data set prior to running the Archiver, and it should have the following characteristics:

```
RECFM=FB     LRECL=80
```

For additional information on this data set, see "Parent/Child Relationships" on page 352.

BLGOUT
Output data set where all Archiver messages are written. It should have the following characteristics:

```
RECFM=VBA    LRECL=137
```

APIPRINT
Output data set where all LLAPI messages are written when LOGGING=Y is specified. The *Tivoli Information Management for z/OS Application Program Interface Guide* contains additional information on the APIPRINT data set.

Return Codes

0 The Archiver processed all of the requested records successfully.

4 The Archiver encountered a non-zero PIDTCODE, PIHTCODE, or PIRTCODE. The record in error is skipped, and processing continues until all of the requested records have been processed. Archiver messages are issued describing the exact failure.

8 The Archiver encountered an error and cannot continue. Archiver messages are issued describing the exact failure.

Input Parameters

Attention: If you specify any of these parameters more than once, only the last value specified is be used by the Archiver program. In certain circumstances, this can also cause unpredictable results.

These are the parameters that can be specified with the Archiver:

FUNCTION=
Defines the function of the Archiver program. Valid options are:

ARCHIVE
Move the specified records from the live database to the archive database and delete the original record from the live database.

COPY
Copy the specified records from the live database to the archive database. The records on the live database are left intact.

PURGE
Delete the specified records from the live database. The archive database is not required or affected. If the Archiver encounters a record that cannot be found (Return Code 12 Reason Code 10 on the API delete transaction), the record is skipped, the Archiver continues processing with the next record, and the Archiver finishes with return code 4 (RC4).

The default is ARCHIVE.

PURGE= Defines the type of delete processing to be used by the ARCHIVE option. Valid values are:

NORMAL
Delete the record from the live database only if the record is created on the archive database with a Return Code=0.

FORCE
Delete the record from the live database if the record is created on the archive database with a Return Code=0 OR if the create function fails because a record already exists on the archive database with the same RNID.

The default is NORMAL.

INPUT= Defines the source of the RNIDs to be processed. Valid values are:

INTERNAL
Use the search arguments specified in the input parameter member (up to 10 ARG= may be specified) in association with the INQPIDT parameter to locate the required records on the live database.

EXTERNAL
Use the RNIDs specified in the input data set allocated to DDNAME BLGINREC. The first record read from DDNAME BLGINREC must match the function specified in the FUNCTION= parameter.

The default is INTERNAL.

CLASS= Defines one Privilege Class name that exists on both live and archive databases and is used for live and archive sessions. This class must have Inquiry, Create, Update, Delete, and database administration authority to the specified records on the live database, and Create, Update, and database administration authority on the archive database.

You can override the Privilege Class name on the archive database with the ARCCLASS parameter.

This is a required parameter, and no default value is supplied. Sets PICAPRIV for the API transactions that will be performed on the archive database. Refer to the *Tivoli Information Management for z/OS Application Program Interface Guide* for additional information on the PICAPRIV field.

ARCCLASS= Defines the Privilege Class name that exists on the archive database. This privilege class must have Create, Update, and database administration authority to the specified records on the archive database.

This is an optional parameter, and no default value is supplied. Sets PICAPRIV for the API transactions that will be performed on the archive database. Refer to the *Tivoli Information Management for z/OS Application Program Interface Guide* for additional information on the PICACLSN field.

LIVESESS= Defines the session parameter member name to access the live database; for example, BLGSES00. The "live" database must be accessed as database 5, the read/write database, if FUNCTION=ARCHIVE or FUNCTION=PURGE. With FUNCTION=COPY, the database can be accessed as read-only if desired assuming that CHECKIO=Y is not specified.

This is a required parameter, and no default value is supplied.

ARCSESS= Defines the session parameter member name to access the archive database; for example, BLGSESAR. The "archive" database must be accessed as database 5, the read/write database. The archive session is only initialized with FUNCTION=ARCHIVE and FUNCTION=COPY.

This is a required parameter if FUNCTION=ARCHIVE or FUNCTION=COPY, and no default value is supplied.

DBID= Defines a read-only database (4, 7, 8, or 9) for the LIVESESS. Specifying this parameter enables you to copy records from your read-only database to your read/write database. To do this, you use the same session parameters member for both the LIVESESS and the ARCSESS. This is only valid when FUNCTION=COPY.

This parameter is optional and is ignored if FUNCTION=COPY is not specified. Database 5 is assumed for both sessions if DBID is not specified. This parameter sets PICADBID for the API transactions that will be performed on the live database. Refer to the *Applications Program Interface Guide and Reference* for additional information on the PICADBID field.

INQPIDT= Defines the name of the inquiry PIDT used to locate the records to be processed when INPUT=INTERNAL. This PIDT is used in association with the search arguments specified in the ARG= parameters.

This is a required parameter when INPUT=INTERNAL, and this parameter is ignored when INPUT=EXTERNAL. No default value is supplied.

DYNMPIDT=

Defines the name of the dynamic PIDT used by the Retrieve and Create functions to copy and load the record into the archive database. This PIDT should only contain the USE(HEADER) parameter. The PIDT BLGYDYN is supplied in SBLMFMT for this purpose and the default is DYNMPIDT=BLGYDYN.

This is a required parameter.

LOGGING= Defines whether to perform API logging. Valid values are:

Y A value of Y causes full logging to DDNAME APIPRINT.

N A value of N suppresses logging.

The default is N. This parameter sets the PICAMSGD parameter to P. For additional information, refer to the *Tivoli Information Management for z/OS Application Program Interface Guide*.

ARG= Defines up to 10 freeform arguments that are used when INPUT=INTERNAL. The maximum length of each argument (including the qualifier) is 33 characters. The arguments specified are used in addition to the inquiry PIDT. Each argument must be prefixed by an ARG= statement. A period (.) at the end of an argument specifies an abbreviated search. A Boolean operator, if specified, must be in the first position of the argument. An argument cannot contain leading or imbedded blanks. Examples of arguments are:

```
ARG=¬STAC/CLOSED
ARG=DATE/1999/01/01
ARG=-DATE/1999/10/31
ARG=PRIO/1
ARG=|PRIO/1
ARG=¬DEPT/T53B
ARG=STAC/CL.
ARG=DATE/1999/**/**
```

This parameter is optional and it is ignored when INPUT=EXTERNAL. No default value is supplied. Although this parameter is optional, you may want to specify it to more closely identify the data you want to process. If you omit this parameter, the Archiver searches against all of the records defined by the INQPIDT= parameter.

APPLID= Defines the application identifier (APPLID) of the Archiver, and must exist in the Privilege Class defined by the CLASS parameter and the ARCCLASS parameter (if specified). This parameter is optional, and the default is APPLID=ARCHIVER.

CHILD= Defines the p-word (prefix word), including the slash (/), that is to be used to locate child records by performing a search of p-word/parent_rnid. The p-word must be 5 characters in length, including the slash. Omitting this parameter causes child processing to be bypassed. When using INPUT=INTERNAL and specifying this parameter, each record is checked for associated child records. This can seriously affect the performance of the Archiver. This method should only be used for Tivoli-defined parent/child relationships. See “Parent/Child Relationships” on page 352 for additional information. Multiple child types are located with the CHILD= argument. If you have multiple child record types associated with a single parent, you should code an RFT to locate all of the child RNIDs.

This parameter is optional, but if you specify this parameter, you must also specify the PARENT= and CHDPIDT= parameters.

PARENT= Defines the s-word (structured word) index pointer contained in the parent record pointing to the child record.

Note: When specifying the s-word index, prefix the character **S** to the index. For example, if the s-word index is 0CBC, you must specify **S**0CBC. See the example on Figure 28 on page 353.

Usually the associated p-word is *NAMA*. This enables the inquiry function to load the pointer to the child record without having to perform a search against every parent. If associated data is located, a search is performed and all resulting child records are also processed. This method should only be used for Tivoli-defined parent/child relationships. See "Parent/Child Relationships" on page 352 for additional information.

This parameter is optional, but if you specify this parameter, you must also specify the CHILD= and CHDPIDT= parameters.

CHDPIDT= Defines the name of the PIDT used to locate child records. Usually, this PIDT would contain the child record type s-word. This method should only be used for Tivoli-defined parent/child relationships. See "Parent/Child Relationships" on page 352 for additional information.

This parameter is optional, but if you specify this parameter, you must also specify the PARENT= and CHILD= parameters.

USERRELN= Defines whether or not you wish to process records with user-defined relationships. The valid values are:

Y The records to be processed contain user-defined relationships. Allocate the DDNAME BLGOPREC in the BLGARC JCL. As each record is processed, its RNID is written to the BLGOPREC data set.

N The records to be processed do not contain user-defined relationships.

This is an optional parameter, but if you specify Y then FUNCTION=COPY and INPUT=EXTERNAL must also be specified. The default is N. You should specify CHECKIO=Y when using this parameter.

See "Parent/Child Relationships" on page 352 for additional information.

HITS= Defines the number of parent records to be processed. The value specified must be an 8-digit decimal number between 00000001 and 99999999. Child records are not included in this number. This parameter controls execution times by limiting the number of records to be processed.

This parameter is ignored when INPUT=EXTERNAL and defaults to 00000001 when INPUT=INTERNAL. The PICA field PICANUMH is set by the HITS= parameter.

CHECKIO= Defines whether to perform check-in/check-out for each record processed. This parameter allows you to check out records on the live database before starting the archive process. The valid values are:

Y Specifies that records should be checked out before archive processing starts and checked in when the archive process completes. The API check-in transaction is not performed to check in the records created on the archive database. Instead, the checked out entry in a retrieved record is ignored on the subsequent create transaction.

N Specifies that no check-in or check-out processing is requested.

This parameter is optional, and the default is Y.

If you do not specify CHECKIO=N, the default of CHECKIO=Y is taken and may cause unnecessary overhead.

Setting CHECKIO=Y prevents two integrity problems:

- It prevents a user from updating a record on the live database between the retrieve and delete transactions.
- It prevents the delete transaction from failing because a user is updating the record.

It is unlikely that either of these situations will occur during normal archive processing.

When using FUNCTION=COPY or FUNCTION=PURGE, setting this parameter to N can greatly increase the throughput of records processed, but setting this parameter to N with FUNCTION=ARCHIVE may cause data integrity problems. You should always specify CHECKIO=Y when USERRELN=Y. See “Parent/Child Relationships” on page 352 for additional information.

COMPRESS=

Defines whether to remove non-essential information from records. Valid values are:

Y Compress entries from the records.

N Do not compress entries from the records.

This parameter is optional, and the default value is N.

Entries in the records which contain a PIDTRDEF value of O are removed before the archive version of the record is created. These entries signify that the SDE for the row does not contain an s-word or a p-word. Refer to the *Tivoli Information Management for z/OS Application Program Interface Guide* for additional information on PIDTRDEF.

Refer to “Record Processing Considerations” on page 356 for additional information on when to code this parameter.

DELENTRY=

Defines whether to remove redundant, not valid, or unwanted entries from records. Defines up to 40 s-word indexes which identify the data in your records that you want deleted from your records before they are created on the archive database. For example, if you want to delete the reporter name and status from a problem record, specify:

```
DELENTRY=S0B59
DELENTRY=S0BEE
```

The s-word indexes that you specify are compared with the PIDTSYMB field of each PIDT entry for each record. An s-word index match causes the PIDT entry to be ignored when the record is created on the archive database. The s-word index specified must be 5 characters prefixed with an S. An s-word index of S0000 is ignored.

This parameter is optional, and no default value is supplied.

Refer to “Record Processing Considerations” on page 356 for additional information on when to code this parameter.

UNCOGNIZE=

Defines whether to uncognize entries in a record. Defines up to 40 s-word indexes which identify the data that you want uncognized in your records before they are created on the archive database. For example, if you want to uncognize the reporter name and status from a problem record, specify:

```
UNCOGNIZE=S0B59
UNCOGNIZE=S0BEE
```

The s-word indexes that you specify are compared with the PIDTSYMB field of each PIDT entry for each record. An s-word index match causes the PIDT entry to be uncognized when the record is created on the archive database. The s-word index specified must be 5 characters prefixed with an S. An s-word index of S0000 is ignored.

This parameter is optional, and no default value is supplied.

Refer to "Record Processing Considerations" on page 356 for additional information on when to code this parameter.

NEWRNID= Defines whether the RNID of the record to be processed should be replaced with the next available system-assigned RNID on the archive database. This is valid for FUNCTION=COPY and FUNCTION=ARCHIVE.

Y The next system-assigned RNID on the archive database replaces the RNID currently assigned to the record when it is created on the archive database.

N Keep the RNID of the record.

This parameter is optional and the default value is N.

Note: By specifying NEWRNID=Y, the RNID of the retrieved record is always replaced by the next system-assigned RNID on the archive database regardless of whether or not the RNID exists on the archive database or if the record has a user-assigned RNID. Do not set NEWRNID=Y for records which contain relationships, or the relationships will be broken.

TEXT= Defines whether freeform text from the records being processed should be copied. Valid values are:

Y Process the freeform text with the record.

N Do not process the freeform text with the record.

This parameter is optional and the default is Y.

TEXTXFER=

Defines the transfer method used when processing record freeform text. Valid values are:

D Use a data set to transfer the freeform text.

B Use a buffer to transfer the freeform text.

This parameter is optional and the default is D.

For performance reasons, a value of B is desirable. If you specify D, it is your responsibility to delete the data sets created by the archive job. For more details, see PICATXTP in the *Tivoli Information Management for z/OS Application Program Interface Guide*.

TEXTAUD= Defines whether the freeform text audit trail should be processed when processing record text. Valid values are:

Y Preserve the freeform text audit trail.

N Discard the freeform text audit trail.

This parameter is optional and the default is Y.

For additional details, see PICATXAU in the *Tivoli Information Management for z/OS Application Program Interface Guide*.

TEXTCNT= Defines the number of text lines to process between 00000001 and 99999999 when TEXTXFER=B.

This value must be 8 characters in length, containing leading zeroes as necessary. The default is 00000200.

For additional details, see PICATXTU in the *Tivoli Information Management for z/OS Application Program Interface Guide*.

TEXTWID= Defines the width of the text to process between 00000001 and 00000132 when TEXTXFER=B.

This value must be 8 characters in length, containing leading zeroes as necessary. The default is 00000060. For additional details, see PICATXTW in *Tivoli Information Management for z/OS Application Program Interface Guide*.

TEXTAREA=

Defines the area of text to transfer when using TEXTXFER=B and the total number of text lines exceeds the amount specified in TEXTCNT. Valid values are:

T The top of the text is copied.

B The bottom of the text is copied.

The default is B.

For additional details, see PICATXTA in the *Tivoli Information Management for z/OS Application Program Interface Guide*.

TIMEOUT= Defines a value which is the number of seconds in which the transaction must complete. If you specify a value of 0 or omit this field, the value is set to 00000300 seconds (five minutes). This value sets the PICATINT parameter, which is defined in the *Tivoli Information Management for z/OS Application Program Interface Guide*.

HISTORY= Defines whether to process history (journal) data in the record. Valid values are:

Y Preserve the history data.

N Discard the history data.

The default is Y.

For additional details, see PICAHIST in the *Tivoli Information Management for z/OS Application Program Interface Guide*.

Required Parameters

Depending on the FUNCTION= specified, certain parameters may be required.

Parameter	*FUNCTION=*			*Default*
	ARCHIVE	COPY	PURGE	
---------	---------	---------	---------	---------
PURGE=	Optional	N/A	N/A	NORMAL
INPUT=	Optional	Optional	Optional	INTERNAL
CLASS=	Required	Required	Required	None
ARCCLASS=	Optional	Optional	N/A	None
LIVESESS=	Required	Required	Required	None
ARCSESS=	Required	Required	N/A	None
APPLID=	Optional	Optional	Optional	ARCHIVER
INQPIDT=	Required if no value is specified for the INPUT= parameter or if INPUT=INTERNAL is specified	Required if no value is specified for the INPUT= parameter or if INPUT=INTERNAL is specified	Required if no value is specified for the INPUT= parameter or if INPUT=INTERNAL is specified	None
DBID=	N/A	Optional	N/A	None
DYNMPIDT=	Required	Required	N/A	BLGYDYN
LOGGING=	Optional	Optional	Optional	N
ARG=	Optional	Optional	Optional	None
CHILD=	Optional	Optional	Optional	None
PARENT=	Optional	Optional	Optional	None
CHDPIDT=	Optional	Optional	Optional	None
USERRELN=	N/A	Optional	N/A	N
HITS=	Required if no value is specified for the INPUT= parameter or if INPUT=INTERNAL is specified	Required if no value is specified for the INPUT= parameter or if INPUT=INTERNAL is specified	Required if no value is specified for the INPUT= parameter or if INPUT=INTERNAL is specified	Defaults to 00000001 if no value is specified for the INPUT= parameter or if INPUT=INTERNAL is specified
CHECKIO=	Optional	Optional	Optional	Y
COMPRESS=	Optional	Optional	N/A	N
DELENTRY=	Optional	Optional	N/A	None
UNCOGNIZE=	Optional	Optional	N/A	None
NEWRNID=	Optional	Optional	N/A	N
TEXT=	Optional	Optional	Optional	Y
TEXTXFER=	Optional	Optional	Optional	D

TEXTAUD=	Optional	Optional	Optional	Y
TEXTCNT=	Optional	Optional	Optional	00000200
TEXTWID=	Optional	Optional	Optional	00000060
TEXTAREA=	Optional	Optional	Optional	B
TIMEOUT=	Optional	Optional	Optional	00000300
HISTORY=	Optional	Optional	Optional	Y

BLGINREC Sample

The Archiver input data set expects the following conditions:

- The first line beginning in column 1 should contain the word ARCHIVE or COPY or PURGE matching the FUNCTION= value in the data set named in the BLGARCP DD statement.
- Each RNID should begin in column 1 with only one RNID per line.
- All numeric RNIDs should have leading zeroes.

This data set is read and processed sequentially.

This sample BLGINREC could be used when INPUT=EXTERNAL in BLGARCP DD to archive only records 00000001, 00000002, and 00000003 (regardless of record type) into the archive database. Once these records are archived, they are deleted from the live database.

```
ARCHIVE                    * (Must be ARCHIVE or COPY or PURGE)
00000001                   * (Only one record per line)
00000002                   * (Numeric records must have leading zeroes)
00000003                   *
```

Figure 20. Sample BLGINREC

BLGARCP Samples

The Archiver parameter data set expects the following conditions:

- Archiver parameter specifications begin in column 1
- Comments, if specified, begin in column 40
- Each line contains a parameter; blank lines are not allowed

This data set is read and processed sequentially.

Sample 1

This sample BLGARCP archives up to 1000 Problem records with a status of CLOSED that were entered in 1997 from the “live” database defined in session BLGSES00 to the “archive” database defined in session BLGSESAR. In this example, the records are copied and compressed to the database defined in BLGSESAR and deleted from the database defined in BLGSES00.

```
FUNCTION=ARCHIVE
PURGE=NORMAL
INPUT=INTERNAL
CLASS=SYSADMIN
LIVESESS=BLGSES00
ARCSESS=BLGSESAR
APPLID=ARCHIVER
INQPIDT=BLGYPRI
DYNMPIDT=BLGYDYN
LOGGING=Y
ARG=STAC/CLOSED
ARG=DATE/1997/01/01
ARG=-DATE/1997/12/31
HITS=00001000
CHECKIO=N                              /* Default is Y
COMPRESS=Y                             /* Default is N
TEXT=Y
TEXTAUD=Y
TEXTXFER=B                             /* Default is D
TEXTCNT=00001000                       /* Default is 200
TEXTWID=00000132                       /* Default is 60
TEXTAREA=B
HISTORY=Y
```

Figure 21. Sample BLGARCP

Sample 2

This sample BLGARCP copies the records with RNID/00063542 and RNID/00064967 from the database defined as database 4 in session BLGSES00 to the database defined as database 5 in session BLGSES00. It is assumed that RNID/0003542 and RNID/00064967 do not exist in database 5. Before the records are created on database 5, the records are compressed, and the Date and Time Occurred data values are removed from the records.

```
FUNCTION=COPY
INPUT=INTERNAL
CLASS=SYSADMIN
LIVESESS=BLGSES00
ARCSESS=BLGSES00
DBID=4                                 /* Records read from DB 4
APPLID=ARCHIVER
INQPIDT=BLGYPRI
DYNMPIDT=BLGYDYN
LOGGING=Y
ARG=RNID/00063542                      /* Search for two specific records
ARG=|RNID/00064967
HITS=00001000
CHECKIO=N                              /* Default is Y
COMPRESS=Y                             /* Default is N
DELENTRY=S0C3D                         /* Delete Date Occurred
DELENTRY=S0C6A                         /* Delete Time Occurred
TEXT=Y
TEXTAUD=Y
TEXTXFER=B                             /* Default is D
TEXTCNT=00001000                       /* Default is 200
TEXTWID=00000132                       /* Default is 60
TEXTAREA=B
HISTORY=Y
```

Figure 22. Sample BLGARCP

Sample 3

This sample BLGARCP copies the record with RNID/00051368 from the database defined as database 5 in session BLGSES00 to the database defined as database 5 in session BLGSES00 (that is, the same database). When the record is created on database 5, it will be compressed and assigned the next available system-assigned RNID.

```
FUNCTION=COPY
INPUT=INTERNAL
CLASS=SYSADMIN
LIVESESS=BLGSES00                      /* Read from and write to DB5
ARCSESS=BLGSES00
APPLID=ARCHIVER
INQPIDT=BLGYPRI
DYNMPIDT=BLGYDYN
LOGGING=Y
ARG=RNID/00051368                      /* Copy 1 record
HITS=00001000
CHECKIO=N                              /* Default is Y
COMPRESS=Y                             /* Default is N
NEWRNID=Y                              /* RNID of record is replaced
TEXT=Y
TEXTAUD=Y
TEXTXFER=B                             /* Default is D
TEXTCNT=00001000                       /* Default is 200
TEXTWID=00000132                       /* Default is 60
TEXTAREA=B
HISTORY=Y
```

Figure 23. Sample BLGARCP

Sample 4

This sample BLGARCP deletes the records specified in the BLGINREC DD in the BLGARCJCL from database 5 in session BLGSES00.

```
FUNCTION=PURGE
INPUT=EXTERNAL
CLASS=SYSADMIN
LIVESESS=BLGSES00
APPLID=ARCHIVER
LOGGING=N
CHECKIO=N                              /* Default is Y
```

Figure 24. Sample BLGARCP

```
PURGE
00051368
00061854
00021537
00021336
```

Figure 25. Sample BLGINREC

Sample 5

This sample BLGARCP copies the record specified in the BLGINREC DD in the BLGARC JCL from database 4 in session BLGSES00 to database 5 in session BLGSES00 8 times. Each time the record is created on database 5, it is compressed and assigned the next available system-assigned RNID. Freeform text and history data are not copied.

```
FUNCTION=COPY
INPUT=EXTERNAL                        /* BLGINREC has multiple entries
CLASS=SYSADMIN
LIVESESS=BLGSES00
ARCSESS=BLGSES00
DBID=4
APPLID=ARCHIVER
DYNMPIDT=BLGYDYN
LOGGING=Y
CHECKIO=N                             /* Default is Y
COMPRESS=Y                            /* Default is N
NEWRNID=Y                             /* RNID of record is replaced
TEXT=N
TEXTXFER=B
HISTORY=N
```

Figure 26. Sample BLGARCP

```
COPY
00051368
00051368
00051368
00051368
00051368
00051368
00051368
00051368
```

Figure 27. Sample BLGINREC

For samples of Tivoli-defined and User-defined relationship processing, refer to “Parent/Child Relationships”.

Parent/Child Relationships

Tivoli-defined Relationships

This sample BLGARCP copies the first 5000 Change records and their associated Activity records that were created in 1997 from database 5 in BLGSES00 to database 5 in BLGSESAR. After the records are successfully copied, they are deleted from database 5 in session BLGSES00. This technique only works for Change (parent) and Activity (child) records which were created with the Tivoli-supplied program exits or which adhere to the following rules:

- The Change records must contain an s-word with associated-data indicating that Activity records exist. For example, Change records created with the Tivoli-supplied program exits contain the s-word index X'0CBC' with p-word NAMA/ which point to a unique Activity name.

- The Activity records must contain a 5-character p-word (including the slash) with the RNID of the Change record. For example, Activity records created with the Tivoli-supplied program exits contain the p-word RNOR/.

```
FUNCTION=ARCHIVE
PURGE=NORMAL
INPUT=INTERNAL
CLASS=SYSADMIN
LIVESESS=BLGSES00
ARCSESS=BLGSESAR
APPLID=ARCHIVER
INQPIDT=BLGYCHI
DYNMPIDT=BLGYDYN
LOGGING=N
ARG=STAC/CLOSED
ARG=DATE/1997/01/01
ARG=-DATE/1997/12/31
CHILD=RNOR/
PARENT=S0CBC
CHDPIDT=BLGYACI
HITS=00005000
CHECKIO=N
TEXT=Y
TEXTAUD=Y
TEXTXFER=B
TEXTCNT=00001000
TEXTWID=00000132
TEXTAREA=B
HISTORY=Y
```

Figure 28. Sample BLGARCP

User-defined Relationships

The sample JCL shown on page 353 shows you how to specify the BLGARCP parameters and the BLGINREC RNIDs in-stream. Records with user-defined relationships which are specified by the BLGINREC DD are copied from database 5 in session BLGSES00 to database 5 in BLGSESAR, and the BLGOPREC data set is created in the COPY step of the JCL. All of the RNIDs that are related to each other must be listed. That is, for every parent specified, all of the children must also be specified.

As the records are copied, the BLGOPREC data set is updated with the RNIDs of the records that were created successfully on the archive database. Then, in the DELETE step of the JCL, the BLGOPREC data set created in the COPY step is specified as the BLGINREC data set, and the RNIDs specified in the BLGINREC data set are deleted.

You must enable CHECKIO processing to prevent records from being updated between the COPY and the DELETE steps. Therefore, the records will remain checked out on database 5 in session BLGSES00 until they are deleted in the DELETE step.

The following sample JCL to invoke the Archiver for user-defined relationships can be found in the SBLMSAMP data set as member BLGUSERJ.

```
//BLGUSER JOB
//*                                                                   *
//* This is a sample jobstream for running the archiver, BLGARC, to   *
//* archive records with user-defined relationships.                  *
//* In the COPY step, the records specified by the BLGINREC DD are    *
//* copied from the live database to the archive data base.  Then,    *
//* in the DELETE step, the records specified in the BLGOPREC data set*
//* created by the COPY step are deleted from the live database.      *
//*                                                                   *
//*-------------------------------------------------------------------*
//COPY EXEC PGM=BLGARC,REGION=4M
//*
//* Do not forget to STEPLIB to your session load library if needed.
//*
//STEPLIB DD DSN=BLM.SBLMMOD1,DISP=SHR
//*
//* This data set will be used as input to the DELETE step of this job.
//*
//BLGOPREC DD DSN=????????.BLGOPREC,DISP=OLD
//*
//* RFTDD DD has your PIDT executable tables
//*
//RFTDD    DD DSN=BLM.SBLMFMT,DISP=SHR
//APIPRINT DD SYSOUT=*
//BLGOUT   DD SYSOUT=*
//SYSMDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//BLGARCP  DD *
FUNCTION=COPY
INPUT=EXTERNAL
CLASS=SYSADMIN
LIVESESS=BLGSES00
ARCSESS=BLGSESAR
APPLID=ARCHIVER
DYNMPIDT=BLGYDYN
LOGGING=N
USERRELN=Y
CHECKIO=Y
TEXT=Y
TEXTAUD=Y
TEXTXFER=B
TEXTCNT=00001000
TEXTWID=00000132
TEXTAREA=B
HISTORY=Y
/*
//* The BLGINREC DD specifies the RNIDs of the records to be copied.
//*
//BLGINREC DD *
COPY
PARENT1
CHILD11
CHILD12
PARENT2
CHILD21
CHILD22
PARENT3
CHILD31
PARENT4
/*
//*
//DELETE  EXEC PGM=BLGARC,REGION=4M
//*
//* Do not forget to STEPLIB to your session load library if needed.
//*
//STEPLIB DD DSN=BLM.SBLMMOD1,DISP=SHR
//*
//* This data set was created in the COPY step of this job by the
//* BLGOPREC DD.
//*
//BLGINREC DD DSN=????????.BLGOPREC,DISP=OLD
```

```
//*
//* RFTDD DD has your PIDT executable tables
//*
//RFTDD    DD DSN=BLM.SBLMFMT,DISP=SHR
//APIPRINT DD SYSOUT=*
//BLGOUT   DD SYSOUT=*
//SYSMDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//BLGARCP  DD *
FUNCTION=PURGE
INPUT=EXTERNAL
CLASS=SYSADMIN
LIVESESS=BLGSESAR
APPLID=ARCHIVER
LOGGING=N
/*
```

This is a sample BLGOPREC for user-defined relationships. Notice that PARENT4 was not created on the archive database successfully, so it is not in the BLGOPREC data set. Notice that the PURGE parameter is automatically specified on line 1 of the BLGOPREC file.

```
PURGE
PARENT1
CHILD11
CHILD12
PARENT2
CHILD21
CHILD22
PARENT3
CHILD31
```

Sample RFT

The RFT in Figure 29 works for both Tivoli-defined and user-defined relationships as long as the RNID of the Change (parent) record is contained in the Activity (child) records. For additional information on RFTs, refer to the *Tivoli Information Management for z/OS Data Reporting User's Guide*.

```
SECTION
   PUT COL(001) VALUE(COPY)
   SEARCH ARGUMENT(!S0B06 STAC/CLOSED DATE/1996/01/01 - DATE/1996/12/31)   +
      MERGE(N)
      /* RNID of the change record */
      PUT COL(001) DATA(RNID/.)
      SET NAME(CHANGE) DATA(RNID/.)
         SEARCH ARGUMENT(!S0B07 ¬!S0B06 RNOR/&CHANGE) MERGE(N)
            /* RNIDs of the activity records */
            PUT COL(001) DATA(RNID/.)
         ESEARCH
   ESEARCH
ESECTION
```

Figure 29. Sample RFT to produce the BLGINREC data set

Additional Tailoring Considerations

Performance Considerations

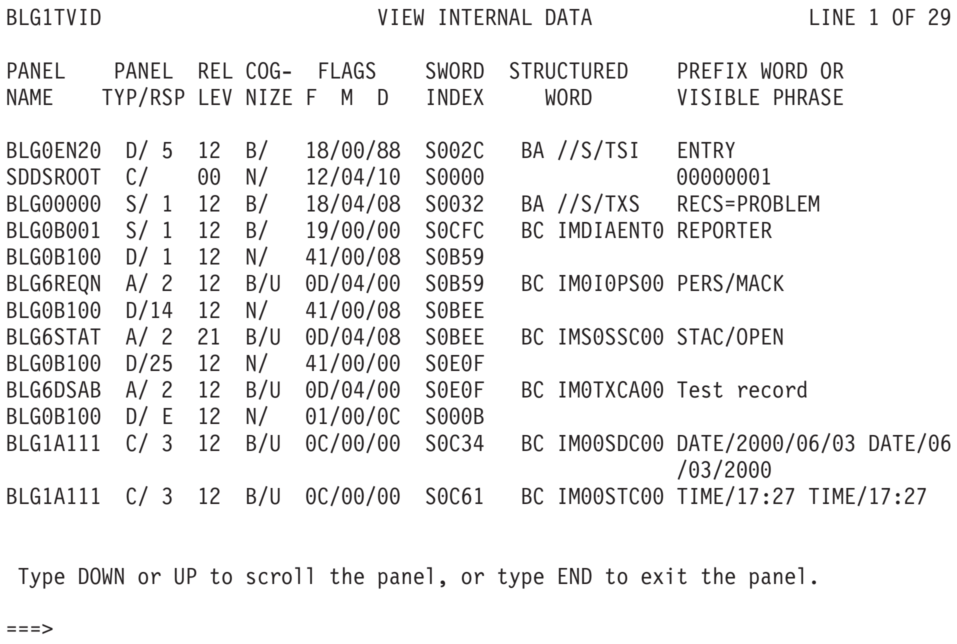
The following is a list of recommendations on how to improve the performance of the Archiver:

- Do not code STEPLIB DD statements in your JCL unless they are required. Place all Tivoli Information Management for z/OS modules and session members in the Linklist or Linkpack Area. Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for additional information. A STEPLIB DD statement causes all calls to load modules to access the STEPLIB DDNAME in an attempt to locate the modules, regardless of whether they reside in the data set(s).

- Disable CHECKIO processing unless USERRELN=Y or you specifically require records to be checked-out prior to retrieval. Coding CHECKIO=Y with FUNCTION=ARCHIVE or FUNCTION=COPY causes additional API transactions to be performed. When FUNCTION=COPY, each record will be checked out on the live database and then checked back in on the live database. When FUNCTION=ARCHIVE, each record will be checked out on the live database.
- When performing FUNCTION=ARCHIVE or FUNCTION=COPY, consider compressing the records by specifying COMPRESS=Y in the BLGARCP parameter data set. This removes redundant entries from each record and reduce the number of PIDT data entries to be processed by the API create transaction. In addition, DASD space is also saved on the archive database.
- When performing FUNCTION=ARCHIVE or FUNCTION=COPY, consider removing unwanted, not valid or redundant data entries by utilizing the DELENTRY parameter in the BLGARCP parameter data set. This reduces the number of PIDT entries to be processed by the API create transaction. In addition, if the entry deleted was cognized, updates to the SDIDS are also eliminated.
- When performing FUNCTION=ARCHIVE or FUNCTION=COPY, consider uncognizing fields that are no longer required as searchable by utilizing the UNCOGNIZE parameter in the BLGARCP parameter data set. This reduces the number of updates required to the SDIDS by the API create transactions.
- When performing FUNCTION=ARCHIVE or FUNCTION=COPY, specifying TEXTXFER=B eliminates the need for the LLAPI to allocate a data set for each text type in each record.
- If LLAPI logging is not required, specify LOGGING=N in the BLGARCP parameter data set. This eliminates the I/O to the APIPRINT data set caused during LLAPI logging.
- Consider defining separate Archiver sessions with different characteristics to those used by the normal interactive users. Refer to the COGENQ parameter in the BLGCLUST macro and the SORTPFX parameter in the BLGPARMS macro in the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference*.

Record Processing Considerations

In order to use the DELENTRY, UNCOGNIZE, and COMPRESS parameters, you must understand how the LLAPI processes the data in your records. When the LLAPI retrieves a record from your live database, a dynamic program interface data table (PIDT) is created. Each PIDT consists of a header and multiple data rows which correspond to the data in your records. For the sake of simplicity, you can use the output from the VIEW INTERNALS command to understand how the PIDT data rows are loaded. Essentially, the lines of output from the VIEW INTERNALS command are loaded line by line from the top to the bottom into the PIDT data rows, and the Archiver assigns an entry number to each row which corresponds to the VIEW INTERNALS line number. The SDDSROOT entry in the VIEW INTERNALS output is not processed, and freeform text entries (1 per freeform text type) are processed last. Refer to Figure 30 on page 357 for a sample of the output produced by the VIEW INTERNALS command.



```
BLG1TVID                     VIEW INTERNAL DATA                 LINE 1 OF 29

PANEL    PANEL  REL COG-  FLAGS    SWORD  STRUCTURED    PREFIX WORD OR
NAME    TYP/RSP LEV NIZE F  M  D   INDEX     WORD       VISIBLE PHRASE

BLG0EN20  D/ 5  12  B/   18/00/88  S002C   BA //S/TSI   ENTRY
SDDSROOT  C/    00  N/   12/04/10  S0000                00000001
BLG00000  S/ 1  12  B/   18/04/08  S0032   BA //S/TXS   RECS=PROBLEM
BLG0B001  S/ 1  12  B/   19/00/00  S0CFC   BC IMDIAENT0 REPORTER
BLG0B100  D/ 1  12  N/   41/00/08  S0B59
BLG6REQN  A/ 2  12  B/U  0D/04/00  S0B59   BC IM0I0PS00 PERS/MACK
BLG0B100  D/14  12  N/   41/00/08  S0BEE
BLG6STAT  A/ 2  21  B/U  0D/04/08  S0BEE   BC IMS0SSC00 STAC/OPEN
BLG0B100  D/25  12  N/   41/00/00  S0E0F
BLG6DSAB  A/ 2  12  B/U  0D/04/00  S0E0F   BC IM0TXCA00 Test record
BLG0B100  D/ E  12  N/   01/00/0C  S000B
BLG1A111  C/ 3  12  B/U  0C/00/00  S0C34   BC IM00SDC00 DATE/2000/06/03 DATE/06
                                                        /03/2000
BLG1A111  C/ 3  12  B/U  0C/00/00  S0C61   BC IM00STC00 TIME/17:27 TIME/17:27


 Type DOWN or UP to scroll the panel, or type END to exit the panel.

===>
```

Figure 30. VIEW INTERNALS Output

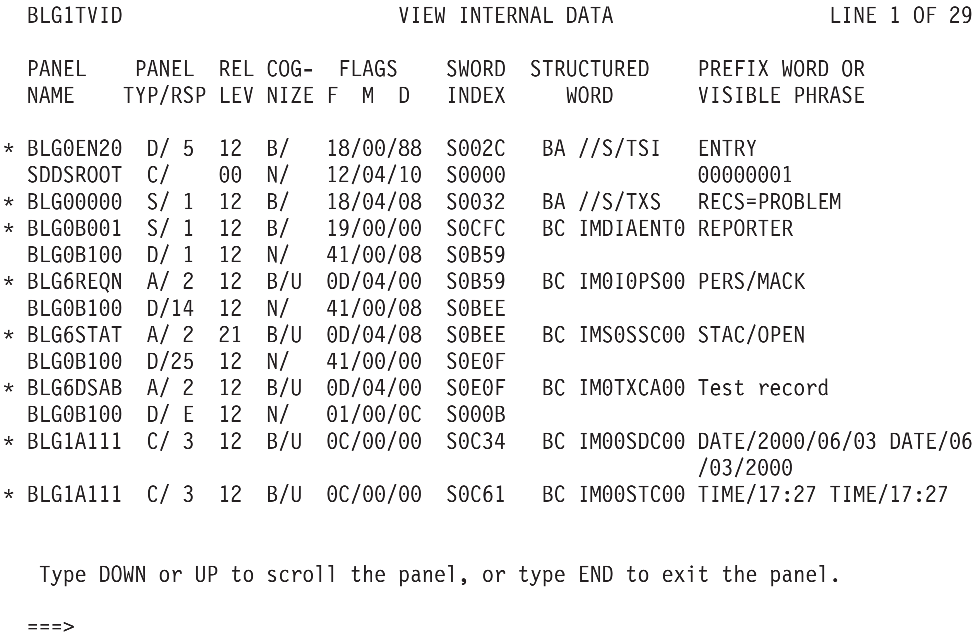
DELENTRY

In order to delete an entry in a record, the s-word index associated with the DELENTRY parameter in the BLGARCP data set must match the s-word index specified with the data in the VIEW INTERNALS output for the record.

When choosing which s-word index to specify, remember the following rules:

- DELENTRY will only delete entries that contain data or a visible phrase.
- DELENTRY cannot delete data for which the associated entry in the VIEW INTERNALS output has an s-word index of X'0000'.

In Figure 31 on page 358, the lines marked by an asterisk can be deleted using the DELENTRY parameter.



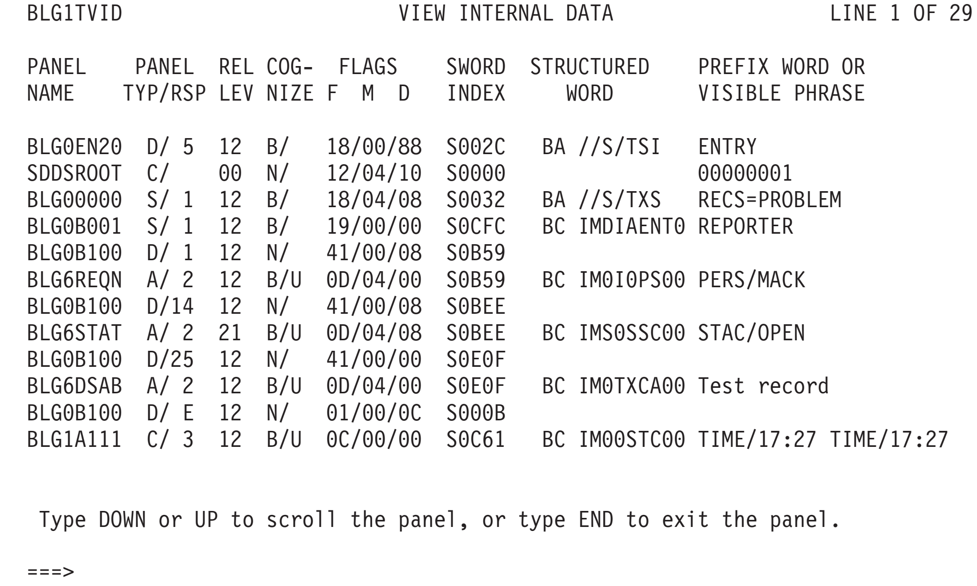
```
  BLG1TVID                      VIEW INTERNAL DATA                 LINE 1 OF 29

  PANEL    PANEL  REL COG-  FLAGS    SWORD  STRUCTURED    PREFIX WORD OR
  NAME    TYP/RSP LEV NIZE F  M  D   INDEX     WORD       VISIBLE PHRASE

* BLG0EN20  D/ 5  12  B/   18/00/88  S002C   BA //S/TSI   ENTRY
  SDDSROOT  C/    00  N/   12/04/10  S0000                00000001
* BLG00000  S/ 1  12  B/   18/04/08  S0032   BA //S/TXS   RECS=PROBLEM
* BLG0B001  S/ 1  12  B/   19/00/00  S0CFC   BC IMDIAENT0 REPORTER
  BLG0B100  D/ 1  12  N/   41/00/08  S0B59
* BLG6REQN  A/ 2  12  B/U  0D/04/00  S0B59   BC IM0I0PS00 PERS/MACK
  BLG0B100  D/14  12  N/   41/00/08  S0BEE
* BLG6STAT  A/ 2  21  B/U  0D/04/08  S0BEE   BC IMS0SSC00 STAC/OPEN
  BLG0B100  D/25  12  N/   41/00/00  S0E0F
* BLG6DSAB  A/ 2  12  B/U  0D/04/00  S0E0F   BC IM0TXCA00 Test record
  BLG0B100  D/ E  12  N/   01/00/0C  S000B
* BLG1A111  C/ 3  12  B/U  0C/00/00  S0C34   BC IM00SDC00 DATE/2000/06/03 DATE/06
                                                          /03/2000
* BLG1A111  C/ 3  12  B/U  0C/00/00  S0C61   BC IM00STC00 TIME/17:27 TIME/17:27


   Type DOWN or UP to scroll the panel, or type END to exit the panel.

  ===>
```

Figure 31. VIEW INTERNALS Output before DELENTRY

Thus, if you wanted to delete date occurred (DATE/2000/06/03), you could specify:

```
DELENTRY=S0C34
```

and then the VIEW INTERNALS data of the archived record would look like:



```
  BLG1TVID                      VIEW INTERNAL DATA                 LINE 1 OF 29

  PANEL    PANEL  REL COG-  FLAGS    SWORD  STRUCTURED    PREFIX WORD OR
  NAME    TYP/RSP LEV NIZE F  M  D   INDEX     WORD       VISIBLE PHRASE

  BLG0EN20  D/ 5  12  B/   18/00/88  S002C   BA //S/TSI   ENTRY
  SDDSROOT  C/    00  N/   12/04/10  S0000                00000001
  BLG00000  S/ 1  12  B/   18/04/08  S0032   BA //S/TXS   RECS=PROBLEM
  BLG0B001  S/ 1  12  B/   19/00/00  S0CFC   BC IMDIAENT0 REPORTER
  BLG0B100  D/ 1  12  N/   41/00/08  S0B59
  BLG6REQN  A/ 2  12  B/U  0D/04/00  S0B59   BC IM0I0PS00 PERS/MACK
  BLG0B100  D/14  12  N/   41/00/08  S0BEE
  BLG6STAT  A/ 2  21  B/U  0D/04/08  S0BEE   BC IMS0SSC00 STAC/OPEN
  BLG0B100  D/25  12  N/   41/00/00  S0E0F
  BLG6DSAB  A/ 2  12  B/U  0D/04/00  S0E0F   BC IM0TXCA00 Test record
  BLG0B100  D/ E  12  N/   01/00/0C  S000B
  BLG1A111  C/ 3  12  B/U  0C/00/00  S0C61   BC IM00STC00 TIME/17:27 TIME/17:27


   Type DOWN or UP to scroll the panel, or type END to exit the panel.

  ===>
```

Figure 32. VIEW INTERNAL Output after DELENTRY=S0C34

Notice that the line with s-word index X'0C3D' which contains no data is not deleted. For information on how to delete this line, refer to “COMPRESS” on page 361.

In the VIEW INTERNALS output in Figure 31, the data TYPE/USER cannot be deleted because the s-word index on the line with the data is X'0000'. Notice that the lines of data in the VIEW INTERNALS output are paired when the data is collected via an assisted-entry panel. That is, there is a line for the data-entry panel and for the assisted-entry panel for each piece of data. When the assisted-entry panel specifies an s-word, the VIEW INTERNALS line for the data will contain an s-word index other than X'0000'. However, when the assisted-entry panel specifies *collect from caller*, the s-word index associated with the data will be X'0000'. For additional information on how data is collected via an assisted-entry panel, refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide*.

Direct-add items (such as items added to a record with a Control panel ADD) only create 1 entry in the VIEW INTERNALS output.

UNCOGNIZE

In order to uncognize an entry in a record, the s-word index associated with the UNCOGNIZE parameter in the BLGARCP data set must match the s-word index specified with the data in the VIEW INTERNALS output for the record.

When UNCOGNIZE is specified, both the s-word and the p-word for the entry are uncognized.

When choosing which s-word index to specify, remember the following rules:

- UNCOGNIZE will only uncognize entries that contain data or a visible phrase.
- UNCOGNIZE cannot uncognize entries for which the associated entry in the VIEW INTERNALS output has an s-word index of X'0000'.

Notice that the VIEW INTERNALS output marks which lines are cognized with the COGNIZE column. If the COGNIZE column has N/, the line is not cognized, and there is no reason to specify UNCOGNIZE to uncognize it. If the line contains B/ or B/U, the line is cognized. If the line contains P/M, cognize only p-word is specified on the assisted-entry panel (additional information on these fields can be found in *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference*). In Figure 33 on page 360, the lines marked by an asterisk may be uncognized using the UNCOGNIZE parameter.

```
  BLG1TVID                      VIEW INTERNAL DATA                 LINE 1 OF 29

  PANEL    PANEL  REL COG-  FLAGS    SWORD  STRUCTURED    PREFIX WORD OR
  NAME    TYP/RSP LEV NIZE F  M  D   INDEX     WORD       VISIBLE PHRASE

* BLG0EN20  D/ 5  12  B/   18/00/88  S002C   BA //S/TSI   ENTRY
  SDDSROOT  C/    00  N/   12/04/10  S0000                00000001
* BLG00000  S/ 1  12  B/   18/04/08  S0032   BA //S/TXS   RECS=PROBLEM
* BLG0B001  S/ 1  12  B/   19/00/00  S0CFC   BC IMDIAENT0 REPORTER
  BLG0B100  D/ 1  12  N/   41/00/08  S0B59
* BLG6REQN  A/ 2  12  B/U  0D/04/00  S0B59   BC IM0I0PS00 PERS/MACK
  BLG0B100  D/14  12  N/   41/00/08  S0BEE
* BLG6STAT  A/ 2  21  B/U  0D/04/08  S0BEE   BC IMS0SSC00 STAC/CLOSED
  BLG0B100  D/25  12  N/   41/00/00  S0E0F
* BLG6DSAB  A/ 2  12  B/U  0D/04/00  S0E0F   BC IM0TXCA00 Test record
  BLG0B100  D/ E  12  N/   01/00/0C  S000B
* BLG1A111  C/ 3  12  B/U  0C/00/00  S0C34   BC IM00SDC00 DATE/2000/06/03 DATE/06
                                                          /03/2000
* BLG1A111  C/ 3  12  B/U  0C/00/00  S0C61   BC IM00STC00 TIME/17:27 TIME/17:27


   Type DOWN or UP to scroll the panel, or type END to exit the panel.

  ===>
```

Figure 33. VIEW INTERNALS Output before UNCOGNIZE

Thus, if you wanted to uncognize the status (STAC/CLOSED) because all of the records on your archive database have a status of closed, you could specify:

```
UNCOGNIZE=S0BEE
```

and then the VIEW INTERNALS data of the archived record would look like:

```
 BLG1TVID                      VIEW INTERNAL DATA                 LINE 1 OF 29

 PANEL    PANEL  REL COG-  FLAGS    SWORD  STRUCTURED    PREFIX WORD OR
 NAME    TYP/RSP LEV NIZE F  M  D   INDEX     WORD       VISIBLE PHRASE

 BLG0EN20  D/ 5  12  B/   18/00/88  S002C   BA //S/TSI   ENTRY
 SDDSROOT  C/    00  N/   12/04/10  S0000                00000001
 BLG00000  S/ 1  12  B/   18/04/08  S0032   BA //S/TXS   RECS=PROBLEM
 BLG0B001  S/ 1  12  B/   19/00/00  S0CFC   BC IMDIAENT0 REPORTER
 BLG0B100  D/ 1  12  N/   41/00/08  S0B59
 BLG6REQN  A/ 2  12  B/U  0D/04/00  S0B59   BC IM0I0PS00 PERS/MACK
 BLG0B100  D/14  12  N/   41/00/08  S0BEE
 BLG6STAT  A/ 2  21  B/U  0D/04/08  S0BEE   BC IMS0SSC00 STAC/CLOSED
 BLG0B100  D/25  12  N/   41/00/00  S0E0F
 BLG6DSAB  A/ 2  12  B/U  0D/04/00  S0E0F   BC IM0TXCA00 Test record
 BLG0B100  D/ E  12  N/   01/00/0C  S000B
 BLG1A111  C/ 3  12  B/U  0C/00/00  S0C34   BC IM00SDC00 DATE/2000/06/03 DATE/06
                                                         /03/2000
 BLG1A111  C/ 3  12  B/U  0C/00/00  S0C61   BC IM00STC00 TIME/17:27 TIME/17:27


  Type DOWN or UP to scroll the panel, or type END to exit the panel.

 ===>
```

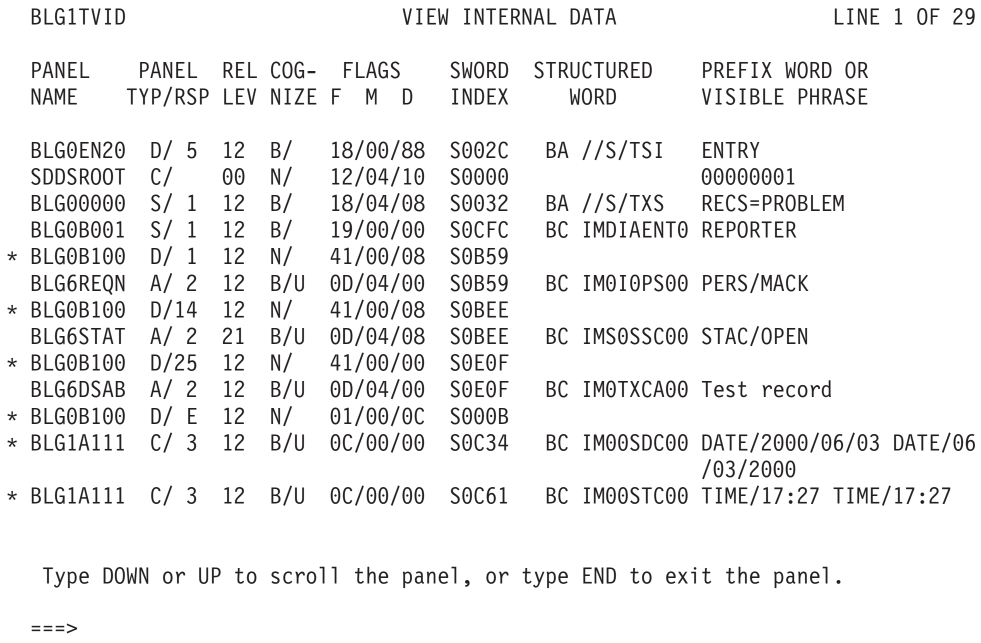
Figure 34. VIEW INTERNAL Output after UNCOGNIZE=S0BEE

COMPRESS

In order to remove non-essential entries from a record, specify the COMPRESS=Y parameter in the BLGARCP data set. The following rules explain which entries will be removed with COMPRESS=Y:

- COMPRESS will remove all entries that do not contain data even if they contain an s-word index.
- COMPRESS will not remove entries that contain data or a visible phrase.

In Figure 35, the lines marked by an asterisk may be compressed using the COMPRESS parameter.



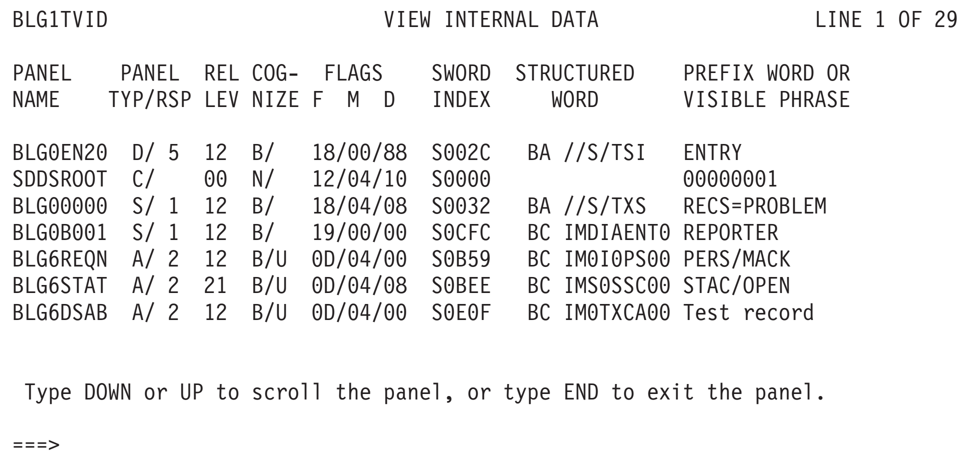
```
  BLG1TVID                      VIEW INTERNAL DATA                 LINE 1 OF 29

  PANEL    PANEL  REL COG-  FLAGS    SWORD  STRUCTURED    PREFIX WORD OR
  NAME    TYP/RSP LEV NIZE F  M  D   INDEX     WORD       VISIBLE PHRASE

  BLG0EN20  D/ 5  12  B/   18/00/88  S002C   BA //S/TSI   ENTRY
  SDDSROOT  C/    00  N/   12/04/10  S0000                00000001
  BLG00000  S/ 1  12  B/   18/04/08  S0032   BA //S/TXS   RECS=PROBLEM
  BLG0B001  S/ 1  12  B/   19/00/00  S0CFC   BC IMDIAENT0 REPORTER
* BLG0B100  D/ 1  12  N/   41/00/08  S0B59
  BLG6REQN  A/ 2  12  B/U  0D/04/00  S0B59   BC IM0I0PS00 PERS/MACK
* BLG0B100  D/14  12  N/   41/00/08  S0BEE
  BLG6STAT  A/ 2  21  B/U  0D/04/08  S0BEE   BC IMS0SSC00 STAC/OPEN
* BLG0B100  D/25  12  N/   41/00/00  S0E0F
  BLG6DSAB  A/ 2  12  B/U  0D/04/00  S0E0F   BC IM0TXCA00 Test record
* BLG0B100  D/ E  12  N/   01/00/0C  S000B
* BLG1A111  C/ 3  12  B/U  0C/00/00  S0C34   BC IM00SDC00 DATE/2000/06/03 DATE/06
                                                          /03/2000
* BLG1A111  C/ 3  12  B/U  0C/00/00  S0C61   BC IM00STC00 TIME/17:27 TIME/17:27


   Type DOWN or UP to scroll the panel, or type END to exit the panel.

  ===>
```

Figure 35. VIEW INTERNALS Output before COMPRESS

Thus, if you specify:

```
COMPRESS=Y
```

then the VIEW INTERNALS data of the archived record would look like:



```
BLG1TVID                      VIEW INTERNAL DATA                 LINE 1 OF 29

PANEL    PANEL  REL COG-  FLAGS    SWORD  STRUCTURED    PREFIX WORD OR
NAME    TYP/RSP LEV NIZE F  M  D   INDEX     WORD       VISIBLE PHRASE

BLG0EN20  D/ 5  12  B/   18/00/88  S002C   BA //S/TSI   ENTRY
SDDSROOT  C/    00  N/   12/04/10  S0000                00000001
BLG00000  S/ 1  12  B/   18/04/08  S0032   BA //S/TXS   RECS=PROBLEM
BLG0B001  S/ 1  12  B/   19/00/00  S0CFC   BC IMDIAENT0 REPORTER
BLG6REQN  A/ 2  12  B/U  0D/04/00  S0B59   BC IM0I0PS00 PERS/MACK
BLG6STAT  A/ 2  21  B/U  0D/04/08  S0BEE   BC IMS0SSC00 STAC/OPEN
BLG6DSAB  A/ 2  12  B/U  0D/04/00  S0E0F   BC IM0TXCA00 Test record


 Type DOWN or UP to scroll the panel, or type END to exit the panel.

===>
```

Figure 36. VIEW INTERNALS Output after COMPRESS=Y

Panel Processing Considerations

If you are planning to use the Archiver, you should also consider the following:

- You need to modify API create/update TSPs to enable special processing:
  - BLGAPI02 - Delete Line 16 which says BRANCH DISABLED.
  - BLGAPI05 - Delete Line 17 which says BRANCH DISABLED.
  - BLGAPI10 - Delete Line 9 which says BRANCH DISABLED.

  This should be done in the both the LIVESESS and ARCSESS sessions.
- The Archiver uses API "panel processing," which means that the archived records are filed using file processing of the data-summary panels which were created. In some cases, the Archiver can use incorrect Create Summary panels when archiving "child"-type records (for example, change activity records). To avoid this situation, you can direct the API to use default file processing for archived records. As shipped, this default file processing performs only a record file and does not add information such as date/time last altered. To enable this for your archived records, do the following:
  - Identify the record types that you will be archiving.
  - For each record type that you want to file using the default file, modify the record's entry in control panel BLG1AACP to target summary panel BLG0DFSM. BLG1AACP contains a list of record type s-words and associated summary panels to use for filing the record. For child records, make sure that both the parent and child s-word rows target BLG0DFSM. Make sure also that the modified BLG1AACP is available to the ARCSESS session.
- If you do not choose to implement the default file processing panels, you must ensure that notification processing is disabled for the Archiver. If you use TSPs and TSO Send for notification, ensure that user exits are in place for the problem and/or change record file panels to prevent TSPs from running during API processing. If you use TSXs for e-mail notification, disable the calls to the TSXs in copies of the record file panels used by the API.
  - BLG1A111 - Problem Record File
  - BLG1A121 - Change Record File.

Do this in the ARCSESS session.

- Only one record type can be archived/copied at a time. Use one of the following values for INQPIDT in the input parameters list:
  BLGYACI - change activity records
  BLGYCHI - change records
  BLGYDCI - data center records
  BLGYHCI - configuration hardware component records
  BLGYHFI - configuration hardware component feature records
  BLGYHNI - configuration financial hardware records
  BLGYHSI - configuration hardware sub-component records
  BLGYHXI - configuration hardware component connection
  BLGYPRI - problem records
  BLGYSCI - configuration software component records
  BLGYSFI - configuration software component feature records
  BLGYSNI - configuration financial software records
  BLGYSVI - configuration service records
  BLGYSXI - configuration software component connection records
  BLGYSYI - configuration system records

  If you created your own record types you will need to create a unique inquiry PIDT to accommodate your records. Use the samples in SBLMSAMP as a model.

- Because the Archiver is a LLAPI job, it will be affected by the setting of the APISECURITY keyword in the BLX-SP parameters member. The APISECURITY keyword specifies whether to activate the BLX-SP security checking for APIs to validate that the MVS application user ID is allowed to access a Tivoli Information Management for z/OS database. For more information on this keyword, refer to the *Tivoli Information Management for z/OS Application Program Interface Guide*.

Logical Database Partition Considerations

If you are using logical database partitioning, you must archive each separate logical partition of a database separately using the Archiver. For each partition, the archive session must use a privilege class whose primary partition is the partition being archived. This ensures that the archived records are placed into the correct partition and that duplicate records will not be produced in the archive database if the same record ID is archived more than once.

A

Panel Lists

This appendix lists the data entry panels that are shipped with Tivoli Information Management for z/OS Version 7.1. “Tivoli Information Management for z/OS Panels Arranged by Panel ID” lists the panels in alphanumeric order by panel identifier. “Tivoli Information Management for z/OS Panels Arranged by Panel Title” on page 399 lists the panels in alphabetic order by panel title.

Tivoli Information Management for z/OS Panels Arranged by Panel ID

BLGESC14	OLD ESCALATION TSP
BLGESC16	OLD ESCALATION TSP
BLGLABND	ABEND CODE ENTRY
BLGLALIS	ALIAS NAME ENTRY
BLGLALSD	ALIAS NAME DISPLAY
BLGLAPNU	APAR NUMBER ENTRY
BLGLAPN1	APAR NUMBER ENTRY
BLGLAPN2	APAR NUMBER ENTRY
BLGLAPN3	APAR NUMBER ENTRY
BLGLAPN4	APAR NUMBER ENTRY
BLGLAPN5	APAR NUMBER ENTRY
BLGLAPST	DISPLAY CHANGE APPROVER STATUS
BLGLAPVR	EDIT CHANGE APPROVER
BLGLBONU	BILL OF MATERIAL ENTRY
BLGLBON1	BILL OF MATERIAL ENTRY
BLGLBON2	BILL OF MATERIAL ENTRY
BLGLBON3	BILL OF MATERIAL ENTRY
BLGLBON4	BILL OF MATERIAL ENTRY
BLGLBON5	BILL OF MATERIAL ENTRY
BLGLCMDD	Command Alias/Authorization Display
BLGLCMDU	Command Alias/Authorization Entry
BLGLCNAM	GENERIC CARD ENTRY

BLGLCNA1	GENERIC CARD ENTRY
BLGLCNA2	GENERIC CARD ENTRY
BLGLCNA3	GENERIC CARD ENTRY
BLGLCNA4	GENERIC CARD ENTRY
BLGLCNA5	GENERIC CARD ENTRY
BLGLCRDN	CARD NAME ENTRY
BLGLDATT	DATA ATTRIBUTE RECORDS ENTRY
BLGLDEVF	DEVICE TYPE ENTRY
BLGLDEVL	DEVICE NAME ENTRY
BLGLDFEA	FEATURE NAME ENTRY
BLGLDMSN	SERIAL NUMBER ENTRY
BLGLDMS1	SERIAL NUMBER ENTRY
BLGLDMS2	SERIAL NUMBER ENTRY
BLGLDMS3	SERIAL NUMBER ENTRY
BLGLDMS4	SERIAL NUMBER ENTRY
BLGLDMS5	SERIAL NUMBER ENTRY
BLGLDNUM	DOCUMENT NUMBER ENTRY
BLGLDNU1	DOCUMENT NUMBER ENTRY
BLGLDNU2	DOCUMENT NUMBER ENTRY
BLGLDOCN	DOCUMENT NAME ENTRY
BLGLDOC1	DOCUMENT NAME ENTRY
BLGLDOC2	DOCUMENT NAME ENTRY
BLGLDSSD	TIME ZONE DISPLAY
BLGLDSTS	UNIVERSAL TIME SUPPORT
BLGLDTMD	UNIVERSAL TIME SUPPORT
BLGLDTTM	UNIVERSAL TIME SUPPORT
BLGLDTYP	DEVICE TYPE ENTRY
BLGLDVNA	DEVICE NAME ENTRY
BLGLDVN1	DEVICE NAME ENTRY
BLGLDVN2	DEVICE NAME ENTRY
BLGLDVN3	DEVICE NAME ENTRY
BLGLDVN4	DEVICE NAME ENTRY
BLGLDVN5	DEVICE NAME ENTRY
BLGLDVSR	SERIAL NUMBER ENTRY
BLGLECID	ENGINEER CHANGE ENTRY

BLGLECI1	ENGINEER CHANGE ENTRY
BLGLECI2	ENGINEER CHANGE ENTRY
BLGLECI3	ENGINEER CHANGE ENTRY
BLGLECI4	ENGINEER CHANGE ENTRY
BLGLECI5	ENGINEER CHANGE ENTRY
BLGLECLV	HARDWARE EC LEVEL ENTRY
BLGLECNU	ENGINEER ANNOUNCEMENT ENTRY
BLGLECN1	ENGINEER ANNOUNCEMENT ENTRY
BLGLECN2	ENGINEER ANNOUNCEMENT ENTRY
BLGLECN3	ENGINEER ANNOUNCEMENT ENTRY
BLGLECN4	ENGINEER ANNOUNCEMENT ENTRY
BLGLECN5	ENGINEER ANNOUNCEMENT ENTRY
BLGLENVF	ENVIRONMENT CONDITION ENTRY
BLGLENVS	ENVIRONMENT CONDITION ENTRY
BLGLERCE	ERROR CODE ENTRY
BLGLFEAT	OPTIONAL FEATURE ENTRY
BLGLFEA1	OPTIONAL FEATURE ENTRY
BLGLFEA2	OPTIONAL FEATURE ENTRY
BLGLFEA3	OPTIONAL FEATURE ENTRY
BLGLFEA4	OPTIONAL FEATURE ENTRY
BLGLFEA5	OPTIONAL FEATURE ENTRY
BLGLFGFD	FIELD GROUP FIELD LIST DISPLAY
BLGLFGFK	FIELD GROUP FIELD LIST ENTRY
BLGLFGFL	FIELD GROUP FIELD LIST ENTRY
BLGLFGID	DESKTOP FIELD GROUP DISPLAY
BLGLFGIN	DESKTOP FIELD GROUP ENTRY
BLGLFLDF	SOFTWARE FIELD NAME ENTRY
BLGLFMIE	FMID ENTRY
BLGLFNAM	FILES AFFECTED ENTRY
BLGLFNA1	FILES AFFECTED ENTRY
BLGLFNA2	FILES AFFECTED ENTRY
BLGLFNA3	FILES AFFECTED ENTRY
BLGLFNA4	FILES AFFECTED ENTRY
BLGLFNA5	FILES AFFECTED ENTRY
BLGLINUM	ITEM NUMBER ENTRY

BLGLIPID	DEVICE ITEM ENTRY
BLGLIPI1	DEVICE ITEM ENTRY
BLGLIPI2	DEVICE ITEM ENTRY
BLGLIPI3	DEVICE ITEM ENTRY
BLGLIPI4	DEVICE ITEM ENTRY
BLGLIPI5	DEVICE ITEM ENTRY
BLGLJ300	ELIGIBLE USERS
BLGLJHPL	SHARING LOGICAL PARTITION IDENTIFIER ENTRY
BLGLJSPL	AUTHORIZED LOGICAL PARTITION IDENTIFIER ENTRY
BLGLLCRM	LINE/CIRCUIT NUMBER ENTRY
BLGLLIDE	LOOP ID ENTRY
BLGLLOCO	LOCATION CODE ENTRY
BLGLLOC1	LOCATION CODE ENTRY
BLGLLOC2	LOCATION CODE ENTRY
BLGLLOC3	LOCATION CODE ENTRY
BLGLLOC4	LOCATION CODE ENTRY
BLGLLOC5	LOCATION CODE ENTRY
BLGLLSPE	LINE SPEED ENTRY
BLGLLTYE	LINE/LOOP TYPE ENTRY
BLGLLVLF	SOFTWARE LEVEL ENTRY
BLGLLVLP	SOFTWARE LEVEL ENTRY
BLGLMAUF	MAJOR UNIT ENTRY
BLGLMIUF	MINOR UNIT ENTRY
BLGLMNAM	AFFECTED DEVICE ENTRY
BLGLMNA1	AFFECTED DEVICE ENTRY
BLGLMNA2	AFFECTED DEVICE ENTRY
BLGLMNA3	AFFECTED DEVICE ENTRY
BLGLMNA4	AFFECTED DEVICE ENTRY
BLGLMNA5	AFFECTED DEVICE ENTRY
BLGLMODF	USER MODIFICATION ENTRY
BLGLMSGC	MESSAGE IDENTITY ENTRY
BLGLNPRO	PROCEDURE NAME ENTRY
BLGLNPR1	PROCEDURE NAME ENTRY
BLGLNPR2	PROCEDURE NAME ENTRY
BLGLNUPR	PROCEDURE NUMBER ENTRY

BLGLNUP1	PROCEDURE NUMBER ENTRY
BLGLNUP2	PROCEDURE NUMBER ENTRY
BLGLOPCF	OPERATION CODE ENTRY
BLGLOPCS	OPERATION CODE ENTRY
BLGLPCSF	SOFTWARE STATEMENT ENTRY
BLGLPIDF	PROGRAM IDENTITY ENTRY
BLGLPRID	PROGRAM IDENTITY ENTRY
BLGLPTFF	SOFTWARE PTF ENTRY
BLGLPTNU	SOFTWARE PTF ENTRY
BLGLPTN1	SOFTWARE PTF ENTRY
BLGLPTN2	SOFTWARE PTF ENTRY
BLGLPTN3	SOFTWARE PTF ENTRY
BLGLPTN4	SOFTWARE PTF ENTRY
BLGLPTN5	SOFTWARE PTF ENTRY
BLGLPUBF	PUBLICATION IDENTITY ENTRY
BLGLPUBS	PUBLICATION IDENTITY ENTRY
BLGLPZFF	SOFTWARE APAR ENTRY
BLGLQHPL	SHARING LOGICAL PARTITION IDENTIFIER DISPLAY
BLGLQSPL	AUTHORIZED LOGICAL PARTITION IDENTIFIER DISPLAY
BLGLQ300	ELIGIBLE USERS
BLGLREAN	ENGINEER ACTION ENTRY
BLGLREA1	ENGINEER ACTION ENTRY
BLGLREA2	ENGINEER ACTION ENTRY
BLGLREA3	ENGINEER ACTION ENTRY
BLGLREA4	ENGINEER ACTION ENTRY
BLGLREA5	ENGINEER ACTION ENTRY
BLGLREGF	SOFTWARE REGISTER ENTRY
BLGLREVD	DISPLAY CHANGE REVIEWER
BLGLREVR	EDIT CHANGE REVIEWER
BLGLRMID	SOFTWARE ROUTINE ENTRY
BLGLRMI1	SOFTWARE ROUTINE ENTRY
BLGLRMI2	SOFTWARE ROUTINE ENTRY
BLGLRMI3	SOFTWARE ROUTINE ENTRY
BLGLRMI4	SOFTWARE ROUTINE ENTRY
BLGLRMI5	SOFTWARE ROUTINE ENTRY

BLGLROUF	SOFTWARE ROUTINE NAME ENTRY
BLGLROUS	SOFTWARE ROUTINE NAME ENTRY
BLGLRPQN	PRICE QUOTATION ENTRY
BLGLRPQ1	PRICE QUOTATION ENTRY
BLGLRPQ2	PRICE QUOTATION ENTRY
BLGLRPQ3	PRICE QUOTATION ENTRY
BLGLRPQ4	PRICE QUOTATION ENTRY
BLGLRPQ5	PRICE QUOTATION ENTRY
BLGLSLEV	SOFTWARE LEVEL ENTRY
BLGLSLE1	SOFTWARE LEVEL ENTRY
BLGLSLE2	SOFTWARE LEVEL ENTRY
BLGLSLE3	SOFTWARE LEVEL ENTRY
BLGLSLE4	SOFTWARE LEVEL ENTRY
BLGLSLE5	SOFTWARE LEVEL ENTRY
BLGLSPID	SOFTWARE COMPONENT ENTRY
BLGLSPI1	SOFTWARE COMPONENT ENTRY
BLGLSPI2	SOFTWARE COMPONENT ENTRY
BLGLSPI3	SOFTWARE COMPONENT ENTRY
BLGLSPI4	SOFTWARE COMPONENT ENTRY
BLGLSPI5	SOFTWARE COMPONENT ENTRY
BLGLSQMP	RELATIONAL DATA MAPPING TABLE ENTRY
BLGLSTMT	SOFTWARE STATEMENT ENTRY
BLGLSUID	SUBSET NUMBERS ENTRY
BLGLSUI1	SUBSET NUMBERS ENTRY
BLGLSUI2	SUBSET NUMBERS ENTRY
BLGLSUI3	SUBSET NUMBERS ENTRY
BLGLSUI4	SUBSET NUMBERS ENTRY
BLGLSUI5	SUBSET NUMBERS ENTRY
BLGLSUPD	PRIVILEGE CLASS
BLGLSUPU	PRIVILEGE CLASS
BLGLSWCC	SOFTWARE COMPLETION CODE ENTRY
BLGLSWRC	SOFTWARE RETURN CODE ENTRY
BLGLSYMC	EXTERNAL SYMPTOM ENTRY
BLGLTSFD	DESKTOP PANEL FIELD LIST DISPLAY
BLGLTSFK	DESKTOP PANEL FIELD LIST ENTRY

	BLGLTSFL	DESKTOP PANEL FIELD LIST ENTRY
	BLGLTSID	DESKTOP PANEL LAYOUT DISPLAY
	BLGLTSIN	DESKTOP PANEL LAYOUT DEFINITION
|	**BLGLTSTD**	DESKTOP LAYOUT TABLE SUPPORT
|	**BLGLTSTK**	DESKTOP LAYOUT TABLE SUPPORT
|	**BLGLTSTL**	DESKTOP LAYOUT TABLE SUPPORT
|	**BLGLTZDD**	TIME ZONE DISPLAY
|	**BLGLTZDF**	UNIVERSAL TIME SUPPORT
	BLGLUMID	USER MODIFICATION ENTRY
	BLGLUMI1	USER MODIFICATION ENTRY
	BLGLUMI2	USER MODIFICATION ENTRY
	BLGLUMI3	USER MODIFICATION ENTRY
	BLGLUMI4	USER MODIFICATION ENTRY
	BLGLUMI5	USER MODIFICATION ENTRY
	BLGLUSER	MAIL INFORMATION ENTRY
	BLGLUTID	UPGRADE/TAPE ENTRY
	BLGLUTI1	UPGRADE/TAPE ENTRY
	BLGLUTI2	UPGRADE/TAPE ENTRY
	BLGLUTI3	UPGRADE/TAPE ENTRY
	BLGLUTI4	UPGRADE/TAPE ENTRY
	BLGLUTI5	UPGRADE/TAPE ENTRY
|	**BLGLVLSB**	DATA MODEL RECORD
	BLGLVLST	VALIDATION DATA ENTRY
	BLGLVSEL	VALIDATION DATA
|	**BLGRFCHG**	CREATE INDEX TSP
|	**BLGRFPRO**	CREATE INDEX TSP
|	**BLGRFSOL**	CREATE INDEX TSP
	BLGSSQMP	RELATIONAL DATA MAPPING TABLE DISPLAY
|	**BLGTFDAT**	UNIVERSAL TIME SUPPORT
|	**BLGTFZON**	UNIVERSAL TIME SUPPORT
|	**BLGTROLE**	PRIVILEGE CLASS
|	**BLGTSPXX**	OLD ESCALATION TSP
	BLG0BU00	PROBLEM SUMMARY
	BLG0BU10	PROBLEM DELETE VERIFICATION
	BLG0BU11	PROBLEM DELETE VERIFICATION

BLG0BU90	VERIFICATION OF USERS RECORD DELETE
BLG0B100	PROBLEM REPORTER ENTRY
BLG0B200	PROBLEM STATUS ENTRY
BLG0B300	PROBLEM CLOSE ENTRY
BLG0B400	PROBLEM SUPPLEMENTAL ENTRY
BLG0B402	PROBLEM SYMPTOM DATA
BLG0B500	PROBLEM SYNOPSIS ENTRY
BLG0B502	PROBLEM RESOLUTION DATA
BLG0B901	USERS TO NOTIFY
BLG0B902	USERS TO NOTIFY
BLG0B903	USERS TO NOTIFY
BLG0CU00	CHANGE REQUEST SUMMARY
BLG0CU01	CHANGE REQUEST SUMMARY
BLG0CU10	CHANGE DELETE VERIFICATION
BLG0CU11	CHANGE DELETE VERIFICATION
BLG0CU21	ACTIVITY SUMMARY
BLG0CU30	ACTIVITY DELETE VERIFICATION
BLG0CU31	ACTIVITY DELETE VERIFICATION
BLG0C015	CHANGE APPROVER DATA MENU
BLG0C051	SOFTWARE ADD
BLG0C052	SOFTWARE ALTER
BLG0C053	SOFTWARE ALTER BY INSTALLING
BLG0C054	SOFTWARE ALTER BY MODIFYING
BLG0C055	SOFTWARE ALTER BY REMOVING
BLG0C056	SOFTWARE REMOVE
BLG0C061	HARDWARE ADD
BLG0C062	HARDWARE ALTER
BLG0C063	HARDWARE ALTER BY INSTALLING
BLG0C064	HARDWARE ALTER BY MODIFYING
BLG0C065	HARDWARE ALTER BY REMOVING
BLG0C066	HARDWARE REMOVE
BLG0C071	DOCUMENT ADD
BLG0C072	DOCUMENT ALTER
BLG0C073	DOCUMENT REMOVE
BLG0C100	CHANGE REQUESTER ENTRY

BLG0C151	SOFTWARE ADD INQUIRY
BLG0C152	SOFTWARE ALTER INQUIRY
BLG0C153	SOFTWARE ALTER BY INSTALLING INQUIRY
BLG0C154	SOFTWARE ALTER BY MODIFYING INQUIRY
BLG0C155	SOFTWARE ALTER BY REMOVING INQUIRY
BLG0C156	SOFTWARE REMOVE INQUIRY
BLG0C161	HARDWARE ADD INQUIRY
BLG0C162	HARDWARE ALTER INQUIRY
BLG0C163	HARDWARE ALTER BY INSTALLING INQUIRY
BLG0C164	HARDWARE ALTER BY MODIFYING INQUIRY
BLG0C165	HARDWARE ALTER BY REMOVING INQUIRY
BLG0C166	HARDWARE REMOVE INQUIRY
BLG0C171	DOCUMENT ADD INQUIRY
BLG0C172	DOCUMENT ALTER INQUIRY
BLG0C173	DOCUMENT REMOVE INQUIRY
BLG0C200	CHANGE STATUS ENTRY
BLG0C300	CHANGE CLOSE ENTRY
BLG0C500	CHANGE APPROVER ENTRY
BLG0C600	ACTIVITY REQUESTER ENTRY
BLG0C620	ACTIVITY STATUS ENTRY
BLG0C630	ACTIVITY CLOSE ENTRY
BLG0C700	CHANGE REVIEWER ENTRY
BLG0C800	SOFTWARE DISTRIBUTION ENTRY
BLG0DATT	DATA ATTRIBUTE RECORDS DISPLAY
BLG0DFSM	Default file summary panel
BLG0DU01	HARDWARE COMPONENT SUMMARY
BLG0DU02	HARDWARE FEATURE SUMMARY
BLG0DU03	SOFTWARE COMPONENT SUMMARY
BLG0DU04	SOFTWARE FEATURE SUMMARY
BLG0DU05	HARDWARE FINANCIAL SUMMARY
BLG0DU06	SOFTWARE FINANCIAL SUMMARY
BLG0DU07	SERVICE SUMMARY
BLG0DU08	DATA CENTER SUMMARY
BLG0DU09	SYSTEM SUMMARY
BLG0DU10	SOFTWARE CONNECTION DELETE VERIFICATION

BLG0DU11	HARDWARE COMPONENT DELETE VERIFICATION
BLG0DU12	HARDWARE FEATURE DELETE VERIFICATION
BLG0DU13	SOFTWARE COMPONENT DELETE VERIFICATION
BLG0DU14	SOFTWARE FEATURE DELETE VERIFICATION
BLG0DU15	HARDWARE FINANCIAL DELETE VERIFICATION
BLG0DU16	SOFTWARE FINANCIAL DELETE VERIFICATION
BLG0DU17	SERVICE DELETE VERIFICATION
BLG0DU18	CENTER DELETE VERIFICATION
BLG0DU19	SYSTEM DELETE VERIFICATION
BLG0DU20	SOFTWARE CONNECTION DELETE VERIFICATION
BLG0DU21	HARDWARE COMPONENT DELETE VERIFICATION
BLG0DU22	HARDWARE FEATURE DELETE VERIFICATION
BLG0DU23	SOFTWARE COMPONENT DELETE VERIFICATION
BLG0DU24	SOFTWARE FEATURE DELETE VERIFICATION
BLG0DU25	HARDWARE FINANCIAL DELETE VERIFICATION
BLG0DU26	SOFTWARE FINANCIAL DELETE VERIFICATION
BLG0DU27	SERVICE DELETE VERIFICATION
BLG0DU28	CENTER DELETE VERIFICATION
BLG0DU29	SYSTEM DELETE VERIFICATION
BLG0DU30	HARDWARE COMPONENT SUMMARY
BLG0DU31	SOFTWARE COMPONENT SUMMARY
BLG0DU32	HARDWARE CONNECTION SUMMARY
BLG0DU33	HARDWARE SUBCOMPONENT SUMMARY
BLG0DU34	HARDWARE CONNECTION DELETE VERIFICATION
BLG0DU35	HARDWARE CONNECTION DELETE VERIFICATION
BLG0DU36	MODEL HARDWARE COMPONENT SUMMARY
BLG0DU37	MODEL HARDWARE COMPONENT SUMMARY
BLG0DU38	MODEL HARDWARE FEATURE SUMMARY
BLG0DU39	MODEL HARDWARE SUBCOMPONENT SUMMARY
BLG0DU40	SOFTWARE CONNECTION SUMMARY
BLG0DU42	HARDWARE SUBCOMPONENT DELETE VERIFICATION
BLG0DU43	HARDWARE SUBCOMPONENT DELETE VERIFICATION
BLG0DU44	MODEL SOFTWARE COMPONENT SUMMARY
BLG0DU45	MODEL SOFTWARE COMPONENT SUMMARY
BLG0DU46	MODEL SOFTWARE FEATURE SUMMARY

BLG0DU47	MODEL HARDWARE COMPONENT DELETE VERIFICATION
BLG0DU48	MODEL HARDWARE COMPONENT DELETE VERIFICATION
BLG0DU49	MODEL HARDWARE SUBCOMPONENT DELETE VERIFICATION
BLG0DU50	MODEL HARDWARE SUBCOMPONENT DELETE VERIFICATION
BLG0DU51	MODEL HARDWARE FEATURE DELETE VERIFICATION
BLG0DU52	MODEL SOFTWARE COMPONENT DELETE VERIFICATION
BLG0DU53	MODEL SOFTWARE COMPONENT DELETE VERIFICATION
BLG0DU54	MODEL HARDWARE FEATURE DELETE VERIFICATION
BLG0DU55	MODEL SOFTWARE FEATURE DELETE VERIFICATION
BLG0DU56	MODEL SOFTWARE FEATURE DELETE VERIFICATION
BLG0DU57	HARDWARE COMPONENT RENAME VERIFICATION
BLG0DU58	HARDWARE CONNECTION RENAME VERIFICATION
BLG0DU59	HARDWARE FEATURE RENAME VERIFICATION
BLG0DU60	HARDWARE SUBCOMPONENT RENAME VERIFICATION
BLG0DU61	HARDWARE FINANCIAL RENAME VERIFICATION
BLG0DU62	SOFTWARE COMPONENT RENAME VERIFICATION
BLG0DU63	SOFTWARE CONNECTION RENAME VERIFICATION
BLG0DU64	SOFTWARE FEATURE RENAME VERIFICATION
BLG0DU65	SOFTWARE FINANCIAL RENAME VERIFICATION
BLG0DU66	MODEL HARDWARE COMPONENT RENAME VERIFICATION
BLG0DU67	MODEL HARDWARE FEATURE RENAME VERIFICATION
BLG0DU68	MODEL HARDWARE SUBCOMPONENT RENAME VERIFICATION
BLG0DU69	MODEL SOFTWARE COMPONENT RENAME VERIFICATION
BLG0DU70	MODEL SOFTWARE FEATURE RENAME VERIFICATION
BLG0DU71	CENTER RENAME VERIFICATION
BLG0DU72	SYSTEM RENAME VERIFICATION
BLG0DU73	SERVICE RENAME VERIFICATION
BLG0DU74	HARDWARE COMPONENT SWAP VERIFICATION
BLG0DU75	HARDWARE SUBCOMPONENT SWAP VERIFICATION
BLG0DU76	SOFTWARE COMPONENT SWAP VERIFICATION
BLG0D100	HARDWARE COMPONENT ENTRY
BLG0D102	HARDWARE COMPONENT ENTRY
BLG0D105	HARDWARE COMPONENT SOURCE DEFINITION
BLG0D107	SPECIFICATION FOR HARDWARE MODEL COPY
BLG0D113	HARDWARE COMPONENT DIAGRAM MAP DATA ENTRY

BLG0D120	HARDWARE FEATURE ENTRY
BLG0D130	HARDWARE CONNECTION ENTRY
BLG0D140	HARDWARE SUBCOMPONENT ENTRY
BLG0D150	HARDWARE COMPONENT PRIMARY SUPPORT ENTRY
BLG0D151	HARDWARE COMPONENT SECONDARY SUPPORT ENTRY
BLG0D152	HARDWARE COMPONENT FINANCIAL ENTRY
BLG0D153	HARDWARE COMPONENT FINANCIAL ENTRY
BLG0D154	HARDWARE FEATURE FINANCIAL ENTRY
BLG0D157	HARDWARE FEATURE SUPPORT ENTRY
BLG0D160	HARDWARE SUBCOMPONENT SUPPORT ENTRY
BLG0D164	HARDWARE SUBCOMPONENT FINANCIAL ENTRY
BLG0D170	HARDWARE COMPONENT LINE ENTRY
BLG0D172	HARDWARE COMPONENT LOOP ENTRY
BLG0D200	SOFTWARE COMPONENT ENTRY
BLG0D204	SPECIFICATION FOR SOFTWARE MODEL COPY
BLG0D205	SOFTWARE COMPONENT SOURCE DEFINITION ENTRY
BLG0D220	SOFTWARE FEATURE ENTRY
BLG0D230	SOFTWARE COMPONENT CONNECTION ENTRY
BLG0D250	SOFTWARE COMPONENT SUPPORT ENTRY
BLG0D251	SOFTWARE COMPONENT SECONDARY SUPPORT ENTRY
BLG0D253	SOFTWARE COMPONENT FINANCIAL ENTRY
BLG0D254	SOFTWARE FEATURE FINANCIAL ENTRY
BLG0D257	SOFTWARE FEATURE SUPPORT ENTRY
BLG0D300	HARDWARE FINANCIAL ENTRY
BLG0D302	HARDWARE FINANCIAL ENTRY
BLG0D305	HARDWARE FINANCIAL - PURCHASE ENTRY
BLG0D306	HARDWARE FINANCIAL - RENTAL/LEASE ENTRY
BLG0D307	HARDWARE FINANCIAL - VPA ENTRY
BLG0D350	SOFTWARE FINANCIAL ENTRY
BLG0D353	SOFTWARE FINANCIAL - BASIC LICENSE ENTRY
BLG0D355	SOFTWARE FINANCIAL - DSLO LICENSE ENTRY
BLG0D356	SOFTWARE FINANCIAL ENTRY
BLG0D357	SOFTWARE FINANCIAL - VLA ENTRY
BLG0D400	SERVICE ENTRY
BLG0D500	DATA CENTER ENTRY

BLG0D600	SYSTEM ENTRY
BLG0D700	MODEL HARDWARE COMPONENT ENTRY
BLG0D703	MODEL HARDWARE COMPONENT ENTRY
BLG0D720	MODEL HARDWARE FEATURE ENTRY
BLG0D740	MODEL HARDWARE SUBCOMPONENT ENTRY
BLG0D750	MODEL HARDWARE COMPONENT PRIMARY SUPPORT ENTRY COMPONENT: __
BLG0D751	MODEL HARDWARE COMPONENT SECONDARY SUPPORT ENTRY COMPONENT: _
BLG0D752	MODEL HARDWARE COMPONENT FINANCIAL ENTRY
BLG0D753	MODEL HARDWARE COMPONENT FINANCIAL ENTRY
BLG0D754	MODEL HARDWARE FEATURE FINANCIAL ENTRY
BLG0D757	MODEL HARDWARE FEATURE SUPPORT ENTRY
BLG0D758	MODEL SOFTWARE COMPONENT PRIMARY SUPPORT ENTRY
BLG0D759	MODEL SOFTWARE COMPONENT SECONDARY SUPPORT ENTRY COMPONENT: _
BLG0D760	MODEL HARDWARE SUBCOMPONENT PRIMARY SUPPORT ENTRY SUBCOMP:
BLG0D764	MODEL HARDWARE SUBCOMPONENT FINANCIAL ENTRY
BLG0D770	MODEL COMPONENT LINE ENTRY
BLG0D772	MODEL COMPONENT LOOP ENTRY
BLG0D780	MODEL SOFTWARE COMPONENT ENTRY
BLG0D783	MODEL SOFTWARE COMPONENT FINANCIAL ENTRY
BLG0D784	MODEL SOFTWARE FEATURE FINANCIAL ENTRY
BLG0D787	MODEL SOFTWARE FEATURE SUPPORT ENTRY
BLG0D790	MODEL SOFTWARE FEATURE ENTRY
BLG0EN10	PRIMARY OPTIONS MENU
BLG0EN20	PRIMARY OPTIONS MENU
BLG0E090	PROBLEM INQUIRY SUMMARY
BLG0E190	PROBLEM REPORTER DATA INQUIRY
BLG0E290	PROBLEM STATUS DATA INQUIRY
BLG0E390	PROBLEM CLOSE DATA INQUIRY
BLG0E490	PROBLEM SYMPTOM DATA INQUIRY
BLG0E590	PROBLEM RESOLUTION DATA INQUIRY
BLG0E690	PROBLEM SUPPLEMENTAL DATA INQUIRY
BLG0E790	TSD BRIDGE DATA INQUIRY
BLG0E890	CONTROL DATA INQUIRY

BLG0FGFL	FIELD GROUP FIELD LIST SELECTION
BLG0F090	CHANGE REQUEST SUMMARY
BLG0F190	CHANGE REQUESTER INQUIRY
BLG0F270	CHANGE/ACTIVITY REQUEST SUMMARY
BLG0F271	CHANGE/ACTIVITY REQUESTER DATA INQUIRY
BLG0F272	CHANGE/ACTIVITY STATUS DATA INQUIRY
BLG0F273	CHANGE/ACTIVITY CLOSE DATA INQUIRY
BLG0F278	CHANGE/ACTIVITY CONTROL DATA INQUIRY
BLG0F280	ACTIVITY SUMMARY
BLG0F281	ACTIVITY REQUESTER DATA INQUIRY
BLG0F282	ACTIVITY STATUS DATA INQUIRY
BLG0F283	ACTIVITY CLOSE DATA INQUIRY
BLG0F288	ACTIVITY CONTROL DATA INQUIRY
BLG0F290	CHANGE STATUS DATA INQUIRY
BLG0F390	CHANGE CLOSE DATA INQUIRY
BLG0F590	CHANGE APPROVER DATA INQUIRY
BLG0F890	CHANGE CONTROL DATA INQUIRY
BLG0GU10	ALIAS RECORD DELETE VERIFICATION
BLG0GU11	ALIAS RECORD DELETE VERIFICATION
BLG0G1A1	SOFTWARE COMPONENT DESCRIPTION/CONTROL DATA INQUIRY
BLG0G1A2	SOFTWARE SUPPORT/FMID/FINANCIAL INQUIRY
BLG0G1A3	SOFTWARE COMPONENT SUMMARY
BLG0G1A4	HARDWARE CONNECTION INQUIRY
BLG0G1A5	HARDWARE CONNECTION SUMMARY
BLG0G1B1	SOFTWARE FEATURE DESCRIPTION/CONTROL DATA INQUIRY
BLG0G1B2	SOFTWARE FEATURE FMID/SUPPORT/FINANCIAL INQUIRY
BLG0G1B3	SOFTWARE FEATURE SUMMARY
BLG0G1B4	SOFTWARE COMPONENT CONNECTION INQUIRY
BLG0G1B5	SOFTWARE COMPONENT CONNECTION SUMMARY
BLG0G137	HARDWARE RECORDS+
BLG0G138	HARDWARE RECORDS+
BLG0G167	SOFTWARE RECORDS+ INQUIRY
BLG0G168	SOFTWARE RECORDS+ SUMMARY
BLG0G170	HARDWARE SUBCOMPONENT SUMMARY
BLG0G171	HARDWARE SUBCOMPONENT DESCRIPTION/CONTROL DATA INQUIRY

BLG0G172	HARDWARE SUBCOMPONENT SUPPORT/EC LEVELS/FINANCIAL INQUIRY
BLG0G182	HARDWARE FEATURE SUPPORT/EC LEVELS/FINANCIAL DATA INQUIRY
BLG0G183	HARDWARE FEATURE SUMMARY
BLG0G184	HARDWARE FEATURE DESCRIPTION/CONTROL DATA INQUIRY
BLG0G190	HARDWARE COMPONENT INQUIRY SUMMARY
BLG0G191	HARDWARE PRIMARY DESCRIPTION/SUPPORT DATA INQUIRY
BLG0G192	HARDWARE SECONDARY DESCRIPTION/SUPPORT DATA INQUIRY
BLG0G193	HARDWARE FINANCIAL/CONTROL DATA INQUIRY
BLG0G290	SYSTEM INQUIRY
BLG0G293	SYSTEM SUMMARY
BLG0G337	ALL FINANCIAL+ INQUIRY
BLG0G338	ALL FINANCIAL+ SUMMARY
BLG0G357	SOFTWARE FINANCIAL DESCRIPTION DATA INQUIRY
BLG0G358	SOFTWARE FINANCIAL SUMMARY
BLG0G359	SOFTWARE FINANCIAL VLA/CONTROL DATA INQUIRY
BLG0G390	HARDWARE FINANCIAL SUMMARY
BLG0G391	HARDWARE FINANCIAL DESCRIPTION DATA INQUIRY
BLG0G392	HARDWARE FINANCIAL SECONDARY DESCRIPTION DATA INQUIRY
BLG0G490	SERVICE INQUIRY
BLG0G493	SERVICE SUMMARY
BLG0G590	DATA CENTER INQUIRY
BLG0G593	DATA CENTER SUMMARY
BLG0G727	MODEL HARDWARE FEATURE INQUIRY
BLG0G728	MODEL HARDWARE FEATURE SUMMARY
BLG0G737	MODEL HARDWARE SUBCOMPONENT INQUIRY
BLG0G738	MODEL HARDWARE RECORDS+
BLG0G739	MODEL HARDWARE SUBCOMPONENT SUMMARY
BLG0G742	MODEL HARDWARE RECORDS+
BLG0G747	MODEL FEATURE RECORDS+
BLG0G748	MODEL COMPONENT RECORDS+
BLG0G749	MODEL FEATURE RECORDS+
BLG0G751	MODEL COMPONENT RECORDS+
BLG0G770	MODEL SOFTWARE COMPONENT SUMMARY

BLG0G771	MODEL SOFTWARE COMPONENT DESCRIPTION/SUPPORT DATA INQUIRY
BLG0G772	MODEL SOFTWARE COMPONENT OTHER DATA INQUIRY
BLG0G780	MODEL SOFTWARE FEATURE SUMMARY
BLG0G781	MODEL SOFTWARE FEATURE DESCRIPTION DATA INQUIRY
BLG0G782	MODEL SOFTWARE FEATURE SUPPORT/FINANCIAL DATA INQUIRY
BLG0G787	MODEL SOFTWARE RECORDS+
BLG0G788	MODEL SOFTWARE RECORDS+
BLG0G790	MODEL HARDWARE COMPONENT SUMMARY
BLG0G791	MODEL HARDWARE DESCRIPTION DATA INQUIRY
BLG0G792	MODEL HARDWARE SUPPORT/FINANCIAL/CONTROL DATA INQUIRY
BLG0G890	CONTROL INQUIRY
BLG0G893	CONTROL SUMMARY
BLG0HU10	COMMAND RECORD DELETE VERIFICATION
BLG0HU11	COMMAND RECORD DELETE VERIFICATION
BLG0IU00	ESCALATION RULES INQUIRY SUMMARY
BLG0I0E0	APAR DISPLAY
BLG0I100	ESCALATION RULES INQUIRY
BLG0I200	ESCALATION CONTROL INQUIRY
BLG0JU00	CLASS SUMMARY
BLG0JU10	CLASS DELETE VERIFICATION
BLG0JU11	CLASS DELETE VERIFICATION
BLG0J100	CLASS DESCRIPTION ENTRY
BLG0J200	PROBLEM AUTHORITY
BLG0J2C0	SOLUTION RECORD
BLG0J220	CHANGE AUTHORITY
BLG0J230	CONFIGURATION AUTHORITY
BLG0J240	FINANCIAL AUTHORITY
BLG0J250	ESCALATION RULES AUTHORITY
BLG0J260	CLASS AUTHORITY
BLG0J270	PMF AUTHORITY
BLG0J280	PEOPLE AUTHORITY
BLG0J290	SRC AUTHORITY
BLG0J295	ADMINISTRATION AUTHORITY
BLG0J300	CLASS USER ENTRY

|	**BLG0J400**	PRIVILEGE CLASS
|	**BLG0J401**	PRIVILEGE CLASS
	BLG0K091	CLASS SUMMARY
	BLG0K190	CLASS DESCRIPTION INQUIRY
	BLG0K291	MANAGEMENT CLASS AUTHORITY
	BLG0K292	SYSTEM CLASS AUTHORITY
	BLG0K293	RECORD AUTHORITY
|	**BLG0K400**	PRIVILEGE CLASS
	BLG0L100	PROBLEM REPORTER DISPLAY
	BLG0L200	PROBLEM STATUS DISPLAY
	BLG0L300	PROBLEM CLOSE DISPLAY
	BLG0L400	PROBLEM SUPPLEMENTAL DISPLAY
	BLG0L500	PROBLEM SYNOPSIS DISPLAY
	BLG0L600	INTERESTED CLASS DISPLAY
	BLG0L700	TSD BRIDGE DISPLAY
	BLG0L901	USERS TO NOTIFY
	BLG0L902	USERS TO NOTIFY
	BLG0L903	USERS TO NOTIFY
	BLG0M100	CHANGE REQUESTER DISPLAY
	BLG0M200	CHANGE STATUS DISPLAY
	BLG0M300	CHANGE CLOSE DISPLAY
	BLG0M500	CHANGE APPROVER DISPLAY
	BLG0M600	ACTIVITY REQUESTER DISPLAY
	BLG0M620	ACTIVITY STATUS DISPLAY
	BLG0M630	ACTIVITY CLOSE DISPLAY
	BLG0M700	CHANGE REVIEWER DISPLAY
	BLG0M800	SOFTWARE DISTRIBUTION DISPLAY
	BLG0N100	HARDWARE COMPONENT SUMMARY DISPLAY
	BLG0N101	HARDWARE COMPONENT DISPLAY
	BLG0N105	HARDWARE COMPONENT SOURCE DISPLAY
	BLG0N110	HARDWARE COMPONENT RECORD UTILITIES
	BLG0N113	HARDWARE COMPONENT DIAGRAM MAP DATA DISPLAY
	BLG0N120	HARDWARE FEATURE SUMMARY DISPLAY
	BLG0N121	HARDWARE FEATURE RECORD UTILITIES
	BLG0N122	HARDWARE FEATURE DISPLAY

BLG0N130	HARDWARE CONNECTION DISPLAY
BLG0N132	HARDWARE CONNECTION DISPLAY
BLG0N133	HARDWARE CONNECTION RECORD UTILITIES
BLG0N134	PATH DISPLAY SPECIFICATION
BLG0N140	HARDWARE SUBCOMPONENT SUMMARY DISPLAY
BLG0N141	HARDWARE SUBCOMPONENT RECORD UTILITIES
BLG0N142	HARDWARE SUBCOMPONENT DISPLAY
BLG0N150	HARDWARE COMPONENT PRIMARY SUPPORT DISPLAY
BLG0N151	HARDWARE COMPONENT SECONDARY SUPPORT DISPLAY
BLG0N153	HARDWARE COMPONENT FINANCIAL DISPLAY
BLG0N154	HARDWARE FEATURE FINANCIAL DISPLAY
BLG0N157	HARDWARE FEATURE SUPPORT DISPLAY
BLG0N160	HARDWARE SUBCOMPONENT SUPPORT DISPLAY
BLG0N164	HARDWARE SUBCOMPONENT FINANCIAL DISPLAY
BLG0N170	HARDWARE COMPONENT LINE DISPLAY
BLG0N172	HARDWARE COMPONENT LOOP DISPLAY
BLG0N200	SOFTWARE COMPONENT SUMMARY DISPLAY
BLG0N201	SOFTWARE COMPONENT DISPLAY
BLG0N205	SOFTWARE COMPONENT SOURCE DEFINITION DISPLAY
BLG0N210	SOFTWARE COMPONENT RECORD UTILITIES
BLG0N220	SOFTWARE FEATURE SUMMARY DISPLAY
BLG0N221	SOFTWARE FEATURE RECORD UTILITIES
BLG0N222	SOFTWARE FEATURE DISPLAY
BLG0N230	SOFTWARE CONNECTION SUMMARY DISPLAY
BLG0N231	SOFTWARE CONNECTION RECORD UTILITIES
BLG0N232	SOFTWARE CONNECTION DISPLAY
BLG0N250	SOFTWARE COMPONENT SUPPORT DISPLAY
BLG0N251	SOFTWARE COMPONENT SECONDARY SUPPORT DISPLAY
BLG0N253	SOFTWARE COMPONENT FINANCIAL DISPLAY
BLG0N254	SOFTWARE FEATURE FINANCIAL DISPLAY
BLG0N257	SOFTWARE FEATURE SUPPORT DISPLAY
BLG0N300	HARDWARE FINANCIAL SUMMARY DISPLAY
BLG0N301	HARDWARE FINANCIAL DISPLAY
BLG0N305	HARDWARE FINANCIAL - PURCHASE DISPLAY
BLG0N306	HARDWARE FINANCIAL - RENTAL/LEASE DISPLAY

BLG0N307	HARDWARE FINANCIAL - VPA DISPLAY
BLG0N310	HARDWARE FINANCIAL RECORD UTILITIES
BLG0N320	HARDWARE FINANCIAL DISPLAY
BLG0N350	SOFTWARE FINANCIAL SUMMARY DISPLAY
BLG0N352	SOFTWARE FINANCIAL DISPLAY
BLG0N353	SOFTWARE FINANCIAL - BASIC LICENSE DISPLAY
BLG0N355	SOFTWARE FINANCIAL - DSLO LICENSE DISPLAY
BLG0N357	SOFTWARE FINANCIAL - VLA DISPLAY
BLG0N360	SOFTWARE FINANCIAL RECORD UTILITIES
BLG0N400	SERVICE SUMMARY DISPLAY
BLG0N410	SERVICE RECORD UTILITIES
BLG0N500	DATA CENTER SUMMARY DISPLAY
BLG0N510	DATA CENTER RECORD UTILITIES
BLG0N600	SYSTEM SUMMARY DISPLAY
BLG0N610	SYSTEM RECORD UTILITIES
BLG0N700	MODEL HARDWARE COMPONENT SUMMARY DISPLAY
BLG0N701	MODEL HARDWARE COMPONENT DISPLAY
BLG0N710	MODEL HARDWARE COMPONENT RECORD UTILITIES
BLG0N720	MODEL HARDWARE FEATURE SUMMARY DISPLAY
BLG0N721	MODEL HARDWARE FEATURE RECORD UTILITIES
BLG0N722	MODEL HARDWARE FEATURE DISPLAY
BLG0N740	MODEL HARDWARE SUBCOMPONENT DISPLAY
BLG0N741	MODEL HARDWARE SUBCOMPONENT RECORD UTILITIES
BLG0N742	MODEL HARDWARE SUBCOMPONENT DISPLAY
BLG0N750	MODEL HARDWARE COMPONENT PRIMARY SUPPORT DISPLAY COMPONENT: _
BLG0N751	MODEL HARDWARE COMPONENT SECONDARY SUPPORT DISPLAY COMPONENT:
BLG0N753	MODEL HARDWARE COMPONENT FINANCIAL DISPLAY
BLG0N754	MODEL HARDWARE FEATURE FINANCIAL DISPLAY
BLG0N757	MODEL HARDWARE FEATURE SUPPORT DISPLAY
BLG0N758	MODEL SOFTWARE COMPONENT PRIMARY SUPPORT DISPLAY COMPONENT: _
BLG0N759	MODEL SOFTWARE COMPONENT SECONDARY SUPPORT DISPLAY COMPONENT:
BLG0N760	MODEL HARDWARE SUBCOMPONENT SUPPORT DISPLAY
BLG0N764	MODEL HARDWARE SUBCOMPONENT FINANCIAL DISPLAY

BLG0N770	MODEL HARDWARE COMPONENT LINE DISPLAY
BLG0N772	MODEL HARDWARE COMPONENT LOOP DISPLAY
BLG0N780	MODEL SOFTWARE COMPONENT SUMMARY DISPLAY
BLG0N781	MODEL SOFTWARE COMPONENT RECORD UTILITIES
BLG0N782	MODEL SOFTWARE COMPONENT DISPLAY
BLG0N783	MODEL SOFTWARE COMPONENT FINANCIAL DISPLAY
BLG0N784	MODEL SOFTWARE FEATURE FINANCIAL
BLG0N787	MODEL SOFTWARE FEATURE SUPPORT DISPLAY
BLG0N790	MODEL SOFTWARE FEATURE SUMMARY DISPLAY
BLG0N791	MODEL SOFTWARE FEATURE RECORD UTILITIES
BLG0N792	MODEL SOFTWARE FEATURE DISPLAY
BLG0PU00	PROFILE SUMMARY
BLG0P100	SESSION DEFAULTS
BLG0P101	SESSION DEFAULTS
BLG0P200	SCREEN CONTROL DEFAULTS
BLG0P300	REPORT CONTROL DEFAULTS
BLG0P410	PRINT SYSOUT DESTINATION ENTRY
BLG0P420	PRINT DATA SET DESTINATION ENTRY
BLG0P430	PRINT DDNAME DESTINATION ENTRY
BLG0P510	STANDARD REPORT SYSOUT DESTINATION ENTRY
BLG0P511	STANDARD REPORT SYSOUT DESTINATION ENTRY
BLG0P520	STANDARD REPORT DATA SET DESTINATION ENTRY
BLG0P521	STANDARD REPORT DATA SET DESTINATION ENTRY
BLG0P530	STANDARD REPORT DDNAME DESTINATION ENTRY
BLG0P531	STANDARD REPORT DDNAME DESTINATION ENTRY
BLG0P700	DATA DEFINITION DEFAULTS
BLG0P810	DRAW REPORT SYSOUT DESTINATION ENTRY
BLG0P811	DRAW REPORT SYSOUT DESTINATION ENTRY
BLG0P820	DRAW REPORT DATA SET DESTINATION ENTRY
BLG0P821	DRAW REPORT DATA SET DESTINATION ENTRY
BLG0P830	DRAW REPORT DDNAME DESTINATION ENTRY
BLG0P831	DRAW REPORT DDNAME DESTINATION ENTRY
BLG0QUP0	RECORD UTILITIES
BLG0Q100	CLASS SUMMARY DISPLAY
BLG0Q200	PROBLEM AUTHORITY

|	**BLG0Q2C0**	SOLUTION RECORD
	BLG0Q213	AUTHORITY SUMMARY
	BLG0Q214	AUTHORITY SUMMARY
	BLG0Q215	AUTHORITY SUMMARY
	BLG0Q216	AUTHORITY SUMMARY
	BLG0Q220	CHANGE AUTHORITY
	BLG0Q230	CONFIGURATION AUTHORITY
	BLG0Q240	FINANCIAL AUTHORITY
	BLG0Q250	ESCALATION RULES AUTHORITY
	BLG0Q260	CLASS AUTHORITY
	BLG0Q270	PMF AUTHORITY
	BLG0Q280	PEOPLE AUTHORITY
	BLG0Q290	SRC AUTHORITY
	BLG0Q300	ELIGIBLE USER DISPLAY
|	**BLG0Q400**	PRIVILEGE CLASS
	BLG0RU00	ESCALATION RULES DISPLAY
	BLG0R010	ESCALATION RULES DELETE VERIFICATION
	BLG0R011	ESCALATION RULES DELETE VERIFICATION
	BLG0R020	RECORD UTILITIES
	BLG0R100	ESCALATION RULES DISPLAY
	BLG0SU50	AUTOMATIC LOG SAVE SUMMARY DISPLAY
	BLG0S010	PROBLEM SUMMARY DISPLAY
	BLG0S011	RECORD UTILITIES
	BLG0S020	CHANGE SUMMARY DISPLAY
	BLG0S021	RECORD UTILITIES
	BLG0S060	ACTIVITY SUMMARY DISPLAY
	BLG0S061	RECORD UTILITIES
	BLG0S500	AUTOMATIC LOG SAVE DATA SET INFORMATION DISPLAY
	BLG0S510	AUTOMATIC LOG SAVE SYNCHRONIZE AND FILTER DISPLAY
	BLG0S530	SQL EXTRACT INFORMATION DISPLAY
	BLG0TSFL	Task Field List Selection
	BLG0TU01	SRC DELETE VERIFICATION
	BLG0TU02	SRC DELETE VERIFICATION
	BLG0T100	SRC SUMMARY DISPLAY
	BLG0T101	RECORD UTILITIES

BLG0T110	SRC DESCRIPTION DISPLAY
BLG0T120	FINAL RESPONSES DISPLAY
BLG0T200	SRC EXECUTION CLASSES
BLG0USER	MAIL INFORMATION DISPLAY
BLG0U000	SRC SUMMARY
BLG0U001	SRC SUMMARY
BLG0U100	SRC DESCRIPTION ENTRY
BLG0U120	FINAL SRC RESPONSE ENTRY
BLG0U200	SRC EXECUTION CLASSES
BLG0VD50	VALIDATION RECORD DELETE VERIFICATION
BLG0VD51	VALIDATION RECORD DELETE VERIFICATION
BLG0VD60	DATA VIEW RECORD DELETE VERIFICATION
BLG0VD61	DATA VIEW RECORD DELETE VERIFICATION
BLG0VD70	DATA ATTRIBUTE RECORD DELETE VERIFICATION
BLG0VD71	DATA ATTRIBUTE RECORD DELETE VERIFICATION
BLG0VE60	DATA VIEW INQUIRY SUMMARY
BLG0VE61	DATA VIEW RECORD INQUIRY
BLG0VE62	AUTHORIZATION CODE INQUIRY
BLG0VE70	DATA ATTRIBUTE INQUIRY SUMMARY
BLG0VE71	DATA ATTRIBUTE RECORD INQUIRY
BLG0VE72	RESPONSE PROCESSING INQUIRY
BLG0VE73	STRUCTURED WORD AND VALIDATION INQUIRY
BLG0VE74	PANEL FLOW INQUIRY
BLG0VE75	CONTROL DATA INQUIRY
BLG0VLST	VALIDATION DATA DISPLAY
BLG0VQ50	VALIDATION SUMMARY DISPLAY
BLG0VQ52	VALIDATION CONTROL DATA DISPLAY
BLG0VQ60	DATA VIEW SUMMARY DISPLAY
BLG0VQ61	DATA VIEW RECORD DISPLAY
BLG0VQ62	AUTHORIZATION CODE DISPLAY
BLG0VQ70	DATA ATTRIBUTE SUMMARY
BLG0VQ71	DATA ATTRIBUTE RECORD DISPLAY
BLG0VQ72	RESPONSE PROCESSING DISPLAY
BLG0VQ73	STRUCTURED WORD AND VALIDATION DISPLAY
BLG0VQ74	PANEL FLOW DISPLAY

	BLG0VU50	VALIDATION SUMMARY
	BLG0VU60	DATA VIEW SUMMARY
	BLG0VU70	DATA ATTRIBUTE SUMMARY
	BLG0V090	SRC SUMMARY
	BLG0V190	SRC DESCRIPTION INQUIRY
	BLG0V500	VALIDATION RECORD ENTRY
	BLG0V502	VALIDATION DISPLAY UPDATE
	BLG0V510	VALIDATION RECORD INQUIRY
	BLG0V600	DATA VIEW RECORD ENTRY
	BLG0V610	AUTHORIZATION CODE ENTRY
	BLG0V700	DATA ATTRIBUTE RECORD ENTRY
	BLG0V710	RESPONSE PROCESSING
	BLG0V720	STRUCTURED WORD AND VALIDATION COLLECTION
	BLG0V730	PANEL FLOW UPDATE
	BLG0W200	PERIODIC CHANGE STATUS REPORT
	BLG0W210	PERIODIC PROBLEM STATUS REPORT
	BLG0W250	EXTRACT DIAGRAM DATA SPECIFICATION
	BLG0W500	REPORT ENTRY
	BLG0W510	BROWSE/PRINT
	BLG0X100	GLOSSARY
	BLG0Y200	CHECK IN
	BLG0Y300	TSD BRIDGE CLEANUP
	BLG0ZD50	VERIFICATION OF LOGSAVE RECORD DELETE
	BLG0ZU00	ESCALATION RULES SUMMARY
	BLG0ZU50	AUTOMATIC LOG SAVE RECORD SUMMARY
	BLG0Z100	ESCALATION RULES ENTRY
|	**BLG0Z200**	TIME ZONE RECORD SUMMARY
|	**BLG0Z250**	TIME ZONE DISPLAY
	BLG0Z500	AUTOMATIC LOG SAVE DATA SET INFORMATION ENTRY
	BLG0Z510	AUTOMATIC LOG SAVE SYNCHRONIZE AND FILTER ENTRY
	BLG0Z530	DB2 EXTRACT INFORMATION ENTRY
|	**BLG1AAPD**	DISPLAY CHANGE APPROVER CONTROL
|	**BLG1AAPU**	UPDATE CHANGE APPROVER CONTROL
|	**BLG1AELD**	ELIGIBLE USER CONTROL
|	**BLG1AELU**	ELIGIBLE USER CONTROL

BLG1APST	APPROVAL STATUS DISPLAY CONTROL
BLG1ARED	DISPLAY CHANGE REVIEWER CONTROL
BLG1AREU	UPDATE CHANGE REVIEWER CONTROL
BLG1ASTS	TIME ZONE CONTROL
BLG1ATSA	SET INDEX FOR TEXT SEARCH
BLG1AZDF	TIME ZONE CONTROL
BLG1A11B	CREATE SOLUTION RECORD
BLG1C015	RESET CHANGE APPROVALS CONTROL
BLG1AROLE	PRIVILEGE CLASS CONTROL
BLG1TDEH	HELP TEXT
BLG1TEHD	HELP TEXT
BLG1TTSA	TEXT SEARCH ARGUMENT INPUT
BLG1UT01	UTILITY ENTRY DIALOG
BLG1UT02	RENAME COMPONENT DIALOG
BLG1UT07	CONFIGURATION DIAGRAM PARAMETER ENTRY
BLG1UT20	SWAP COMPONENTS
BLG1UT31	CONFIGURATION DIAGRAM INPUT DDNAME ENTRY
BLG1UT32	CONFIGURATION DIAGRAM INPUT DATA SET ENTRY
BLG1VQ50	UNIVERSAL TIME SUPPORT CONTROL
BLG1VQ70	UNIVERSAL TIME SUPPORT CONTROL
BLG1VU51	DATA MODEL RECORD CONTROL
BLG1VU61	DATA VIEW TABLE SUPPORT CONTROL
BLG1VU71	DATA MODEL RECORD CONTROL
BLG2DTTM	DATETIME RECORD HELP PANEL
BLG2GNRC	HELP PANEL SKELETON
BLG2J400	PRIVILEGE CLASS HELP
BLG2Q400	PRIVILEGE CLASS HELP
BLG2SUPP	PRIVILEGE CLASS HELP
BLG2TBDD	DATA VIEW TABLE SUPPORT HELP
BLG2TBDL	DATA VIEW TABLE SUPPORT HELP
BLG2TSTD	DATA VIEW DESKTOP TABLE SUPPORT HELP
BLG2TSTL	DATA VIEW DESKTOP TABLE SUPPORT HELP
BLG2TTSA	TEXT SEARCH ARGUMENT INPUT HELP
BLG2ZSC7	ONLINE OVERVIEW HELP
BLG6APSL	ACTION APPROVE/REJECT

|	**BLG6APST**	APPROVER STATUS
|	**BLG6APVR**	APPROVER
|	**BLG6DTRC**	UNIVERSAL TIME SUPPORT
|	**BLG6DTYP**	ALTERNATE PROBLEM
|	**BLG6DVCH**	CHILD DATA VIEW ASSISTED ENTRY
|	**BLG6EFND**	PROFILE-ENTRY FIELD NUMBER DISPLAY AE
|	**BLG6PELI**	PRIVILEGE CLASS
|	**BLG6REVD**	REVIEWER
|	**BLG6TBAL**	DESKTOP LAYOUT TABLE SUPPORT
|	**BLG6TBLD**	DATA VIEW – DESKTOP TABLE SUPPORT
|	**BLG6TBLP**	DATA VIEW TABLE SUPPORT
|	**BLG6TBNM**	DATA VIEW TABLE SUPPORT
|	**BLG6TBSI**	DATA VIEW TABLE SUPPORT
|	**BLG6TBSW**	DATA VIEW TABLE SUPPORT
|	**BLG6URN5**	INDEX RECORD VALIDATION
|	**BLG6UTZN**	PROFILE TIME ZONE AE
	BLHLI122	PORTS
	BLHLI124	CO-PROCESSORS
	BLHLI125	HARD DISKS
	BLHLI126	LOGICAL DRIVES
	BLHLI127	NETWORK NODES
	BLHLI128	IPX LAN
	BLHLI129	IPX LAN CONNECTIONS
	BLHLI130	CONFIGURATION FILES
	BLHLI131	INSTALLED SOFTWARE
|	**BLHLNTXT**	TEXT SEARCH INPUT LIST
|	**BLHLX0BC**	UPDATE BCC ADDRESS
|	**BLHLX0CC**	UPDATE CC ADDRESS
|	**BLHLX0TO**	UPDATE TO ADDRESS
|	**BLHLX1BC**	DISPLAY BCC ADDRESSES
|	**BLHLX1CC**	DISPLAY CC ADDRESSES
|	**BLHLX1TO**	DISPLAY TO ADDRESSES
|	**BLHTSNDE**	TSP FOR INDEX DELETE PROCESSING
	BLH0IU20	HARDWARE INVENTORY DELETE VERIFICATION
	BLH0IU21	HARDWARE INVENTORY DELETE VERIFICATION

	BLH0I000	TIVOLI INVENTORY QUERY MENU
	BLH0I001	TIVOLI INVENTORY
	BLH0I002	TIVOLI INVENTORY
	BLH0I004	TIVOLI INVENTORY REPORT MENU
	BLH0I010	ASSOCIATED DATA MENU
	BLH0I011	PC ASSOCIATED MENU
	BLH0I012	UNIX ASSOCIATED DATA
	BLH0I022	HARDWARE QUERY MENU
	BLH0I023	SOFTWARE QUERY MENU
	BLH0I12A	NETWARE SERVER
	BLH0I120	COMPUTER SYSTEM
	BLH0I121	PC BIOS
	BLH0I123	PC MEMORY
	BLH0I132	NT INFORMATION
	BLH0I22A	NETWARE SERVER QUERY
	BLH0I220	COMPUTER SYSTEM QUERY
	BLH0I221	PC BIOS QUERY
	BLH0I222	PORTS QUERY
	BLH0I223	PC MEMORY QUERY
	BLH0I224	CO-PROCESSORS QUERY
	BLH0I225	HARD DISK QUERY
	BLH0I226	LOGICAL DRIVES QUERY
	BLH0I227	NETWORK NODES QUERY
	BLH0I228	IPX LAN QUERY
	BLH0I229	IPX LAN CONNECTIONS QUERY
	BLH0I230	CONFIGURATION FILES QUERY
	BLH0I231	INSTALLED SOFTWARE QUERY
	BLH0I232	NT INFORMATION QUERY
|	**BLH0N000**	INDEX ENTRY OPTIONS MENU
|	**BLH0N001**	INDEX RECORD ENTRY
|	**BLH0N010**	INDEX RECORD ENTRY FILE
|	**BLH0N090**	INDEX RECORD VERIFY DELETE
|	**BLH0N095**	INDEX RECORD VERIFY DELETE
|	**BLH0N096**	INDEX RECORD VERIFY MULTIPLE DELETE
|	**BLH0N110**	INDEX RECORD SUMMARY DISPLAY

|	**BLH0N200**	INDEX RECORD INQUIRY
|	**BLH0S000**	SOLUTION DIALOG ENTRY
|	**BLH0S001**	SOLUTION RECORD ENTRY
|	**BLH0S010**	SOLUTION ENTRY SUMMARY
|	**BLH0S011**	SOLUTION RECORD UTILITIES
|	**BLH0S090**	SOLUTION RECORD VERIFY DELETE
|	**BLH0S091**	SOLUTION VERIFY MULTIPLE DELETE
|	**BLH0S100**	SOLUTION RECORD DISPLAY
|	**BLH0S110**	SOLUTION DISPLAY SUMMARY
|	**BLH0S111**	SOLUTION RECORD UTILITIES
|	**BLH0S200**	SOLUTION RECORD INQUIRY
|	**BLH0S201**	SOLUTION CONTROL DATA INQUIRY
|	**BLH0X010**	GENERIC TEXT ENTRY
|	**BLH0X011**	UPDATE ADDRESSES SELECTION
|	**BLH0X015**	FILE MESSAGE RECORD
|	**BLH0X090**	GENERIC TEXT DELETE VERIFY
|	**BLH0X091**	GENERIC TEXT MULTIPLE DELETE
|	**BLH0X110**	GENERIC TEXT DISPLAY SUMMARY
|	**BLH0X111**	DISPLAY ADDRESSES SELECTION
|	**BLH1N000**	INDEX RECORD ACCESS CONTROL
|	**BLH1N001**	INDEX RECORD PROCESSING CONTROL
|	**BLH1N003**	UPDATE INDEX RECORD OWNERSHIP CONTROL
|	**BLH1N004**	DELETE INDEX RECORD OWNERSHIP CONTROL
|	**BLH1N008**	INDEX RECORD PRINT CONTROL
|	**BLH1N009**	INDEX RECORD DELETE CONTROL
|	**BLH1N010**	INDEX RECORD CREATE CONTROL
|	**BLH1N011**	CREATE SOLUTION INDEX RECORD CONTROL
|	**BLH1N012**	CREATE PROBLEM INDEX RECORD CONTROL
|	**BLH1N013**	CREATE CHANGE INDEX RECORD CONTROL
|	**BLH1N019**	INDEX RECORD DELETE PROCESSING CONTROL
|	**BLH1N020**	INDEX RECORD CREATE END PROCESSING CONTROL
|	**BLH1N040**	INDEX RECORD FILE CONTROL
|	**BLH1S000**	SOLUTION RECORD ACCESS CONTROL
|	**BLH1S003**	SOLUTION RECORD OWNERSHIP TEST CONTROL
|	**BLH1S004**	SOLUTION DELETE OWNERSHIP TEST CONTROL

|	**BLH1S006**	SOLUTION RECORD COPY CONTROL
|	**BLH1S008**	SOLUTION RECORD PRINT CONTROL
|	**BLH1S009**	SOLUTION RECORD DELETE CONTROL
|	**BLH1S010**	VALIDATE PROBLEM RNID CONTROL
|	**BLH1S011**	START SOLUTION DIALOG CONTROL
|	**BLH1S012**	SOLUTION RECORD MULTIPLE DELETE CONTROL
|	**BLH1S013**	ADD PROBLEM DATA TO SOLUTION CONTROL
|	**BLH1S014**	VALIDATE SOLUTION FF TEXT DATA CONTROL
|	**BLH1S040**	SOLUTION RECORD FILE CONTROL
|	**BLH1TMTT**	GENERIC TEXT TEXT ENTRY
|	**BLH1TTXT**	TEXT SEARCH RESULTS LIST
|	**BLH1X000**	GENERIC TEXT RECORD ACCESS CONTROL
|	**BLH1X004**	GENERIC TEXT OWNERSHIP CONTROL
|	**BLH1X006**	GENERIC TEXT COPY CONTROL
|	**BLH1X009**	GENERIC TEXT DELETE CONTROL
|	**BLH1X010**	VALIDATE RECORD ENTRY CONTROL
|	**BLH1X011**	VALIDATE MODEL RECORD ENTRY CONTROL
|	**BLH1X012**	GENERIC TEXT DELETE CONTROL
|	**BLH1X040**	GENERIC TEXT FILE CONTROL
|	**BLH2S110**	SOLUTION DISPLAY SUMMARY HELP
|	**BLH2X110**	GENERIC TEXT DISPLAY SUMMARY HELP
|	**BLH2X111**	DISPLAY ADDRESSES HELP
|	**BLH6INDX**	TEXT INDEX NAME
	BLM0B000	PEOPLE SUMMARY
	BLM0B001	PEOPLE INFORMATION
	BLM0B002	CONTACT INFORMATION
	BLM0B050	PEOPLE SUMMARY DISPLAY
	BLM0B051	PEOPLE INFORMATION DISPLAY
	BLM0B052	CONTACT INFORMATION DISPLAY
	BLM0B060	PEOPLE RECORD UTILITIES
	BLM0B100	PEOPLE INQUIRY SUMMARY
	BLM0B101	PEOPLE INFORMATION INQUIRY
	BLM0B102	CONTACT INFORMATION INQUIRY
	BLM0B108	CONTROL DATA INQUIRY
	BLM0B150	PEOPLE DELETE VERIFICATION

BLM0B151	PEOPLE DELETE VERIFICATION
BLM0CU10	CALL DELETE VERIFICATION
BLM0CU11	CALL DELETE VERIFICATION
BLM0S010	CALL SUMMARY DISPLAY
BLM0YE00	PRIMARY OPTIONS MENU
BLM0YE50	PRIMARY OPTIONS MENU
BLM0YU00	RECORD SUMMARY
BLM0YU10	RECORD DELETE VERIFICATION
BLM0YU11	RECORD DELETE VERIFICATION
BLM0YU50	PARENT RECORD SUMMARY
BLM0YU51	PARENT RECORD SUMMARY
BLM0YU60	RECORD DELETE VERIFICATION
BLM0YU61	RECORD DELETE VERIFICATION
BLM0YU80	CHILD RECORD SUMMARY
BLM0YU90	CHILD DELETE VERIFICATION
BLM0YU91	CHILD DELETE VERIFICATION
BLM0Y100	RECORD ENTRY
BLM0Y203	RECORD INQUIRY
BLM0Y251	PANEL SET INQUIRY SUMMARY
BLM0Y261	RECORD CONTROL DATA INQUIRY
BLM0Y300	RECORD SUMMARY DISPLAY
BLM0Y310	PANEL SET DISPLAY
BLM0Y330	RECORD UTILITIES
BLM0Y600	PARENT RECORD ENTRY
BLM0Y650	CHILD RECORD ENTRY
BLM0Y703	PARENT RECORD INQUIRY
BLM0Y713	CHILD RECORD INQUIRY
BLM0Y751	PARENT INQUIRY SUMMARY
BLM0Y761	RECORD CONTROL DATA INQUIRY
BLM0Y771	CHILD INQUIRY SUMMARY
BLM0Y781	PARENT/CHILD INQUIRY SUMMARY
BLM0Y793	RECORD+ INQUIRY
BLM0Y800	RECORD SUMMARY DISPLAY
BLM0Y810	PARENT RECORD DISPLAY
BLM0Y830	RECORD UTILITIES

BLM0Y850	CHILD SUMMARY DISPLAY
BLM0Y860	CHILD RECORD DISPLAY
BLM0Y880	RECORD UTILITIES
BLM2FD6B	DATE/TIME FIELD DISPLAY HELP
BLM6DTDF	DATE/TIME FIELD DISPLAY FORM
BLM6EPPO	DE FIELD PROCESSING ORDER
BLM8CCA0	PANEL SET CREATE
BLM8CC00	PANEL NAME ENTRY
BLM8CC10	PANEL COPY SPECIFICATION
BLM8CD00	PANEL NAME ENTRY
BLM8CF00	FIELD ATTRIBUTES
BLM8CF10	FIELD ATTRIBUTES
BLM8CL10	PANEL NAME ENTRY
BLM8CL20	STRING FIND
BLM8CL30	DICTIONARY PRINT
BLM8CL50	PANEL NAME ENTRY
BLM8CL60	PANEL NAME ENTRY
BLM8CT00	PANEL TEST ENTRY
BLM8CU00	PANEL NAME ENTRY
BLM8CU1A	PANEL FLOW PROCESSING
BLM8CU1B	DATA COLLECTION PROCESSING
BLM8CU1E	PANEL FLOW PROCESSING
BLM8CU1H	DATA COLLECTION PROCESSING
BLM8CU11	SELECTION PANEL UPDATE
BLM8CU12	NULL REPLY SUMMARY
BLM8CU13	SELECTION CONTROL SUMMARY
BLM8CU17	COMMON UPDATE
BLM8CU2A	PANEL FLOW PROCESSING
BLM8CU2B	DATA COLLECTION PROCESSING
BLM8CU2E	PANEL FLOW PROCESSING
BLM8CU2H	DATA COLLECTION PROCESSING
BLM8CU21	OPTION PANEL UPDATE
BLM8CU22	NULL REPLY SUMMARY
BLM8CU23	OPTION CONTROL SUMMARY
BLM8CU27	COMMON UPDATE

BLM8CU31	MESSAGE PANEL UPDATE
BLM8CU37	COMMON UPDATE
BLM8CU43	CONTROL LINE UPDATE
BLM8CU47	COMMON UPDATE
BLM8CU5A	PANEL FLOW UPDATE
BLM8CU5B	STRUCTURED WORD AND VALIDATION COLLECTION
BLM8CU5D	RESPONSE PROCESSING
BLM8CU5E	NULL REPLY UPDATE
BLM8CU5I	PREFIX AND VALIDATION
BLM8CU51	ASSISTED ENTRY UPDATE
BLM8CU53	ASSISTED ENTRY SUMMARY
BLM8CU55	ASSISTED ENTRY SUMMARY
BLM8CU57	COMMON UPDATE
BLM8CU6A	CONTROL FLOW PROCESSING
BLM8CU6B	DATA COLLECTION PROCESSING
BLM8CU61	CONTROL PANEL UPDATE
BLM8CU63	CONTROL LINE SUMMARY
BLM8CU64	TEST DATA PROCESSING UPDATE
BLM8CU67	COMMON UPDATE
BLM8CU7A	PANEL FLOW PROCESSING
BLM8CU7B	DATA COLLECTION PROCESSING
BLM8CU7E	PANEL FLOW PROCESSING
BLM8CU7H	DATA COLLECTION PROCESSING
BLM8CU71	TOP TITLE
BLM8CU72	NULL REPLY SUMMARY
BLM8CU73	FIELD CONTROL SUMMARY
BLM8CU76	TOP TITLE
BLM8CU77	COMMON UPDATE
BLM8CU78	TOP TITLE
BLM8CU79	TOP TITLE
BLM8CU8A	PANEL FLOW UPDATE
BLM8CU8B	COLUMN DISPLAY UPDATE
BLM8CU81	PANEL TITLE
BLM8CU83	COLUMN CONTROL SUMMARY
BLM8CU85	GENERAL UPDATE

BLM8CU87	COMMON UPDATE
BLM8CU9A	ADDDATA SPECIFICATION
BLM8CU9B	BRANCH
BLM8CU9C	CLEAR
BLM8CU9D	LINK
BLM8CU9E	FINDSJRNL SPECIFICATION
BLM8CU9F	FINDSDATA SPECIFICATION
BLM8CU9G	LABEL
BLM8CU9H	PROCESS SPECIFICATION
BLM8CU9I	ISPEXEC SPECIFICATION
BLM8CU9J	TESTFIELD SPECIFICATION
BLM8CU9K	TESTFLOW SPECIFICATION
BLM8CU9L	CONTROL LINE SUMMARY
BLM8CU9M	WORDFIX CONTROL LINE SUMMARY
BLM8CU9N	TRACE SPECIFICATION
BLM8CU9O	MOVEVAR SPECIFICATION
BLM8CU9P	DATA FIELD SPECIFICATION
BLM8CU9Q	FLAG FIELD SPECIFICATION
BLM8CU9R	PRINT
BLM8CU9S	SETFIELD SPECIFICATION
BLM8CU9T	MESSAGE SPECIFICATION
BLM8CU9U	WORDFIX ADD SPECIFICATION
BLM8CU9V	WORDFIX DELETE SPECIFICATION
BLM8CU9W	WORDFIX S-WORD SPECIFICATION
BLM8CU9X	UNFLATTEN
BLM8CU9Y	WORDFIX P-WORD SPECIFICATION
BLM8CU9Z	FLATTEN SPECIFICATION
BLM8CU91	TERMINAL SIMULATOR PANEL UPDATE
BLM8CU97	COMMON UPDATE
BTN0BU00	PROBLEM SUMMARY
BTN0BU10	PROBLEM DELETE VERIFICATION
BTN0BU11	PROBLEM DELETE VERIFICATION
BTN0B100	PROBLEM DATA
BTN0B101	PROBLEM CALLER INFORMATION
BTN0B102	INITIAL PROBLEM DATA

BTN0B200	PROBLEM STATUS ENTRY
BTN0B300	PROBLEM CLOSE ENTRY
BTN0B500	PROBLEM SYNOPSIS ENTRY
BTN0CETX	CHANGE TEXT ENTRY PANEL
BTN0CU00	CHANGE REQUEST SUMMARY
BTN0CU01	CHANGE REQUEST SUMMARY
BTN0C100	CHANGE REQUESTER ENTRY
BTN0C101	INITIAL CHANGE DATA
BTN0C200	CHANGE STATUS ENTRY
BTN0C300	CHANGE CLOSE ENTRY
BTN0DU01	HARDWARE COMPONENT SUMMARY
BTN0DU03	SOFTWARE COMPONENT SUMMARY
BTN0DU08	PERSON RECORD
BTN0DU18	PERSON DELETE VERIFICATION
BTN0DU28	PERSON DELETE VERIFICATION
BTN0DU30	HARDWARE COMPONENT SUMMARY
BTN0DU31	SOFTWARE COMPONENT SUMMARY
BTN0D100	HARDWARE COMPONENT ENTRY
BTN0D102	HARDWARE COMPONENT ENTRY
BTN0D200	SOFTWARE COMPONENT ENTRY
BTN0D500	PERSON RECORD
BTN0ENCC	CHANGE CONTROLLER PRIMARY OPTIONS
BTN0ENCF	CONFIGURATION CONTROLLING PRIMARY OPTIONS
BTN0ENHD	HELP DESK PRIMARY OPTIONS
BTN0ENNO	NETWORK SUPPORT PRIMARY OPTIONS
BTN0ENOM	OPERATIONS MANAGER PRIMARY OPTIONS
BTN0ENPC	PROBLEM CONTROLLER PRIMARY OPTIONS
BTN0ENSU	SUPPORT PRIMARY OPTIONS
BTN0ENSY	SYSTEM ADMINISTRATOR PRIMARY OPTIONS
BTN0EN20	PRIMARY OPTIONS MENU
BTN0EPPC	PRIVILEGE CLASS SELECTION
BTN0E000	QUICK PROBLEM SEARCH PANEL
BTN0E090	PROBLEM INQUIRY SUMMARY
BTN0E190	PROBLEM REPORTER DATA INQUIRY
BTN0E290	PROBLEM STATUS DATA INQUIRY

BTN0E390	PROBLEM CLOSE DATA INQUIRY
BTN0E890	CONTROL DATA INQUIRY
BTN0F000	QUICK CHANGE SEARCH PANEL
BTN0F090	CHANGE REQUEST SUMMARY
BTN0F190	CHANGE REQUESTER INQUIRY
BTN0F290	CHANGE STATUS DATA INQUIRY
BTN0F390	CHANGE CLOSE DATA INQUIRY
BTN0G000	QUICK CONFIG SEARCH PANEL
BTN0G1A1	SOFTWARE COMPONENT INQUIRY
BTN0G1A3	SOFTWARE COMPONENT INQUIRY SUMMARY
BTN0G190	HARDWARE COMPONENT INQUIRY SUMMARY
BTN0G191	HARDWARE COMPONENT INQUIRY
BTN0H000	PERSON SEARCH OPTIONS
BTN0JU00	CLASS SUMMARY
BTN0J100	CLASS DESCRIPTION ENTRY
BTN0J200	AUTHORITY ENTRY
BTN0J300	CLASS USER ENTRY
BTN0L100	PROBLEM REPORTER DISPLAY
BTN0L200	PROBLEM STATUS DISPLAY
BTN0L300	PROBLEM CLOSE DISPLAY
BTN0L500	PROBLEM SYNOPSIS DISPLAY
BTN0M010	CHANGE TEXT DISPLAY
BTN0M100	CHANGE REQUESTER DISPLAY
BTN0M200	CHANGE STATUS DISPLAY
BTN0N100	HARDWARE COMPONENT SUMMARY DISPLAY
BTN0N101	HARDWARE COMPONENT DISPLAY
BTN0N134	PATH DISPLAY SPECIFICATION
BTN0N200	SOFTWARE COMPONENT SUMMARY DISPLAY
BTN0N201	SOFTWARE COMPONENT DISPLAY
BTN0N500	PERSON RECORD
BTN0N510	PERSON RECORD
BTN0Q100	CLASS SUMMARY DISPLAY
BTN0Q200	AUTHORITY DISPLAY
BTN0Q300	CLASS USER DISPLAY
BTN0S010	PROBLEM SUMMARY DISPLAY

BTN0S020	CHANGE SUMMARY DISPLAY
BTN0S021	RECORD UTILITIES
BTN0W020	PROBLEM REPORT ENTRY
BTN0W030	CHANGE REPORT ENTRY
BTN0XAUT	AUTHORITY ENTRY
BTN0XDES	CLASS DESCRIPTION ENTRY
BTN0XHID	CLASS DESCRIPTION ENTRY
BTN0XLPC	Load Privilege Class Records
BTN0XSUM	CLASS SUMMARY
BTN0XUSR	CLASS USER ENTRY
BTN1C126	IIF CONTROL
EYMAM100	MAPPING REFERENCE INQUIRY
EYMBM100	MAPPING REFERENCE DISPLAY
EYMBM500	MAPPING REFERENCE DATA
EYM5MARK	AUTOBRIDGE POSTPROCESSOR
EYM5M100	MAPPING REFERENCE ENTRY
EYM5M110	MAPPING REFERENCE UPDATE
EYM5M200	MAPPING REFERENCE DELETE VERIFICATION
EYM5M250	MAPPING REFERENCE DELETE VERIFICATION
EYM5M500	MAPPING REFERENCE DATA

Tivoli Information Management for z/OS Panels Arranged by Panel Title

BLGLABND	ABEND CODE ENTRY
BLG6APSL	ACTION APPROVE/REJECT
BLG0F283	ACTIVITY CLOSE DATA INQUIRY
BLG0M630	ACTIVITY CLOSE DISPLAY
BLG0C630	ACTIVITY CLOSE ENTRY
BLG0F288	ACTIVITY CONTROL DATA INQUIRY
BLG0CU30	ACTIVITY DELETE VERIFICATION
BLG0CU31	ACTIVITY DELETE VERIFICATION
BLG0C600	ACTIVITY REQUESTER ENTRY
BLG0F281	ACTIVITY REQUESTER DATA INQUIRY
BLG0M600	ACTIVITY REQUESTER DISPLAY
BLG0C620	ACTIVITY STATUS ENTRY
BLG0F282	ACTIVITY STATUS DATA INQUIRY

	BLG0M620	ACTIVITY STATUS DISPLAY
	BLG0CU21	ACTIVITY SUMMARY
	BLG0F280	ACTIVITY SUMMARY
	BLG0S060	ACTIVITY SUMMARY DISPLAY
|	**BLH1S013**	ADD PROBLEM DATA TO SOLUTION CONTROL
	BLM8CU9A	ADDDATA SPECIFICATION
	BLG0J295	ADMINISTRATION AUTHORITY
	BLGLMNAM	AFFECTED DEVICE ENTRY
	BLGLMNA1	AFFECTED DEVICE ENTRY
	BLGLMNA2	AFFECTED DEVICE ENTRY
	BLGLMNA3	AFFECTED DEVICE ENTRY
	BLGLMNA4	AFFECTED DEVICE ENTRY
	BLGLMNA5	AFFECTED DEVICE ENTRY
	BLGLALSD	ALIAS NAME DISPLAY
	BLGLALIS	ALIAS NAME ENTRY
	BLG0GU10	ALIAS RECORD DELETE VERIFICATION
	BLG0GU11	ALIAS RECORD DELETE VERIFICATION
	BLG0G337	ALL FINANCIAL+ INQUIRY
	BLG0G338	ALL FINANCIAL+ SUMMARY
|	**BLG6DTYP**	ALTERNATE PROBLEM
	BLG0I0E0	APAR DISPLAY
	BLGLAPNU	APAR NUMBER ENTRY
	BLGLAPN1	APAR NUMBER ENTRY
	BLGLAPN2	APAR NUMBER ENTRY
	BLGLAPN3	APAR NUMBER ENTRY
	BLGLAPN4	APAR NUMBER ENTRY
	BLGLAPN5	APAR NUMBER ENTRY
|	**BLG1APST**	APPROVAL STATUS DISPLAY CONTROL
|	**BLG6APVR**	APPROVER
|	**BLG6APST**	APPROVER STATUS
	BLM8CU53	ASSISTED ENTRY SUMMARY
	BLM8CU55	ASSISTED ENTRY SUMMARY
	BLM8CU51	ASSISTED ENTRY UPDATE
	BLH0I010	ASSOCIATED DATA MENU
	BTN0Q200	AUTHORITY DISPLAY

BTN0J200	AUTHORITY ENTRY
BTN0XAUT	AUTHORITY ENTRY
BLG0Q213	AUTHORITY SUMMARY
BLG0Q214	AUTHORITY SUMMARY
BLG0Q215	AUTHORITY SUMMARY
BLG0Q216	AUTHORITY SUMMARY
BLG0VE62	AUTHORIZATION CODE INQUIRY
BLG0VQ62	AUTHORIZATION CODE DISPLAY
BLG0V610	AUTHORIZATION CODE ENTRY
BLGLJSPL	AUTHORIZED LOGICAL PARTITION IDENTIFIER ENTRY
BLGLQSPL	AUTHORIZED LOGICAL PARTITION IDENTIFIER DISPLAY
EYM5MARK	AUTOBRIDGE POSTPROCESSOR
BLG0SU50	AUTOMATIC LOG SAVE SUMMARY DISPLAY
BLG0S500	AUTOMATIC LOG SAVE DATA SET INFORMATION DISPLAY
BLG0S510	AUTOMATIC LOG SAVE SYNCHRONIZE AND FILTER DISPLAY
BLG0ZU50	AUTOMATIC LOG SAVE RECORD SUMMARY
BLG0Z500	AUTOMATIC LOG SAVE DATA SET INFORMATION ENTRY
BLG0Z510	AUTOMATIC LOG SAVE SYNCHRONIZE AND FILTER ENTRY
BLGLBONU	BILL OF MATERIAL ENTRY
BLGLBON1	BILL OF MATERIAL ENTRY
BLGLBON2	BILL OF MATERIAL ENTRY
BLGLBON3	BILL OF MATERIAL ENTRY
BLGLBON4	BILL OF MATERIAL ENTRY
BLGLBON5	BILL OF MATERIAL ENTRY
BLM8CU9B	BRANCH
BLG0W510	BROWSE/PRINT
BLM0CU10	CALL DELETE VERIFICATION
BLM0CU11	CALL DELETE VERIFICATION
BLM0S010	CALL SUMMARY DISPLAY
BLGLCRDN	CARD NAME ENTRY
BLG0DU18	CENTER DELETE VERIFICATION
BLG0DU28	CENTER DELETE VERIFICATION
BLG0DU71	CENTER RENAME VERIFICATION
BLG0C015	CHANGE APPROVER DATA MENU
BLG0C500	CHANGE APPROVER ENTRY

BLG0F590	CHANGE APPROVER DATA INQUIRY
BLG0M500	CHANGE APPROVER DISPLAY
BLG0J220	CHANGE AUTHORITY
BLG0Q220	CHANGE AUTHORITY
BLG0F390	CHANGE CLOSE DATA INQUIRY
BTN0F390	CHANGE CLOSE DATA INQUIRY
BLG0M300	CHANGE CLOSE DISPLAY
BLG0C300	CHANGE CLOSE ENTRY
BTN0C300	CHANGE CLOSE ENTRY
BLG0F890	CHANGE CONTROL DATA INQUIRY
BTN0ENCC	CHANGE CONTROLLER PRIMARY OPTIONS
BLG0CU10	CHANGE DELETE VERIFICATION
BLG0CU11	CHANGE DELETE VERIFICATION
BTN0W030	CHANGE REPORT ENTRY
BLG0CU00	CHANGE REQUEST SUMMARY
BLG0CU01	CHANGE REQUEST SUMMARY
BLG0F090	CHANGE REQUEST SUMMARY
BTN0CU00	CHANGE REQUEST SUMMARY
BTN0CU01	CHANGE REQUEST SUMMARY
BTN0F090	CHANGE REQUEST SUMMARY
BLG0C100	CHANGE REQUESTER ENTRY
BLG0F190	CHANGE REQUESTER INQUIRY
BLG0M100	CHANGE REQUESTER DISPLAY
BTN0C100	CHANGE REQUESTER ENTRY
BTN0F190	CHANGE REQUESTER INQUIRY
BTN0M100	CHANGE REQUESTER DISPLAY
BLG0C700	CHANGE REVIEWER ENTRY
BLG0M700	CHANGE REVIEWER DISPLAY
BLG0F290	CHANGE STATUS DATA INQUIRY
BTN0F290	CHANGE STATUS DATA INQUIRY
BLG0M200	CHANGE STATUS DISPLAY
BTN0M200	CHANGE STATUS DISPLAY
BLG0C200	CHANGE STATUS ENTRY
BTN0C200	CHANGE STATUS ENTRY
BLG0S020	CHANGE SUMMARY DISPLAY

BTN0S020	CHANGE SUMMARY DISPLAY
BTN0M010	CHANGE TEXT DISPLAY
BTN0CETX	CHANGE TEXT ENTRY PANEL
BLG0F270	CHANGE/ACTIVITY REQUEST SUMMARY
BLG0F271	CHANGE/ACTIVITY REQUESTER DATA INQUIRY
BLG0F272	CHANGE/ACTIVITY STATUS DATA INQUIRY
BLG0F273	CHANGE/ACTIVITY CLOSE DATA INQUIRY
BLG0F278	CHANGE/ACTIVITY CONTROL DATA INQUIRY
BLG0Y200	CHECK IN
BLG6DVCH	CHILD DATA VIEW ASSISTED ENTRY
BLM0YU90	CHILD DELETE VERIFICATION
BLM0YU91	CHILD DELETE VERIFICATION
BLM0Y771	CHILD INQUIRY SUMMARY
BLM0Y860	CHILD RECORD DISPLAY
BLM0Y650	CHILD RECORD ENTRY
BLM0Y713	CHILD RECORD INQUIRY
BLM0YU80	CHILD RECORD SUMMARY
BLM0Y850	CHILD SUMMARY DISPLAY
BLG0J260	CLASS AUTHORITY
BLG0Q260	CLASS AUTHORITY
BLG0JU10	CLASS DELETE VERIFICATION
BLG0JU11	CLASS DELETE VERIFICATION
BLG0J100	CLASS DESCRIPTION ENTRY
BLG0K190	CLASS DESCRIPTION INQUIRY
BTN0J100	CLASS DESCRIPTION ENTRY
BTN0XDES	CLASS DESCRIPTION ENTRY
BTN0XHID	CLASS DESCRIPTION ENTRY
BLG0JU00	CLASS SUMMARY
BLG0K091	CLASS SUMMARY
BTN0JU00	CLASS SUMMARY
BTN0XSUM	CLASS SUMMARY
BLG0Q100	CLASS SUMMARY DISPLAY
BTN0Q100	CLASS SUMMARY DISPLAY
BTN0Q300	CLASS USER DISPLAY
BLG0J300	CLASS USER ENTRY

BTN0J300	CLASS USER ENTRY
BTN0XUSR	CLASS USER ENTRY
BLM8CU9C	CLEAR
BLHLI124	CO-PROCESSORS
BLH0I224	CO-PROCESSORS QUERY
BLM8CU83	COLUMN CONTROL SUMMARY
BLM8CU8B	COLUMN DISPLAY UPDATE
BLGLCMDD	Command Alias/Authorization Display
BLGLCMDU	Command Alias/Authorization Entry
BLG0HU10	COMMAND RECORD DELETE VERIFICATION
BLG0HU11	COMMAND RECORD DELETE VERIFICATION
BLM8CU17	COMMON UPDATE
BLM8CU27	COMMON UPDATE
BLM8CU37	COMMON UPDATE
BLM8CU47	COMMON UPDATE
BLM8CU57	COMMON UPDATE
BLM8CU67	COMMON UPDATE
BLM8CU77	COMMON UPDATE
BLM8CU87	COMMON UPDATE
BLM8CU97	COMMON UPDATE
BLH0I120	COMPUTER SYSTEM
BLH0I220	COMPUTER SYSTEM QUERY
BLG0J230	CONFIGURATION AUTHORITY
BLG0Q230	CONFIGURATION AUTHORITY
BTN0ENCF	CONFIGURATION CONTROLLER PRIMARY OPTIONS
BLG1UT07	CONFIGURATION DIAGRAM PARAMETER ENTRY
BLG1UT31	CONFIGURATION DIAGRAM INPUT DDNAME ENTRY
BLG1UT32	CONFIGURATION DIAGRAM INPUT DATA SET ENTRY
BLHLI130	CONFIGURATION FILES
BLH0I230	CONFIGURATION FILES QUERY
BLM0B002	CONTACT INFORMATION
BLM0B052	CONTACT INFORMATION DISPLAY
BLM0B102	CONTACT INFORMATION INQUIRY
BLG0E890	CONTROL DATA INQUIRY
BLG0VE75	CONTROL DATA INQUIRY

	BLM0B108	CONTROL DATA INQUIRY
	BTN0E890	CONTROL DATA INQUIRY
	BLM8CU6A	CONTROL FLOW PROCESSING
	BLG0G890	CONTROL INQUIRY
	BLM8CU63	CONTROL LINE SUMMARY
	BLM8CU9L	CONTROL LINE SUMMARY
	BLM8CU43	CONTROL LINE UPDATE
	BLM8CU61	CONTROL PANEL UPDATE
	BLG0G893	CONTROL SUMMARY
|	**BLH1N013**	CREATE CHANGE INDEX RECORD CONTROL
|	**BLGRFCHG**	CREATE INDEX TSP
|	**BLGRFPRO**	CREATE INDEX TSP
|	**BLGRFSOL**	CREATE INDEX TSP
|	**BLH1N012**	CREATE PROBLEM INDEX RECORD CONTROL
|	**BLH1N011**	CREATE SOLUTION INDEX RECORD CONTROL
|	**BLG1A11B**	CREATE SOLUTION RECORD
	BLG0VE70	DATA ATTRIBUTE INQUIRY SUMMARY
	BLGLDATT	DATA ATTRIBUTE RECORDS ENTRY
	BLG0DATT	DATA ATTRIBUTE RECORDS DISPLAY
	BLG0VD70	DATA ATTRIBUTE RECORD DELETE VERIFICATION
	BLG0VD71	DATA ATTRIBUTE RECORD DELETE VERIFICATION
	BLG0VE71	DATA ATTRIBUTE RECORD INQUIRY
	BLG0VQ71	DATA ATTRIBUTE RECORD DISPLAY
	BLG0V700	DATA ATTRIBUTE RECORD ENTRY
	BLG0VQ70	DATA ATTRIBUTE SUMMARY
	BLG0VU70	DATA ATTRIBUTE SUMMARY
	BLG0D500	DATA CENTER ENTRY
	BLG0G590	DATA CENTER INQUIRY
	BLG0N510	DATA CENTER RECORD UTILITIES
	BLG0DU08	DATA CENTER SUMMARY
	BLG0G593	DATA CENTER SUMMARY
	BLG0N500	DATA CENTER SUMMARY DISPLAY
	BLM8CU1B	DATA COLLECTION PROCESSING
	BLM8CU1H	DATA COLLECTION PROCESSING
	BLM8CU2B	DATA COLLECTION PROCESSING

	BLM8CU2H	DATA COLLECTION PROCESSING
	BLM8CU6B	DATA COLLECTION PROCESSING
	BLM8CU7B	DATA COLLECTION PROCESSING
	BLM8CU7H	DATA COLLECTION PROCESSING
	BLG0P700	DATA DEFINITION DEFAULTS
	BLM8CU9P	DATA FIELD SPECIFICATION
|	**BLGLVLSB**	DATA MODEL RECORD
|	**BLG1VU51**	DATA MODEL RECORD CONTROL
|	**BLG1VU71**	DATA MODEL RECORD CONTROL
|	**BLG6TBLD**	DATA VIEW DESKTOP TABLE SUPPORT
|	**BLG2TSTD**	DATA VIEW DESKTOP TABLE SUPPORT HELP
|	**BLG2TSTL**	DATA VIEW DESKTOP TABLE SUPPORT HELP
	BLG0VE60	DATA VIEW INQUIRY SUMMARY
	BLG0VD60	DATA VIEW RECORD DELETE VERIFICATION
	BLG0VD61	DATA VIEW RECORD DELETE VERIFICATION
	BLG0VE61	DATA VIEW RECORD INQUIRY
	BLG0VQ61	DATA VIEW RECORD DISPLAY
	BLG0V600	DATA VIEW RECORD ENTRY
	BLG0VQ60	DATA VIEW SUMMARY DISPLAY
	BLG0VU60	DATA VIEW SUMMARY
|	**BLG6TBLP**	DATA VIEW TABLE SUPPORT
|	**BLG6TBNM**	DATA VIEW TABLE SUPPORT
|	**BLG6TBSI**	DATA VIEW TABLE SUPPORT
|	**BLG6TBSW**	DATA VIEW TABLE SUPPORT
|	**BLG1VU61**	DATA VIEW TABLE SUPPORT CONTROL
|	**BLG2TBDD**	DATA VIEW TABLE SUPPORT HELP
|	**BLG2TBDL**	DATA VIEW TABLE SUPPORT HELP
|	**BLM2FD6B**	DATE/TIME FIELD DISPLAY HELP
|	**BLG6DTDF**	DATE/TIME FIELD DISPLAY FORM
|	**BLG2DTTM**	DATETIME RECORD HELP PANEL
	BLG0Z530	DB2 EXTRACT INFORMATION ENTRY
|	**BLM6EFPO**	DE FIELD PROCESSING ORDER
	BLG0DFSM	DEFAULT FILE SUMMARY PANEL
|	**BLH1N004**	DELETE INDEX RECORD OWNERSHIP CONTROL
	BLGLFGID	DESKTOP FIELD GROUP DISPLAY

	BLGLFGIN	DESKTOP FIELD GROUP ENTRY
|	**BLGLTSTD**	DESKTOP LAYOUT TABLE SUPPORT
|	**BLGLTSTK**	DESKTOP LAYOUT TABLE SUPPORT
|	**BLGLTSTL**	DESKTOP LAYOUT TABLE SUPPORT
|	**BLG6TBAL**	DESKTOP LAYOUT TABLE SUPPORT
	BLGLTSFD	DESKTOP PANEL FIELD LIST DISPLAY
	BLGLTSFK	DESKTOP PANEL FIELD LIST ENTRY
	BLGLTSFL	DESKTOP PANEL FIELD LIST ENTRY
	BLGLTSID	DESKTOP PANEL LAYOUT DISPLAY
	BLGLTSIN	DESKTOP PANEL LAYOUT DEFINITION
	BLGLIPID	DEVICE ITEM ENTRY
	BLGLIPI1	DEVICE ITEM ENTRY
	BLGLIPI2	DEVICE ITEM ENTRY
	BLGLIPI3	DEVICE ITEM ENTRY
	BLGLIPI4	DEVICE ITEM ENTRY
	BLGLIPI5	DEVICE ITEM ENTRY
	BLGLDEVL	DEVICE NAME ENTRY
	BLGLDVNA	DEVICE NAME ENTRY
	BLGLDVN1	DEVICE NAME ENTRY
	BLGLDVN2	DEVICE NAME ENTRY
	BLGLDVN3	DEVICE NAME ENTRY
	BLGLDVN4	DEVICE NAME ENTRY
	BLGLDVN5	DEVICE NAME ENTRY
	BLGLDEVF	DEVICE TYPE ENTRY
	BLGLDTYP	DEVICE TYPE ENTRY
	BLM8CL30	DICTIONARY PRINT
|	**BLH2X111**	DISPLAY ADDRESSES HELP
|	**BLH0X111**	DISPLAY ADDRESSES SELECTION
|	**BLHLX1BC**	DISPLAY BCC ADDRESSES
|	**BLHLX1CC**	DISPLAY CC ADDRESSES
|	**BLG1AAPD**	DISPLAY CHANGE APPROVER CONTROL
|	**BLGLAPST**	DISPLAY CHANGE APPROVER STATUS
|	**BLGLREVD**	DISPLAY CHANGE REVIEWER
|	**BLG1ARED**	DISPLAY CHANGE REVIEWER CONTROL
|	**BLHLX1TO**	DISPLAY TO ADDRESSES

	BLG0C071	DOCUMENT ADD
	BLG0C171	DOCUMENT ADD INQUIRY
	BLG0C072	DOCUMENT ALTER
	BLG0C172	DOCUMENT ALTER INQUIRY
	BLGLDOCN	DOCUMENT NAME ENTRY
	BLGLDOC1	DOCUMENT NAME ENTRY
	BLGLDOC2	DOCUMENT NAME ENTRY
	BLGLDNUM	DOCUMENT NUMBER ENTRY
	BLGLDNU1	DOCUMENT NUMBER ENTRY
	BLGLDNU2	DOCUMENT NUMBER ENTRY
	BLG0C073	DOCUMENT REMOVE
	BLG0C173	DOCUMENT REMOVE INQUIRY
	BLG0P820	DRAW REPORT DATA SET DESTINATION ENTRY
	BLG0P821	DRAW REPORT DATA SET DESTINATION ENTRY
	BLG0P830	DRAW REPORT DDNAME DESTINATION ENTRY
	BLG0P831	DRAW REPORT DDNAME DESTINATION ENTRY
	BLG0P810	DRAW REPORT SYSOUT DESTINATION ENTRY
	BLG0P811	DRAW REPORT SYSOUT DESTINATION ENTRY
|	**BLGLAPVR**	EDIT CHANGE APPROVER
|	**BLGLREVR**	EDIT CHANGE REVIEWER
|	**BLG1AELD**	ELIGIBLE USER CONTROL
|	**BLG1AELU**	ELIGIBLE USER CONTROL
	BLG0Q300	ELIGIBLE USER DISPLAY
|	**BLGLJ300**	ELIGIBLE USERS
|	**BLGLQ300**	ELIGIBLE USERS
	BLGLREAN	ENGINEER ACTION ENTRY
	BLGLREA1	ENGINEER ACTION ENTRY
	BLGLREA2	ENGINEER ACTION ENTRY
	BLGLREA3	ENGINEER ACTION ENTRY
	BLGLREA4	ENGINEER ACTION ENTRY
	BLGLREA5	ENGINEER ACTION ENTRY
	BLGLECNU	ENGINEER ANNOUNCEMENT ENTRY
	BLGLECN1	ENGINEER ANNOUNCEMENT ENTRY
	BLGLECN2	ENGINEER ANNOUNCEMENT ENTRY
	BLGLECN3	ENGINEER ANNOUNCEMENT ENTRY

BLGLECN4	ENGINEER ANNOUNCEMENT ENTRY
BLGLECN5	ENGINEER ANNOUNCEMENT ENTRY
BLGLECID	ENGINEER CHANGE ENTRY
BLGLECI1	ENGINEER CHANGE ENTRY
BLGLECI2	ENGINEER CHANGE ENTRY
BLGLECI3	ENGINEER CHANGE ENTRY
BLGLECI4	ENGINEER CHANGE ENTRY
BLGLECI5	ENGINEER CHANGE ENTRY
BLGLENVF	ENVIRONMENT CONDITION ENTRY
BLGLENVS	ENVIRONMENT CONDITION ENTRY
BLGLERCE	ERROR CODE ENTRY
BLG0I200	ESCALATION CONTROL INQUIRY
BLG0IU00	ESCALATION RULES INQUIRY SUMMARY
BLG0I100	ESCALATION RULES INQUIRY
BLG0J250	ESCALATION RULES AUTHORITY
BLG0Q250	ESCALATION RULES AUTHORITY
BLG0RU00	ESCALATION RULES DISPLAY
BLG0R010	ESCALATION RULES DELETE VERIFICATION
BLG0R011	ESCALATION RULES DELETE VERIFICATION
BLG0R100	ESCALATION RULES DISPLAY
BLG0ZU00	ESCALATION RULES SUMMARY
BLG0Z100	ESCALATION RULES ENTRY
BLGLSYMC	EXTERNAL SYMPTOM ENTRY
BLG0W250	EXTRACT DIAGRAM DATA SPECIFICATION
BLGLDFEA	FEATURE NAME ENTRY
BLM8CF00	FIELD ATTRIBUTES
BLM8CF10	FIELD ATTRIBUTES
BLM8CU73	FIELD CONTROL SUMMARY
BLGLFGFD	FIELD GROUP FIELD LIST DISPLAY
BLGLFGFK	FIELD GROUP FIELD LIST ENTRY
BLGLFGFL	FIELD GROUP FIELD LIST ENTRY
BLG0FGFL	FIELD GROUP FIELD LIST SELECTION
BLH0X015	FILE MESSAGE RECORD
BLGLFNAM	FILES AFFECTED ENTRY
BLGLFNA1	FILES AFFECTED ENTRY

BLGLFNA2	FILES AFFECTED ENTRY
BLGLFNA3	FILES AFFECTED ENTRY
BLGLFNA4	FILES AFFECTED ENTRY
BLGLFNA5	FILES AFFECTED ENTRY
BLG0T120	FINAL RESPONSES DISPLAY
BLG0U120	FINAL SRC RESPONSE ENTRY
BLG0J240	FINANCIAL AUTHORITY
BLG0Q240	FINANCIAL AUTHORITY
BLM8CU9F	FINDSDATA SPECIFICATION
BLM8CU9E	FINDSJRNL SPECIFICATION
BLM8CU9Q	FLAG FIELD SPECIFICATION
BLM8CU9Z	FLATTEN SPECIFICATION
BLGLFMIE	FMID ENTRY
BLM8CU85	GENERAL UPDATE
BLGLCNAM	GENERIC CARD ENTRY
BLGLCNA1	GENERIC CARD ENTRY
BLGLCNA2	GENERIC CARD ENTRY
BLGLCNA3	GENERIC CARD ENTRY
BLGLCNA4	GENERIC CARD ENTRY
BLGLCNA5	GENERIC CARD ENTRY
BLH1X006	GENERIC TEXT COPY CONTROL
BLH1X009	GENERIC TEXT DELETE CONTROL
BLH1X012	GENERIC TEXT DELETE CONTROL
BLH0X090	GENERIC TEXT DELETE VERIFY
BLH0X110	GENERIC TEXT DISPLAY SUMMARY
BLH2X110	GENERIC TEXT DISPLAY SUMMARY HELP
BLH0X010	GENERIC TEXT ENTRY
BLH1X040	GENERIC TEXT FILE CONTROL
BLH0X091	GENERIC TEXT MULTIPLE DELETE
BLH1X004	GENERIC TEXT OWNERSHIP CONTROL
BLH1X000	GENERIC TEXT RECORD ACCESS CONTROL
BLH1TMTT	GENERIC TEXT TEXT ENTRY
BLG0X100	GLOSSARY
BLH0I225	HARD DISK QUERY
BLHLI125	HARD DISKS

BLG0C061	HARDWARE ADD
BLG0C161	HARDWARE ADD INQUIRY
BLG0C062	HARDWARE ALTER
BLG0C063	HARDWARE ALTER BY INSTALLING
BLG0C064	HARDWARE ALTER BY MODIFYING
BLG0C065	HARDWARE ALTER BY REMOVING
BLG0C163	HARDWARE ALTER BY INSTALLING INQUIRY
BLG0C164	HARDWARE ALTER BY MODIFYING INQUIRY
BLG0C165	HARDWARE ALTER BY REMOVING INQUIRY
BLG0C162	HARDWARE ALTER INQUIRY
BLG0DU01	HARDWARE COMPONENT SUMMARY
BLG0DU11	HARDWARE COMPONENT DELETE VERIFICATION
BLG0DU21	HARDWARE COMPONENT DELETE VERIFICATION
BLG0DU30	HARDWARE COMPONENT SUMMARY
BLG0DU57	HARDWARE COMPONENT RENAME VERIFICATION
BLG0DU74	HARDWARE COMPONENT SWAP VERIFICATION
BLG0D100	HARDWARE COMPONENT ENTRY
BLG0D102	HARDWARE COMPONENT ENTRY
BLG0D105	HARDWARE COMPONENT SOURCE DEFINITION
BLG0D113	HARDWARE COMPONENT DIAGRAM MAP DATA ENTRY
BLG0D150	HARDWARE COMPONENT PRIMARY SUPPORT ENTRY
BLG0D151	HARDWARE COMPONENT SECONDARY SUPPORT ENTRY
BLG0D152	HARDWARE COMPONENT FINANCIAL ENTRY
BLG0D153	HARDWARE COMPONENT FINANCIAL ENTRY
BLG0D170	HARDWARE COMPONENT LINE ENTRY
BLG0D172	HARDWARE COMPONENT LOOP ENTRY
BLG0G190	HARDWARE COMPONENT INQUIRY SUMMARY
BLG0N100	HARDWARE COMPONENT SUMMARY DISPLAY
BLG0N101	HARDWARE COMPONENT DISPLAY
BLG0N105	HARDWARE COMPONENT SOURCE DISPLAY
BLG0N110	HARDWARE COMPONENT RECORD UTILITIES
BLG0N113	HARDWARE COMPONENT DIAGRAM MAP DATA DISPLAY
BLG0N150	HARDWARE COMPONENT PRIMARY SUPPORT DISPLAY
BLG0N151	HARDWARE COMPONENT SECONDARY SUPPORT DISPLAY
BLG0N153	HARDWARE COMPONENT FINANCIAL DISPLAY

BLG0N170	HARDWARE COMPONENT LINE DISPLAY
BLG0N172	HARDWARE COMPONENT LOOP DISPLAY
BTN0DU01	HARDWARE COMPONENT SUMMARY
BTN0DU30	HARDWARE COMPONENT SUMMARY
BTN0D100	HARDWARE COMPONENT ENTRY
BTN0D102	HARDWARE COMPONENT ENTRY
BTN0G190	HARDWARE COMPONENT INQUIRY SUMMARY
BTN0G191	HARDWARE COMPONENT INQUIRY
BTN0N100	HARDWARE COMPONENT SUMMARY DISPLAY
BTN0N101	HARDWARE COMPONENT DISPLAY
BLG0DU32	HARDWARE CONNECTION SUMMARY
BLG0DU34	HARDWARE CONNECTION DELETE VERIFICATION
BLG0DU35	HARDWARE CONNECTION DELETE VERIFICATION
BLG0DU58	HARDWARE CONNECTION RENAME VERIFICATION
BLG0D130	HARDWARE CONNECTION ENTRY
BLG0G1A4	HARDWARE CONNECTION INQUIRY
BLG0G1A5	HARDWARE CONNECTION SUMMARY
BLG0N130	HARDWARE CONNECTION DISPLAY
BLG0N132	HARDWARE CONNECTION DISPLAY
BLG0N133	HARDWARE CONNECTION RECORD UTILITIES
BLGLECLV	HARDWARE EC LEVEL ENTRY
BLG0DU02	HARDWARE FEATURE SUMMARY
BLG0DU12	HARDWARE FEATURE DELETE VERIFICATION
BLG0DU22	HARDWARE FEATURE DELETE VERIFICATION
BLG0DU59	HARDWARE FEATURE RENAME VERIFICATION
BLG0D120	HARDWARE FEATURE ENTRY
BLG0D154	HARDWARE FEATURE FINANCIAL ENTRY
BLG0D157	HARDWARE FEATURE SUPPORT ENTRY
BLG0G182	HARDWARE FEATURE SUPPORT/EC LEVELS/FINANCIAL DATA INQUIRY
BLG0G183	HARDWARE FEATURE SUMMARY
BLG0G184	HARDWARE FEATURE DESCRIPTION/CONTROL DATA INQUIRY
BLG0N120	HARDWARE FEATURE SUMMARY DISPLAY
BLG0N121	HARDWARE FEATURE RECORD UTILITIES
BLG0N122	HARDWARE FEATURE DISPLAY
BLG0N154	HARDWARE FEATURE FINANCIAL DISPLAY

BLG0N157	HARDWARE FEATURE SUPPORT DISPLAY
BLG0DU05	HARDWARE FINANCIAL SUMMARY
BLG0DU15	HARDWARE FINANCIAL DELETE VERIFICATION
BLG0DU25	HARDWARE FINANCIAL DELETE VERIFICATION
BLG0DU61	HARDWARE FINANCIAL RENAME VERIFICATION
BLG0D300	HARDWARE FINANCIAL ENTRY
BLG0D302	HARDWARE FINANCIAL ENTRY
BLG0D305	HARDWARE FINANCIAL - PURCHASE ENTRY
BLG0D306	HARDWARE FINANCIAL - RENTAL/LEASE ENTRY
BLG0D307	HARDWARE FINANCIAL - VPA ENTRY
BLG0G193	HARDWARE FINANCIAL/CONTROL DATA INQUIRY
BLG0G390	HARDWARE FINANCIAL SUMMARY
BLG0G391	HARDWARE FINANCIAL DESCRIPTION DATA INQUIRY
BLG0G392	HARDWARE FINANCIAL SECONDARY DESCRIPTION DATA INQUIRY
BLG0N300	HARDWARE FINANCIAL SUMMARY DISPLAY
BLG0N301	HARDWARE FINANCIAL DISPLAY
BLG0N305	HARDWARE FINANCIAL - PURCHASE DISPLAY
BLG0N306	HARDWARE FINANCIAL - RENTAL/LEASE DISPLAY
BLG0N307	HARDWARE FINANCIAL - VPA DISPLAY
BLG0N310	HARDWARE FINANCIAL RECORD UTILITIES
BLG0N320	HARDWARE FINANCIAL DISPLAY
BLH0IU20	HARDWARE INVENTORY DELETE VERIFICATION
BLH0IU21	HARDWARE INVENTORY DELETE VERIFICATION
BLG0G191	HARDWARE PRIMARY DESCRIPTION/SUPPORT DATA INQUIRY
BLH0I022	HARDWARE QUERY MENU
BLG0G137	HARDWARE RECORDS+
BLG0G138	HARDWARE RECORDS+
BLG0C066	HARDWARE REMOVE
BLG0C166	HARDWARE REMOVE INQUIRY
BLG0G192	HARDWARE SECONDARY DESCRIPTION/SUPPORT DATA INQUIRY
BLG0DU33	HARDWARE SUBCOMPONENT SUMMARY
BLG0DU42	HARDWARE SUBCOMPONENT DELETE VERIFICATION
BLG0DU43	HARDWARE SUBCOMPONENT DELETE VERIFICATION
BLG0DU60	HARDWARE SUBCOMPONENT RENAME VERIFICATION
BLG0DU75	HARDWARE SUBCOMPONENT SWAP VERIFICATION

	BLG0D140	HARDWARE SUBCOMPONENT ENTRY
	BLG0D160	HARDWARE SUBCOMPONENT SUPPORT ENTRY
	BLG0D164	HARDWARE SUBCOMPONENT FINANCIAL ENTRY
	BLG0G170	HARDWARE SUBCOMPONENT SUMMARY
	BLG0G171	HARDWARE SUBCOMPONENT DESCRIPTION/CONTROL DATA INQUIRY
	BLG0G172	HARDWARE SUBCOMPONENT SUPPORT/EC LEVELS/FINANCIAL INQUIRY
	BLG0N140	HARDWARE SUBCOMPONENT SUMMARY DISPLAY
	BLG0N141	HARDWARE SUBCOMPONENT RECORD UTILITIES
	BLG0N142	HARDWARE SUBCOMPONENT DISPLAY
	BLG0N160	HARDWARE SUBCOMPONENT SUPPORT DISPLAY
	BLG0N164	HARDWARE SUBCOMPONENT FINANCIAL DISPLAY
	BTN0ENHD	HELP DESK PRIMARY OPTIONS
|	**BLG2GNRC**	HELP PANEL SKELETON
|	**BLG1TDEH**	HELP TEXT
|	**BLG1TEHD**	HELP TEXT
|	**BTN1C126**	IIF CONTROL
|	**BLH0N000**	INDEX ENTRY OPTIONS MENU
|	**BLH1N000**	INDEX RECORD ACCESS CONTROL
|	**BLH1N010**	INDEX RECORD CREATE CONTROL
|	**BLH1N020**	INDEX RECORD CREATE END PROCESSING CONTROL
|	**BLH1N009**	INDEX RECORD DELETE CONTROL
|	**BLH1N019**	INDEX RECORD DELETE PROCESSING CONTROL
|	**BLH0N001**	INDEX RECORD ENTRY
|	**BLH0N010**	INDEX RECORD ENTRY FILE
|	**BLH1N040**	INDEX RECORD FILE CONTROL
|	**BLH0N200**	INDEX RECORD INQUIRY
|	**BLH1N008**	INDEX RECORD PRINT CONTROL
|	**BLH1N001**	INDEX RECORD PROCESSING CONTROL
|	**BLH0N110**	INDEX RECORD SUMMARY DISPLAY
|	**BLG6URN5**	INDEX RECORD VALIDATION
|	**BLH0N090**	INDEX RECORD VERIFY DELETE
|	**BLH0N095**	INDEX RECORD VERIFY DELETE
|	**BLH0N096**	INDEX RECORD VERIFY MULTIPLE DELETE
	BTN0C101	INITIAL CHANGE DATA

BTN0B102	INITIAL PROBLEM DATA
BLHLI131	INSTALLED SOFTWARE
BLH0I231	INSTALLED SOFTWARE QUERY
BLG0L600	INTERESTED CLASS DISPLAY
BLH0I001	INVENTORY MENU
BLH0I002	INVENTORY MENU
BLHLI128	IPX LAN
BLHLI129	IPX LAN CONNECTIONS
BLH0I229	IPX LAN CONNECTIONS QUERY
BLH0I228	IPX LAN QUERY
BLM8CU9I	ISPEXEC SPECIFICATION
BLGLINUM	ITEM NUMBER ENTRY
BLM8CU9G	LABEL
BLGLLSPE	LINE SPEED ENTRY
BLGLLCRM	LINE/CIRCUIT NUMBER ENTRY
BLGLLTYE	LINE/LOOP TYPE ENTRY
BLM8CU9D	LINK
BTN0XLPC	LOAD PRIVILEGE CLASS RECORDS
BLGLLOCO	LOCATION CODE ENTRY
BLGLLOC1	LOCATION CODE ENTRY
BLGLLOC2	LOCATION CODE ENTRY
BLGLLOC3	LOCATION CODE ENTRY
BLGLLOC4	LOCATION CODE ENTRY
BLGLLOC5	LOCATION CODE ENTRY
BLHLI126	LOGICAL DRIVES
BLH0I226	LOGICAL DRIVES QUERY
BLGLLIDE	LOOP ID ENTRY
BLGLUSER	MAIL INFORMATION ENTRY
BLG0USER	MAIL INFORMATION DISPLAY
BLGLMAUF	MAJOR UNIT ENTRY
BLG0K291	MANAGEMENT CLASS AUTHORITY
EYMAM100	MAPPING REFERENCE INQUIRY
EYMBM100	MAPPING REFERENCE DISPLAY
EYMBM500	MAPPING REFERENCE DATA
EYM5M100	MAPPING REFERENCE ENTRY

EYM5M110	MAPPING REFERENCE UPDATE
EYM5M200	MAPPING REFERENCE DELETE VERIFICATION
EYM5M250	MAPPING REFERENCE DELETE VERIFICATION
EYM5M500	MAPPING REFERENCE DATA
BLGLMSGC	MESSAGE IDENTITY ENTRY
BLM8CU31	MESSAGE PANEL UPDATE
BLM8CU9T	MESSAGE SPECIFICATION
BLGLMIUF	MINOR UNIT ENTRY
BLG0D770	MODEL COMPONENT LINE ENTRY
BLG0D772	MODEL COMPONENT LOOP ENTRY
BLG0G748	MODEL COMPONENT RECORDS+
BLG0G751	MODEL COMPONENT RECORDS+
BLG0G747	MODEL FEATURE RECORDS+
BLG0G749	MODEL FEATURE RECORDS+
BLG0DU36	MODEL HARDWARE COMPONENT SUMMARY
BLG0DU37	MODEL HARDWARE COMPONENT SUMMARY
BLG0DU47	MODEL HARDWARE COMPONENT DELETE VERIFICATION
BLG0DU48	MODEL HARDWARE COMPONENT DELETE VERIFICATION
BLG0DU66	MODEL HARDWARE COMPONENT RENAME VERIFICATION
BLG0D700	MODEL HARDWARE COMPONENT ENTRY
BLG0D703	MODEL HARDWARE COMPONENT ENTRY
BLG0D750	MODEL HARDWARE COMPONENT PRIMARY SUPPORT ENTRY COMPONENT: __
BLG0D751	MODEL HARDWARE COMPONENT SECONDARY SUPPORT ENTRY COMPONENT: _
BLG0D752	MODEL HARDWARE COMPONENT FINANCIAL ENTRY
BLG0D753	MODEL HARDWARE COMPONENT FINANCIAL ENTRY
BLG0G790	MODEL HARDWARE COMPONENT SUMMARY
BLG0N700	MODEL HARDWARE COMPONENT SUMMARY DISPLAY
BLG0N701	MODEL HARDWARE COMPONENT DISPLAY
BLG0N710	MODEL HARDWARE COMPONENT RECORD UTILITIES
BLG0N750	MODEL HARDWARE COMPONENT PRIMARY SUPPORT DISPLAY COMPONENT: _
BLG0N751	MODEL HARDWARE COMPONENT SECONDARY SUPPORT DISPLAY COMPONENT:
BLG0N753	MODEL HARDWARE COMPONENT FINANCIAL DISPLAY
BLG0N770	MODEL HARDWARE COMPONENT LINE DISPLAY

BLG0N772	MODEL HARDWARE COMPONENT LOOP DISPLAY
BLG0G791	MODEL HARDWARE DESCRIPTION DATA INQUIRY
BLG0DU38	MODEL HARDWARE FEATURE SUMMARY
BLG0DU51	MODEL HARDWARE FEATURE DELETE VERIFICATION
BLG0DU54	MODEL HARDWARE FEATURE DELETE VERIFICATION
BLG0DU67	MODEL HARDWARE FEATURE RENAME VERIFICATION
BLG0D720	MODEL HARDWARE FEATURE ENTRY
BLG0D754	MODEL HARDWARE FEATURE FINANCIAL ENTRY
BLG0D757	MODEL HARDWARE FEATURE SUPPORT ENTRY
BLG0G727	MODEL HARDWARE FEATURE INQUIRY
BLG0G728	MODEL HARDWARE FEATURE SUMMARY
BLG0N720	MODEL HARDWARE FEATURE SUMMARY DISPLAY
BLG0N721	MODEL HARDWARE FEATURE RECORD UTILITIES
BLG0N722	MODEL HARDWARE FEATURE DISPLAY
BLG0N754	MODEL HARDWARE FEATURE FINANCIAL DISPLAY
BLG0N757	MODEL HARDWARE FEATURE SUPPORT DISPLAY
BLG0G738	MODEL HARDWARE RECORDS+
BLG0G742	MODEL HARDWARE RECORDS+
BLG0DU39	MODEL HARDWARE SUBCOMPONENT SUMMARY
BLG0DU49	MODEL HARDWARE SUBCOMPONENT DELETE VERIFICATION
BLG0DU50	MODEL HARDWARE SUBCOMPONENT DELETE VERIFICATION
BLG0DU68	MODEL HARDWARE SUBCOMPONENT RENAME VERIFICATION
BLG0D740	MODEL HARDWARE SUBCOMPONENT ENTRY
BLG0D760	MODEL HARDWARE SUBCOMPONENT PRIMARY SUPPORT ENTRY SUBCOMP:
BLG0D764	MODEL HARDWARE SUBCOMPONENT FINANCIAL ENTRY
BLG0G737	MODEL HARDWARE SUBCOMPONENT INQUIRY
BLG0G739	MODEL HARDWARE SUBCOMPONENT SUMMARY
BLG0G792	MODEL HARDWARE SUPPORT/FINANCIAL/CONTROL DATA INQUIRY
BLG0N740	MODEL HARDWARE SUBCOMPONENT DISPLAY
BLG0N741	MODEL HARDWARE SUBCOMPONENT RECORD UTILITIES
BLG0N742	MODEL HARDWARE SUBCOMPONENT DISPLAY
BLG0N760	MODEL HARDWARE SUBCOMPONENT SUPPORT DISPLAY
BLG0N764	MODEL HARDWARE SUBCOMPONENT FINANCIAL DISPLAY
BLG0DU44	MODEL SOFTWARE COMPONENT SUMMARY

A. Panel Lists

BLG0DU45	MODEL SOFTWARE COMPONENT SUMMARY
BLG0DU52	MODEL SOFTWARE COMPONENT DELETE VERIFICATION
BLG0DU53	MODEL SOFTWARE COMPONENT DELETE VERIFICATION
BLG0DU69	MODEL SOFTWARE COMPONENT RENAME VERIFICATION
BLG0D758	MODEL SOFTWARE COMPONENT PRIMARY SUPPORT ENTRY
BLG0D759	MODEL SOFTWARE COMPONENT SECONDARY SUPPORT ENTRY COMPONENT: _
BLG0D780	MODEL SOFTWARE COMPONENT ENTRY
BLG0D783	MODEL SOFTWARE COMPONENT FINANCIAL ENTRY
BLG0G770	MODEL SOFTWARE COMPONENT SUMMARY
BLG0G771	MODEL SOFTWARE COMPONENT DESCRIPTION/SUPPORT DATA INQUIRY
BLG0G772	MODEL SOFTWARE COMPONENT OTHER DATA INQUIRY
BLG0N758	MODEL SOFTWARE COMPONENT PRIMARY SUPPORT DISPLAY COMPONENT: _
BLG0N759	MODEL SOFTWARE COMPONENT SECONDARY SUPPORT DISPLAY COMPONENT:
BLG0N780	MODEL SOFTWARE COMPONENT SUMMARY DISPLAY
BLG0N781	MODEL SOFTWARE COMPONENT RECORD UTILITIES
BLG0N782	MODEL SOFTWARE COMPONENT DISPLAY
BLG0N783	MODEL SOFTWARE COMPONENT FINANCIAL DISPLAY
BLG0DU46	MODEL SOFTWARE FEATURE SUMMARY
BLG0DU55	MODEL SOFTWARE FEATURE DELETE VERIFICATION
BLG0DU56	MODEL SOFTWARE FEATURE DELETE VERIFICATION
BLG0DU70	MODEL SOFTWARE FEATURE RENAME VERIFICATION
BLG0D784	MODEL SOFTWARE FEATURE FINANCIAL ENTRY
BLG0D787	MODEL SOFTWARE FEATURE SUPPORT ENTRY
BLG0D790	MODEL SOFTWARE FEATURE ENTRY
BLG0G780	MODEL SOFTWARE FEATURE SUMMARY
BLG0G781	MODEL SOFTWARE FEATURE DESCRIPTION DATA INQUIRY
BLG0G782	MODEL SOFTWARE FEATURE SUPPORT/FINANCIAL DATA INQUIRY
BLG0N784	MODEL SOFTWARE FEATURE FINANCIAL
BLG0N787	MODEL SOFTWARE FEATURE SUPPORT DISPLAY
BLG0N790	MODEL SOFTWARE FEATURE SUMMARY DISPLAY
BLG0N791	MODEL SOFTWARE FEATURE RECORD UTILITIES
BLG0N792	MODEL SOFTWARE FEATURE DISPLAY
BLG0G787	MODEL SOFTWARE RECORDS+

	BLG0G788	MODEL SOFTWARE RECORDS+
	BLM8CU9O	MOVEVAR SPECIFICATION
	BLH0I12A	NETWARE SERVER
	BLH0I22A	NETWARE SERVER QUERY
	BLHLI127	NETWORK NODES
	BLH0I227	NETWORK NODES QUERY
	BTN0ENNO	NETWORK SUPPORT PRIMARY OPTIONS
	BLH0I132	NT INFORMATION
	BLH0I232	NT INFORMATION QUERY
	BLM8CU12	NULL REPLY SUMMARY
	BLM8CU22	NULL REPLY SUMMARY
	BLM8CU72	NULL REPLY SUMMARY
	BLM8CU5E	NULL REPLY UPDATE
|	**BLGESC14**	OLD ESCALATION TSP
|	**BLGESC16**	OLD ESCALATION TSP
|	**BLGTSPXX**	OLD ESCALATION TSP
|	**BLG2ZSC7**	ONLINE OVERVIEW HELP
	BLGLOPCF	OPERATION CODE ENTRY
	BLGLOPCS	OPERATION CODE ENTRY
	BTN0ENOM	OPERATIONS MANAGER PRIMARY OPTIONS
	BLM8CU23	OPTION CONTROL SUMMARY
	BLM8CU21	OPTION PANEL UPDATE
	BLGLFEAT	OPTIONAL FEATURE ENTRY
	BLGLFEA1	OPTIONAL FEATURE ENTRY
	BLGLFEA2	OPTIONAL FEATURE ENTRY
	BLGLFEA3	OPTIONAL FEATURE ENTRY
	BLGLFEA4	OPTIONAL FEATURE ENTRY
	BLGLFEA5	OPTIONAL FEATURE ENTRY
	BLM8CC10	PANEL COPY SPECIFICATION
	BLG0VQ74	PANEL FLOW DISPLAY
	BLG0VE74	PANEL FLOW INQUIRY
	BLM8CU1A	PANEL FLOW PROCESSING
	BLM8CU1E	PANEL FLOW PROCESSING
	BLM8CU2A	PANEL FLOW PROCESSING
	BLM8CU2E	PANEL FLOW PROCESSING

BLM8CU7A	PANEL FLOW PROCESSING
BLM8CU7E	PANEL FLOW PROCESSING
BLG0V730	PANEL FLOW UPDATE
BLM8CU5A	PANEL FLOW UPDATE
BLM8CU8A	PANEL FLOW UPDATE
BLM8CC00	PANEL NAME ENTRY
BLM8CD00	PANEL NAME ENTRY
BLM8CL10	PANEL NAME ENTRY
BLM8CL50	PANEL NAME ENTRY
BLM8CL60	PANEL NAME ENTRY
BLM8CU00	PANEL NAME ENTRY
BLM8CCA0	PANEL SET CREATE
BLM0Y310	PANEL SET DISPLAY
BLM0Y251	PANEL SET INQUIRY SUMMARY
BLM8CT00	PANEL TEST ENTRY
BLM8CU81	PANEL TITLE
BLM0Y751	PARENT INQUIRY SUMMARY
BLM0Y810	PARENT RECORD DISPLAY
BLM0Y600	PARENT RECORD ENTRY
BLM0Y703	PARENT RECORD INQUIRY
BLM0YU50	PARENT RECORD SUMMARY
BLM0YU51	PARENT RECORD SUMMARY
BLM0Y781	PARENT/CHILD INQUIRY SUMMARY
BLG0N134	PATH DISPLAY SPECIFICATION
BTN0N134	PATH DISPLAY SPECIFICATION
BLH0I011	PC ASSOCIATED DATA
BLH0I121	PC BIOS
BLH0I221	PC BIOS QUERY
BLH0I123	PC MEMORY
BLH0I223	PC MEMORY QUERY
BLG0J280	PEOPLE AUTHORITY
BLG0Q280	PEOPLE AUTHORITY
BLM0B150	PEOPLE DELETE VERIFICATION
BLM0B151	PEOPLE DELETE VERIFICATION
BLM0B001	PEOPLE INFORMATION

BLM0B051	PEOPLE INFORMATION DISPLAY
BLM0B101	PEOPLE INFORMATION INQUIRY
BLM0B100	PEOPLE INQUIRY SUMMARY
BLM0B060	PEOPLE RECORD UTILITIES
BLM0B000	PEOPLE SUMMARY
BLM0B050	PEOPLE SUMMARY DISPLAY
BLG0W200	PERIODIC CHANGE STATUS REPORT
BLG0W210	PERIODIC PROBLEM STATUS REPORT
BTN0DU18	PERSON DELETE VERIFICATION
BTN0DU28	PERSON DELETE VERIFICATION
BTN0DU08	PERSON RECORD
BTN0D500	PERSON RECORD
BTN0N500	PERSON RECORD
BTN0N510	PERSON RECORD
BTN0H000	PERSON SEARCH OPTIONS
BLG0J270	PMF AUTHORITY
BLG0Q270	PMF AUTHORITY
BLHLI122	PORTS
BLH0I222	PORTS QUERY
BLM8CU5I	PREFIX AND VALIDATION
BLGLRPQN	PRICE QUOTATION ENTRY
BLGLRPQ1	PRICE QUOTATION ENTRY
BLGLRPQ2	PRICE QUOTATION ENTRY
BLGLRPQ3	PRICE QUOTATION ENTRY
BLGLRPQ4	PRICE QUOTATION ENTRY
BLGLRPQ5	PRICE QUOTATION ENTRY
BLG0EN10	PRIMARY OPTIONS MENU
BLG0EN20	PRIMARY OPTIONS MENU
BLM0YE00	PRIMARY OPTIONS MENU
BLM0YE50	PRIMARY OPTIONS MENU
BTN0EN20	PRIMARY OPTIONS MENU
BLM8CU9R	PRINT
BLG0P420	PRINT DATA SET DESTINATION ENTRY
BLG0P430	PRINT DDNAME DESTINATION ENTRY
BLG0P410	PRINT SYSOUT DESTINATION ENTRY

|	**BLG0J400**	PRIVILEGE CLASS
|	**BLG0J401**	PRIVILEGE CLASS
|	**BLG0K400**	PRIVILEGE CLASS
|	**BLG0Q400**	PRIVILEGE CLASS
|	**BLGLSUPD**	PRIVILEGE CLASS
|	**BLGLSUPU**	PRIVILEGE CLASS
|	**BLGTROLE**	PRIVILEGE CLASS
|	**BLG1ROLE**	PRIVILEGE CLASS CONTROL
|	**BLG2J400**	PRIVILEGE CLASS HELP
|	**BLG2Q400**	PRIVILEGE CLASS HELP
|	**BLG2SUPP**	PRIVILEGE CLASS HELP
	BTN0EPPC	PRIVILEGE CLASS SELECTION
	BLG0J200	PROBLEM AUTHORITY
	BLG0Q200	PROBLEM AUTHORITY
	BTN0B101	PROBLEM CALLER INFORMATION
	BLG0E390	PROBLEM CLOSE DATA INQUIRY
	BTN0E390	PROBLEM CLOSE DATA INQUIRY
	BLG0L300	PROBLEM CLOSE DISPLAY
	BTN0L300	PROBLEM CLOSE DISPLAY
	BLG0B300	PROBLEM CLOSE ENTRY
	BTN0B300	PROBLEM CLOSE ENTRY
	BTN0ENPC	PROBLEM CONTROLLER PRIMARY OPTIONS
	BTN0B100	PROBLEM DATA
	BLG0BU10	PROBLEM DELETE VERIFICATION
	BLG0BU11	PROBLEM DELETE VERIFICATION
	BTN0BU10	PROBLEM DELETE VERIFICATION
	BTN0BU11	PROBLEM DELETE VERIFICATION
	BLG0E090	PROBLEM INQUIRY SUMMARY
	BTN0E090	PROBLEM INQUIRY SUMMARY
	BTN0W020	PROBLEM REPORT ENTRY
	BLG0B100	PROBLEM REPORTER ENTRY
	BLG0E190	PROBLEM REPORTER DATA INQUIRY
	BLG0L100	PROBLEM REPORTER DISPLAY
	BTN0E190	PROBLEM REPORTER DATA INQUIRY
	BTN0L100	PROBLEM REPORTER DISPLAY

BLG0B502	PROBLEM RESOLUTION DATA
BLG0E590	PROBLEM RESOLUTION DATA INQUIRY
BLG0E290	PROBLEM STATUS DATA INQUIRY
BLG0L200	PROBLEM STATUS DISPLAY
BTN0E290	PROBLEM STATUS DATA INQUIRY
BTN0L200	PROBLEM STATUS DISPLAY
BLG0B200	PROBLEM STATUS ENTRY
BTN0B200	PROBLEM STATUS ENTRY
BLG0BU00	PROBLEM SUMMARY
BLG0S010	PROBLEM SUMMARY DISPLAY
BTN0BU00	PROBLEM SUMMARY
BTN0S010	PROBLEM SUMMARY DISPLAY
BLG0B400	PROBLEM SUPPLEMENTAL ENTRY
BLG0E690	PROBLEM SUPPLEMENTAL DATA INQUIRY
BLG0L400	PROBLEM SUPPLEMENTAL DISPLAY
BLG0B402	PROBLEM SYMPTOM DATA
BLG0E490	PROBLEM SYMPTOM DATA INQUIRY
BLG0B500	PROBLEM SYNOPSIS ENTRY
BLG0L500	PROBLEM SYNOPSIS DISPLAY
BTN0B500	PROBLEM SYNOPSIS ENTRY
BTN0L500	PROBLEM SYNOPSIS DISPLAY
BLGLNPRO	PROCEDURE NAME ENTRY
BLGLNPR1	PROCEDURE NAME ENTRY
BLGLNPR2	PROCEDURE NAME ENTRY
BLGLNUPR	PROCEDURE NUMBER ENTRY
BLGLNUP1	PROCEDURE NUMBER ENTRY
BLGLNUP2	PROCEDURE NUMBER ENTRY
BLM8CU9H	PROCESS SPECIFICATION
BLG6EFND	PROFILE-ENTRY FIELD NUMBER DISPLAY AE
BLG0PU00	PROFILE SUMMARY
BLG6UTZN	PROFILE TIME ZONE AE
BLGLPIDF	PROGRAM IDENTITY ENTRY
BLGLPRID	PROGRAM IDENTITY ENTRY
BLGLPUBF	PUBLICATION IDENTITY ENTRY
BLGLPUBS	PUBLICATION IDENTITY ENTRY

BTN0F000	QUICK CHANGE SEARCH PANEL
BTN0G000	QUICK CONFIG SEARCH PANEL
BTN0E000	QUICK PROBLEM SEARCH PANEL
BLG0K293	RECORD AUTHORITY
BLM0Y261	RECORD CONTROL DATA INQUIRY
BLM0Y761	RECORD CONTROL DATA INQUIRY
BLM0YU10	RECORD DELETE VERIFICATION
BLM0YU11	RECORD DELETE VERIFICATION
BLM0YU60	RECORD DELETE VERIFICATION
BLM0YU61	RECORD DELETE VERIFICATION
BLM0Y100	RECORD ENTRY
BLM0Y203	RECORD INQUIRY
BLM0YU00	RECORD SUMMARY
BLM0Y300	RECORD SUMMARY DISPLAY
BLM0Y800	RECORD SUMMARY DISPLAY
BLG0QUP0	RECORD UTILITIES
BLG0R020	RECORD UTILITIES
BLG0S011	RECORD UTILITIES
BLG0S021	RECORD UTILITIES
BLG0S061	RECORD UTILITIES
BLG0T101	RECORD UTILITIES
BLM0Y330	RECORD UTILITIES
BLM0Y830	RECORD UTILITIES
BLM0Y880	RECORD UTILITIES
BTN0S021	RECORD UTILITIES
BLM0Y793	RECORD+ INQUIRY
BLGLSQMP	RELATIONAL DATA MAPPING TABLE ENTRY
BLGSSQMP	RELATIONAL DATA MAPPING TABLE DISPLAY
BLG1UT02	RENAME COMPONENT DIALOG
BLG0P300	REPORT CONTROL DEFAULTS
BLG0W500	REPORT ENTRY
BLG1C015	RESET CHANGE APPROVALS CONTROL
BLG0VE72	RESPONSE PROCESSING INQUIRY
BLG0VQ72	RESPONSE PROCESSING DISPLAY
BLG0V710	RESPONSE PROCESSING

BLM8CU5D	RESPONSE PROCESSING
BLG6REVD	REVIEWER
BLG0P200	SCREEN CONTROL DEFAULTS
BLM8CU13	SELECTION CONTROL SUMMARY
BLM8CU11	SELECTION PANEL UPDATE
BLGLDMSN	SERIAL NUMBER ENTRY
BLGLDMS1	SERIAL NUMBER ENTRY
BLGLDMS2	SERIAL NUMBER ENTRY
BLGLDMS3	SERIAL NUMBER ENTRY
BLGLDMS4	SERIAL NUMBER ENTRY
BLGLDMS5	SERIAL NUMBER ENTRY
BLGLDVSR	SERIAL NUMBER ENTRY
BLG0DU17	SERVICE DELETE VERIFICATION
BLG0DU27	SERVICE DELETE VERIFICATION
BLG0D400	SERVICE ENTRY
BLG0G490	SERVICE INQUIRY
BLG0N410	SERVICE RECORD UTILITIES
BLG0DU73	SERVICE RENAME VERIFICATION
BLG0DU07	SERVICE SUMMARY
BLG0G493	SERVICE SUMMARY
BLG0N400	SERVICE SUMMARY DISPLAY
BLG0P100	SESSION DEFAULTS
BLG0P101	SESSION DEFAULTS
BLG1ATSA	SET INDEX FOR TEXT SEARCH
BLM8CU9S	SETFIELD SPECIFICATION
BLGLJHPL	SHARING LOGICAL PARTITION IDENTIFIER ENTRY
BLGLQHPL	SHARING LOGICAL PARTITION IDENTIFIER DISPLAY
BLG0C051	SOFTWARE ADD
BLG0C151	SOFTWARE ADD INQUIRY
BLG0C052	SOFTWARE ALTER
BLG0C053	SOFTWARE ALTER BY INSTALLING
BLG0C054	SOFTWARE ALTER BY MODIFYING
BLG0C055	SOFTWARE ALTER BY REMOVING
BLG0C153	SOFTWARE ALTER BY INSTALLING INQUIRY
BLG0C154	SOFTWARE ALTER BY MODIFYING INQUIRY

BLG0C155	SOFTWARE ALTER BY REMOVING INQUIRY
BLG0C152	SOFTWARE ALTER INQUIRY
BLGLPZFF	SOFTWARE APAR ENTRY
BLGLSWCC	SOFTWARE COMPLETION CODE ENTRY
BLGLSPID	SOFTWARE COMPONENT ENTRY
BLGLSPI1	SOFTWARE COMPONENT ENTRY
BLGLSPI2	SOFTWARE COMPONENT ENTRY
BLGLSPI3	SOFTWARE COMPONENT ENTRY
BLGLSPI4	SOFTWARE COMPONENT ENTRY
BLGLSPI5	SOFTWARE COMPONENT ENTRY
BLG0DU03	SOFTWARE COMPONENT SUMMARY
BLG0DU13	SOFTWARE COMPONENT DELETE VERIFICATION
BLG0DU23	SOFTWARE COMPONENT DELETE VERIFICATION
BLG0DU31	SOFTWARE COMPONENT SUMMARY
BLG0DU62	SOFTWARE COMPONENT RENAME VERIFICATION
BLG0DU76	SOFTWARE COMPONENT SWAP VERIFICATION
BLG0D200	SOFTWARE COMPONENT ENTRY
BLG0D205	SOFTWARE COMPONENT SOURCE DEFINITION ENTRY
BLG0D230	SOFTWARE COMPONENT CONNECTION ENTRY
BLG0D250	SOFTWARE COMPONENT SUPPORT ENTRY
BLG0D251	SOFTWARE COMPONENT SECONDARY SUPPORT ENTRY
BLG0D253	SOFTWARE COMPONENT FINANCIAL ENTRY
BLG0G1A1	SOFTWARE COMPONENT DESCRIPTION/CONTROL DATA INQUIRY
BLG0G1A3	SOFTWARE COMPONENT SUMMARY
BLG0G1B4	SOFTWARE COMPONENT CONNECTION INQUIRY
BLG0G1B5	SOFTWARE COMPONENT CONNECTION SUMMARY
BLG0N200	SOFTWARE COMPONENT SUMMARY DISPLAY
BLG0N201	SOFTWARE COMPONENT DISPLAY
BLG0N205	SOFTWARE COMPONENT SOURCE DEFINITION DISPLAY
BLG0N210	SOFTWARE COMPONENT RECORD UTILITIES
BLG0N250	SOFTWARE COMPONENT SUPPORT DISPLAY
BLG0N251	SOFTWARE COMPONENT SECONDARY SUPPORT DISPLAY
BLG0N253	SOFTWARE COMPONENT FINANCIAL DISPLAY
BTN0DU03	SOFTWARE COMPONENT SUMMARY
BTN0DU31	SOFTWARE COMPONENT SUMMARY

BTN0D200	SOFTWARE COMPONENT ENTRY
BTN0G1A1	SOFTWARE COMPONENT INQUIRY
BTN0G1A3	SOFTWARE COMPONENT INQUIRY SUMMARY
BTN0N200	SOFTWARE COMPONENT SUMMARY DISPLAY
BTN0N201	SOFTWARE COMPONENT DISPLAY
BLG0DU10	SOFTWARE CONNECTION DELETE VERIFICATION
BLG0DU20	SOFTWARE CONNECTION DELETE VERIFICATION
BLG0DU40	SOFTWARE CONNECTION SUMMARY
BLG0DU63	SOFTWARE CONNECTION RENAME VERIFICATION
BLG0N230	SOFTWARE CONNECTION SUMMARY DISPLAY
BLG0N231	SOFTWARE CONNECTION RECORD UTILITIES
BLG0N232	SOFTWARE CONNECTION DISPLAY
BLG0C800	SOFTWARE DISTRIBUTION ENTRY
BLG0M800	SOFTWARE DISTRIBUTION DISPLAY
BLG0DU04	SOFTWARE FEATURE SUMMARY
BLG0DU14	SOFTWARE FEATURE DELETE VERIFICATION
BLG0DU24	SOFTWARE FEATURE DELETE VERIFICATION
BLG0DU64	SOFTWARE FEATURE RENAME VERIFICATION
BLG0D220	SOFTWARE FEATURE ENTRY
BLG0D254	SOFTWARE FEATURE FINANCIAL ENTRY
BLG0D257	SOFTWARE FEATURE SUPPORT ENTRY
BLG0G1B1	SOFTWARE FEATURE DESCRIPTION/CONTROL DATA INQUIRY
BLG0G1B2	SOFTWARE FEATURE FMID/SUPPORT/FINANCIAL INQUIRY
BLG0G1B3	SOFTWARE FEATURE SUMMARY
BLG0N220	SOFTWARE FEATURE SUMMARY DISPLAY
BLG0N221	SOFTWARE FEATURE RECORD UTILITIES
BLG0N222	SOFTWARE FEATURE DISPLAY
BLG0N254	SOFTWARE FEATURE FINANCIAL DISPLAY
BLG0N257	SOFTWARE FEATURE SUPPORT DISPLAY
BLGLFLDF	SOFTWARE FIELD NAME ENTRY
BLG0DU06	SOFTWARE FINANCIAL SUMMARY
BLG0DU16	SOFTWARE FINANCIAL DELETE VERIFICATION
BLG0DU26	SOFTWARE FINANCIAL DELETE VERIFICATION
BLG0DU65	SOFTWARE FINANCIAL RENAME VERIFICATION
BLG0D350	SOFTWARE FINANCIAL ENTRY

BLG0D353	SOFTWARE FINANCIAL - BASIC LICENSE ENTRY
BLG0D355	SOFTWARE FINANCIAL - DSLO LICENSE ENTRY
BLG0D356	SOFTWARE FINANCIAL ENTRY
BLG0D357	SOFTWARE FINANCIAL - VLA ENTRY
BLG0G357	SOFTWARE FINANCIAL DESCRIPTION DATA INQUIRY
BLG0G358	SOFTWARE FINANCIAL SUMMARY
BLG0G359	SOFTWARE FINANCIAL VLA/CONTROL DATA INQUIRY
BLG0N350	SOFTWARE FINANCIAL SUMMARY DISPLAY
BLG0N352	SOFTWARE FINANCIAL DISPLAY
BLG0N353	SOFTWARE FINANCIAL - BASIC LICENSE DISPLAY
BLG0N355	SOFTWARE FINANCIAL - DSLO LICENSE DISPLAY
BLG0N357	SOFTWARE FINANCIAL - VLA DISPLAY
BLG0N360	SOFTWARE FINANCIAL RECORD UTILITIES
BLGLLVLF	SOFTWARE LEVEL ENTRY
BLGLLVLP	SOFTWARE LEVEL ENTRY
BLGLSLEV	SOFTWARE LEVEL ENTRY
BLGLSLE1	SOFTWARE LEVEL ENTRY
BLGLSLE2	SOFTWARE LEVEL ENTRY
BLGLSLE3	SOFTWARE LEVEL ENTRY
BLGLSLE4	SOFTWARE LEVEL ENTRY
BLGLSLE5	SOFTWARE LEVEL ENTRY
BLGLPTFF	SOFTWARE PTF ENTRY
BLGLPTNU	SOFTWARE PTF ENTRY
BLGLPTN1	SOFTWARE PTF ENTRY
BLGLPTN2	SOFTWARE PTF ENTRY
BLGLPTN3	SOFTWARE PTF ENTRY
BLGLPTN4	SOFTWARE PTF ENTRY
BLGLPTN5	SOFTWARE PTF ENTRY
BLH0I023	SOFTWARE QUERY MENU
BLG0G167	SOFTWARE RECORDS+ INQUIRY
BLG0G168	SOFTWARE RECORDS+ SUMMARY
BLGLREGF	SOFTWARE REGISTER ENTRY
BLG0C056	SOFTWARE REMOVE
BLG0C156	SOFTWARE REMOVE INQUIRY
BLGLSWRC	SOFTWARE RETURN CODE ENTRY

	BLGLRMID	SOFTWARE ROUTINE ENTRY
	BLGLRMI1	SOFTWARE ROUTINE ENTRY
	BLGLRMI2	SOFTWARE ROUTINE ENTRY
	BLGLRMI3	SOFTWARE ROUTINE ENTRY
	BLGLRMI4	SOFTWARE ROUTINE ENTRY
	BLGLRMI5	SOFTWARE ROUTINE ENTRY
	BLGLROUF	SOFTWARE ROUTINE NAME ENTRY
	BLGLROUS	SOFTWARE ROUTINE NAME ENTRY
	BLGLPCSF	SOFTWARE STATEMENT ENTRY
	BLGLSTMT	SOFTWARE STATEMENT ENTRY
	BLG0G1A2	SOFTWARE SUPPORT/FMID/FINANCIAL INQUIRY
|	**BLH0S201**	SOLUTION CONTROL DATA INQUIRY
|	**BLH1S004**	SOLUTION DELETE OWNERSHIP TEST CONTROL
|	**BLH0S000**	SOLUTION DIALOG ENTRY
|	**BLH0S110**	SOLUTION DISPLAY SUMMARY
|	**BLH0S010**	SOLUTION ENTRY SUMMARY
|	**BLH2S110**	SOLUTION ENTRY SUMMARY HELP
|	**BLG0J2C0**	SOLUTION RECORD
|	**BLG0Q2C0**	SOLUTION RECORD
|	**BLH1S000**	SOLUTION RECORD ACCESS CONTROL
|	**BLH1S006**	SOLUTION RECORD COPY CONTROL
|	**BLH1S009**	SOLUTION RECORD DELETE CONTROL
|	**BLH0S100**	SOLUTION RECORD DISPLAY
|	**BLH0S001**	SOLUTION RECORD ENTRY
|	**BLH1S040**	SOLUTION RECORD FILE CONTROL
|	**BLH0S200**	SOLUTION RECORD INQUIRY
|	**BLH1S012**	SOLUTION RECORD MULTIPLE DELETE CONTROL
|	**BLH1S003**	SOLUTION RECORD OWNERSHIP TEST CONTROL
|	**BLH1S008**	SOLUTION RECORD PRINT CONTROL
|	**BLH0S011**	SOLUTION RECORD UTILITIES
|	**BLH0S111**	SOLUTION RECORD UTILITIES
|	**BLH0S090**	SOLUTION RECORD VERIFY DELETE
|	**BLH0S091**	SOLUTION VERIFY MULTIPLE DELETE
	BLG0D107	SPECIFICATION FOR HARDWARE MODEL COPY
	BLG0D204	SPECIFICATION FOR SOFTWARE MODEL COPY

BLG0S530	SQL EXTRACT INFORMATION DISPLAY
BLG0J290	SRC AUTHORITY
BLG0Q290	SRC AUTHORITY
BLG0TU01	SRC DELETE VERIFICATION
BLG0TU02	SRC DELETE VERIFICATION
BLG0T110	SRC DESCRIPTION DISPLAY
BLG0U100	SRC DESCRIPTION ENTRY
BLG0V190	SRC DESCRIPTION INQUIRY
BLG0T200	SRC EXECUTION CLASSES
BLG0U200	SRC EXECUTION CLASSES
BLG0U000	SRC SUMMARY
BLG0U001	SRC SUMMARY
BLG0V090	SRC SUMMARY
BLG0T100	SRC SUMMARY DISPLAY
BLG0P510	STANDARD REPORT SYSOUT DESTINATION ENTRY
BLG0P511	STANDARD REPORT SYSOUT DESTINATION ENTRY
BLG0P520	STANDARD REPORT DATA SET DESTINATION ENTRY
BLG0P521	STANDARD REPORT DATA SET DESTINATION ENTRY
BLG0P530	STANDARD REPORT DDNAME DESTINATION ENTRY
BLG0P531	STANDARD REPORT DDNAME DESTINATION ENTRY
BLH1S011	START SOLUTION DIALOG CONTROL
BLM8CL20	STRING FIND
BLG0VE73	STRUCTURED WORD AND VALIDATION INQUIRY
BLG0VQ73	STRUCTURED WORD AND VALIDATION DISPLAY
BLG0V720	STRUCTURED WORD AND VALIDATION COLLECTION
BLM8CU5B	STRUCTURED WORD AND VALIDATION COLLECTION
BLGLSUID	SUBSET NUMBERS ENTRY
BLGLSUI1	SUBSET NUMBERS ENTRY
BLGLSUI2	SUBSET NUMBERS ENTRY
BLGLSUI3	SUBSET NUMBERS ENTRY
BLGLSUI4	SUBSET NUMBERS ENTRY
BLGLSUI5	SUBSET NUMBERS ENTRY
BTN0ENSU	SUPPORT PRIMARY OPTIONS
BLG1UT20	SWAP COMPONENTS
BTN0ENSY	SYSTEM ADMINISTRATOR PRIMARY OPTIONS

BLG0K292	SYSTEM CLASS AUTHORITY
BLG0DU19	SYSTEM DELETE VERIFICATION
BLG0DU29	SYSTEM DELETE VERIFICATION
BLG0D600	SYSTEM ENTRY
BLG0G290	SYSTEM INQUIRY
BLG0N610	SYSTEM RECORD UTILITIES
BLG0DU72	SYSTEM RENAME VERIFICATION
BLG0DU09	SYSTEM SUMMARY
BLG0G293	SYSTEM SUMMARY
BLG0N600	SYSTEM SUMMARY DISPLAY
BLG0TSFL	Task Field List Selection
BLM8CU91	TERMINAL SIMULATOR PANEL UPDATE
BLM8CU64	TEST DATA PROCESSING UPDATE
BLM8CU9J	TESTFIELD SPECIFICATION
BLM8CU9K	TESTFLOW SPECIFICATION
BLH6INDX	TEXT INDEX NAME
BLG1TTSA	TEXT SEARCH ARGUMENT INPUT
BLG2TTSA	TEXT SEARCH ARGUMENT INPUT HELP
BLHLNTXT	TEXT SEARCH INPUT LIST
BLH1TTXT	TEXT SEARCH RESULTS LIST
BLG1ASTS	TIME ZONE CONTROL
BLG1AZDF	TIME ZONE CONTROL
BLGLDSSD	TIME ZONE DISPLAY
BLGLTZDD	TIME ZONE DISPLAY
BLG0Z250	TIME ZONE DISPLAY
BLG0Z200	TIME ZONE RECORD SUMMARY
BLH0I000	TIVOLI INVENTORY QUERY MENU
BLH0I004	TIVOLI INVENTORY REPORT MENU
BLM8CU71	TOP TITLE
BLM8CU76	TOP TITLE
BLM8CU78	TOP TITLE
BLM8CU79	TOP TITLE
BLM8CU9N	TRACE SPECIFICATION
BLG0Y300	TSD BRIDGE CLEANUP
BLG0E790	TSD BRIDGE DATA INQUIRY

	BLG0L700	TSD BRIDGE DISPLAY
|	**BLHTSNDE**	TSP FOR INDEX DELETE PROCESSING
	BLM8CU9X	UNFLATTEN
|	**BLGLDSTS**	UNIVERSAL TIME SUPPORT
|	**BLGLDTMD**	UNIVERSAL TIME SUPPORT
|	**BLGLDTTM**	UNIVERSAL TIME SUPPORT
|	**BLGLTZDF**	UNIVERSAL TIME SUPPORT
|	**BLGTFDAT**	UNIVERSAL TIME SUPPORT
|	**BLGTFZON**	UNIVERSAL TIME SUPPORT
|	**BLG6DTRC**	UNIVERSAL TIME SUPPORT
|	**BLG1VQ50**	UNIVERSAL TIME SUPPORT CONTROL
|	**BLG1VQ70**	UNIVERSAL TIME SUPPORT CONTROL
	BLH0I012	UNIX ASSOCIATED DATA
|	**BLH0X011**	UPDATE ADDRESSES SELECTION
|	**BLHLX0BC**	UPDATE BCC ADDRESS
|	**BLHLX0CC**	UPDATE CC ADDRESS
|	**BLG1AAPU**	UPDATE CHANGE APPROVER CONTROL
|	**BLG1AREU**	UPDATE CHANGE REVIEWER CONTROL
|	**BLH1N003**	UPDATE INDEX RECORD OWNERSHIP CONTROL
|	**BLHLX0TO**	UPDATE TO ADDRESS
	BLGLUTID	UPGRADE/TAPE ENTRY
	BLGLUTI1	UPGRADE/TAPE ENTRY
	BLGLUTI2	UPGRADE/TAPE ENTRY
	BLGLUTI3	UPGRADE/TAPE ENTRY
	BLGLUTI4	UPGRADE/TAPE ENTRY
	BLGLUTI5	UPGRADE/TAPE ENTRY
	BLGLMODF	USER MODIFICATION ENTRY
	BLGLUMID	USER MODIFICATION ENTRY
	BLGLUMI1	USER MODIFICATION ENTRY
	BLGLUMI2	USER MODIFICATION ENTRY
	BLGLUMI3	USER MODIFICATION ENTRY
	BLGLUMI4	USER MODIFICATION ENTRY
	BLGLUMI5	USER MODIFICATION ENTRY
	BLG0B901	USERS TO NOTIFY
	BLG0B902	USERS TO NOTIFY

	BLG0B903	USERS TO NOTIFY
	BLG0L901	USERS TO NOTIFY
	BLG0L902	USERS TO NOTIFY
	BLG0L903	USERS TO NOTIFY
	BLG1UT01	UTILITY ENTRY DIALOG
|	**BLH1X011**	VALIDATE MODEL RECORD ENTRY CONTROL
|	**BLH1S010**	VALIDATE PROBLEM RNID CONTROL
|	**BLH1X010**	VALIDATE RECORD ENTRY CONTROL
|	**BLH1S014**	VALIDATE SOLUTION FF TEXT DATA CONTROL
	BLG0VQ52	VALIDATION CONTROL DATA DISPLAY
	BLGLVLST	VALIDATION DATA ENTRY
	BLGLVSEL	VALIDATION DATA
	BLG0VLST	VALIDATION DATA DISPLAY
	BLG0V502	VALIDATION DISPLAY UPDATE
	BLG0VD50	VALIDATION RECORD DELETE VERIFICATION
	BLG0VD51	VALIDATION RECORD DELETE VERIFICATION
	BLG0V500	VALIDATION RECORD ENTRY
	BLG0V510	VALIDATION RECORD INQUIRY
	BLG0VQ50	VALIDATION SUMMARY DISPLAY
	BLG0VU50	VALIDATION SUMMARY
	BLG0BU90	VERIFICATION OF USERS RECORD DELETE
	BLG0ZD50	VERIFICATION OF LOGSAVE RECORD DELETE
	BLM8CU9U	WORDFIX ADD SPECIFICATION
	BLM8CU9M	WORDFIX CONTROL LINE SUMMARY
	BLM8CU9V	WORDFIX DELETE SPECIFICATION
	BLM8CU9Y	WORDFIX P-WORD SPECIFICATION
	BLM8CU9W	WORDFIX S-WORD SPECIFICATION

B

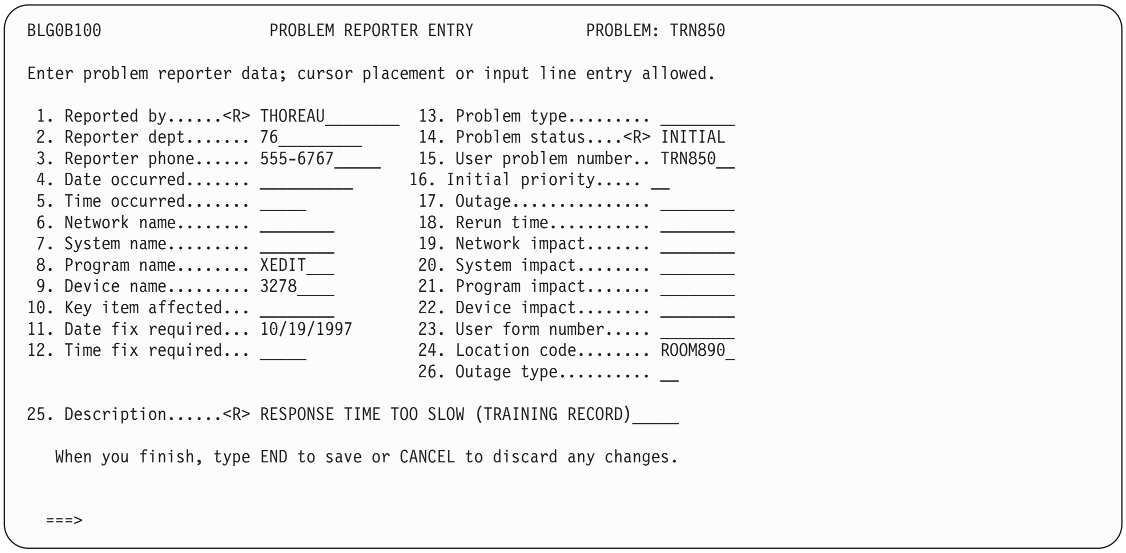
Training Records

Use the panels that follow as a guide when creating training records. The panels show data to enter in each record. To access these panels, enter the immediate response chain (IRC) 5,1 on the command line of Primary Options Menu panel for the Management application. Once you create the first record, you can copy and modify it to create the rest.

Create training records so that users can follow the training exercises in the *Tivoli Information Management for z/OS User's Guide*. To create training records, you must be authorized to create problem records.

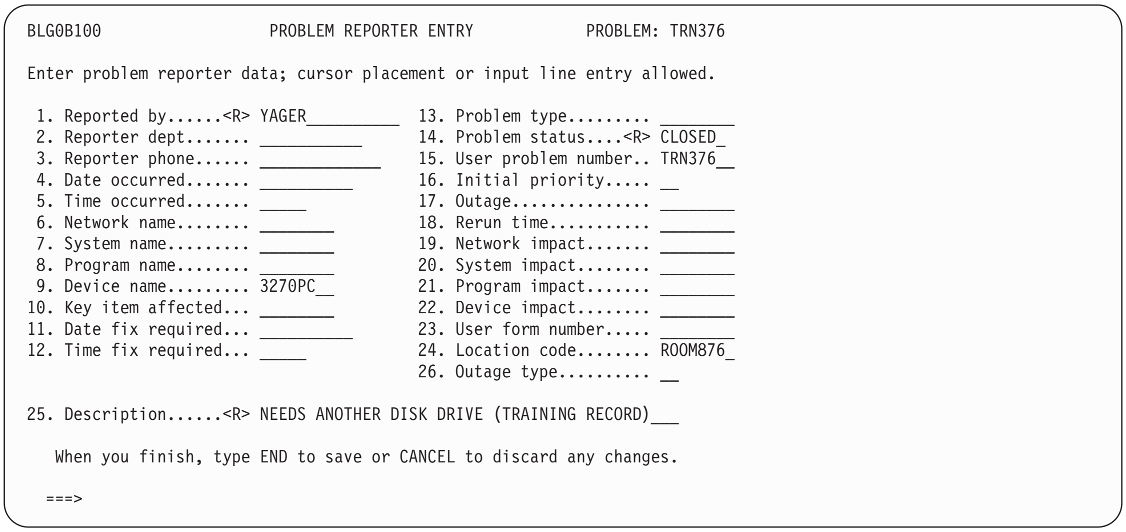
Creating Training Records

TRN850



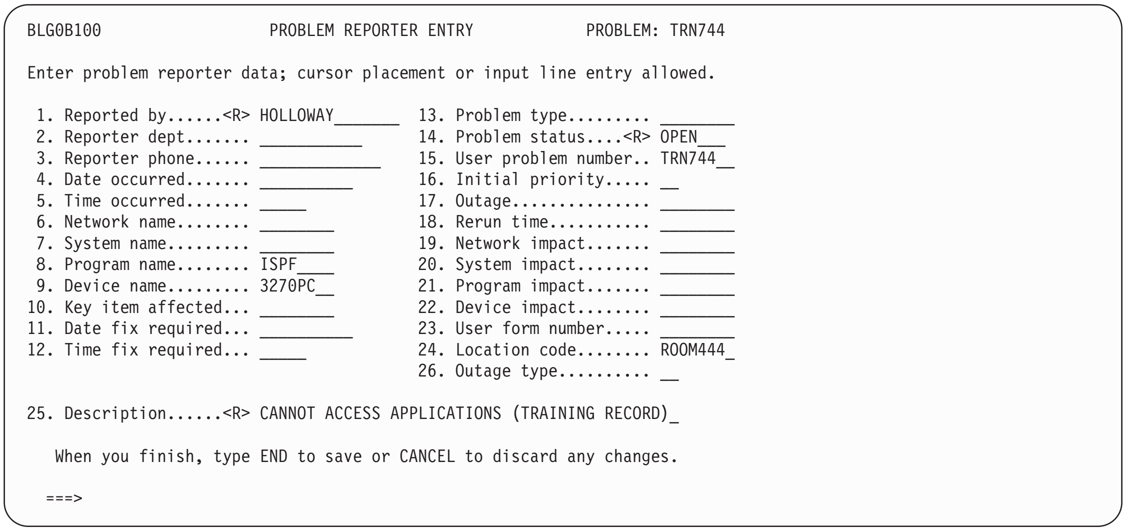
```
BLG0B100                  PROBLEM REPORTER ENTRY            PROBLEM: TRN850

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> THOREAU________  13. Problem type......... ________
 2. Reporter dept....... 76_________      14. Problem status....<R> INITIAL
 3. Reporter phone...... 555-6767_____    15. User problem number.. TRN850__
 4. Date occurred....... __________      16. Initial priority..... __
 5. Time occurred....... _____            17. Outage............... ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ XEDIT___         20. System impact........ ________
 9. Device name......... 3278____         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... 10/19/1997       23. User form number..... ________
12. Time fix required... _____            24. Location code........ ROOM890_
                                          26. Outage type.......... __

25. Description......<R> RESPONSE TIME TOO SLOW (TRAINING RECORD)_____

   When you finish, type END to save or CANCEL to discard any changes.


  ===>
```

TRN376



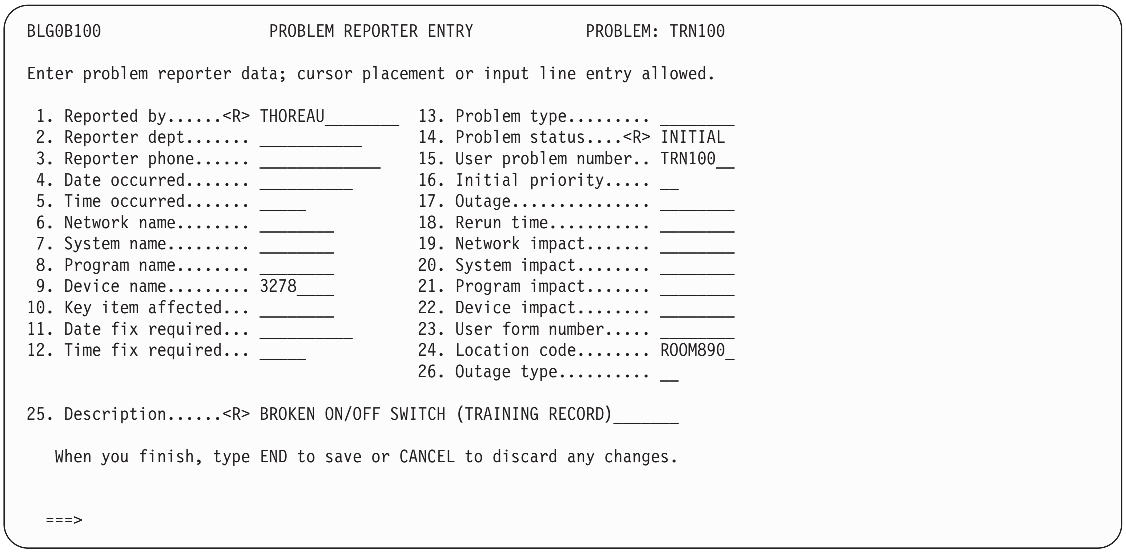
```
BLG0B100                  PROBLEM REPORTER ENTRY           PROBLEM: TRN376

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> YAGER__________  13. Problem type......... ________
 2. Reporter dept....... ___________      14. Problem status....<R> CLOSED_
 3. Reporter phone...... _____________    15. User problem number.. TRN376__
 4. Date occurred....... __________       16. Initial priority..... __
 5. Time occurred....... _____            17. Outage............... ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ ________         20. System impact........ ________
 9. Device name......... 3270PC__         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... __________       23. User form number..... ________
12. Time fix required... _____            24. Location code........ ROOM876_
                                          26. Outage type.......... __

25. Description......<R> NEEDS ANOTHER DISK DRIVE (TRAINING RECORD)___

   When you finish, type END to save or CANCEL to discard any changes.

  ===>
```

TRN744



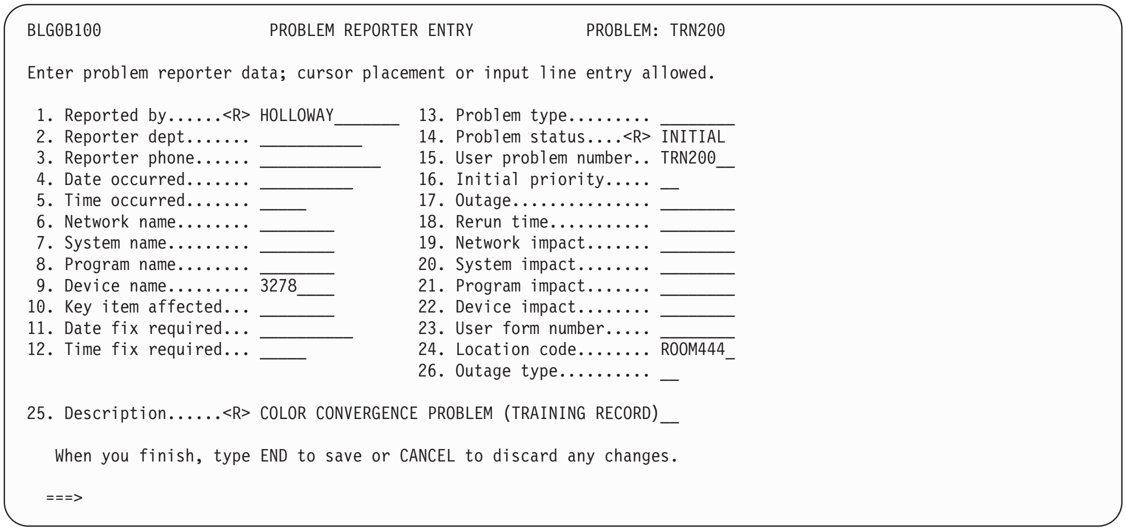
```
BLG0B100                  PROBLEM REPORTER ENTRY           PROBLEM: TRN744

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> HOLLOWAY_______  13. Problem type......... ________
 2. Reporter dept....... ___________      14. Problem status....<R> OPEN___
 3. Reporter phone...... _____________    15. User problem number.. TRN744__
 4. Date occurred....... __________       16. Initial priority..... __
 5. Time occurred....... _____            17. Outage............... ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ ISPF____         20. System impact........ ________
 9. Device name......... 3270PC__         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... __________       23. User form number..... ________
12. Time fix required... _____            24. Location code........ ROOM444_
                                          26. Outage type.......... __

25. Description......<R> CANNOT ACCESS APPLICATIONS (TRAINING RECORD)_

   When you finish, type END to save or CANCEL to discard any changes.

  ===>
```

TRN100



```
BLG0B100                  PROBLEM REPORTER ENTRY            PROBLEM: TRN100

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> THOREAU________  13. Problem type......... ________
 2. Reporter dept....... ___________      14. Problem status....<R> INITIAL
 3. Reporter phone...... _____________    15. User problem number.. TRN100__
 4. Date occurred....... __________       16. Initial priority..... __
 5. Time occurred....... _____            17. Outage............... ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ ________         20. System impact........ ________
 9. Device name......... 3278____         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... __________       23. User form number..... ________
12. Time fix required... _____            24. Location code........ ROOM890_
                                          26. Outage type.......... __

25. Description......<R> BROKEN ON/OFF SWITCH (TRAINING RECORD)_______

   When you finish, type END to save or CANCEL to discard any changes.


  ===>
```

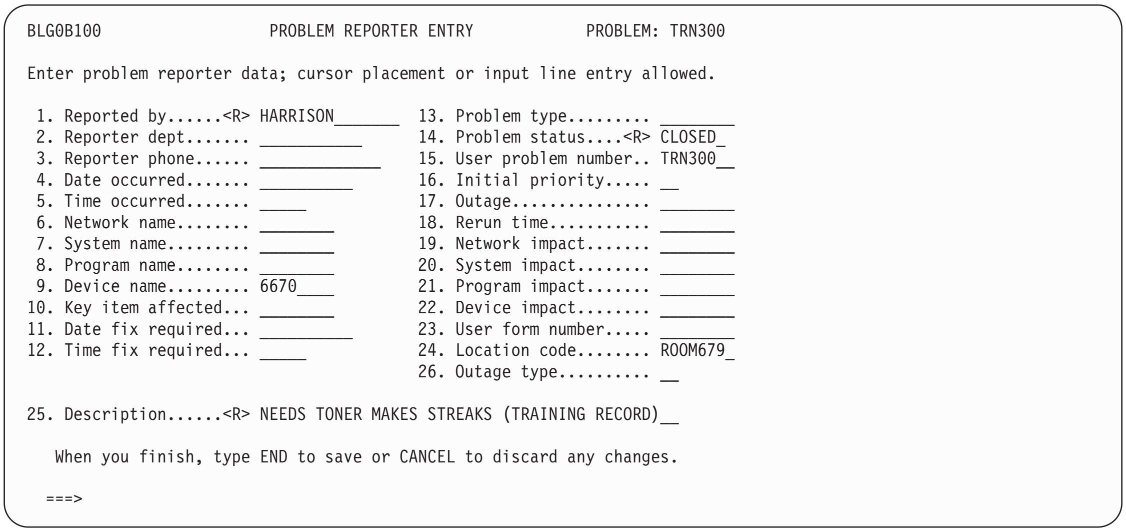
TRN200



```
BLG0B100                  PROBLEM REPORTER ENTRY            PROBLEM: TRN200

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> HOLLOWAY_______  13. Problem type......... ________
 2. Reporter dept....... ___________      14. Problem status....<R> INITIAL
 3. Reporter phone...... _____________    15. User problem number.. TRN200__
 4. Date occurred....... __________       16. Initial priority..... __
 5. Time occurred....... _____            17. Outage............... ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ ________         20. System impact........ ________
 9. Device name......... 3278____         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... __________       23. User form number..... ________
12. Time fix required... _____            24. Location code........ ROOM444_
                                          26. Outage type.......... __

25. Description......<R> COLOR CONVERGENCE PROBLEM (TRAINING RECORD)__

   When you finish, type END to save or CANCEL to discard any changes.

  ===>
```

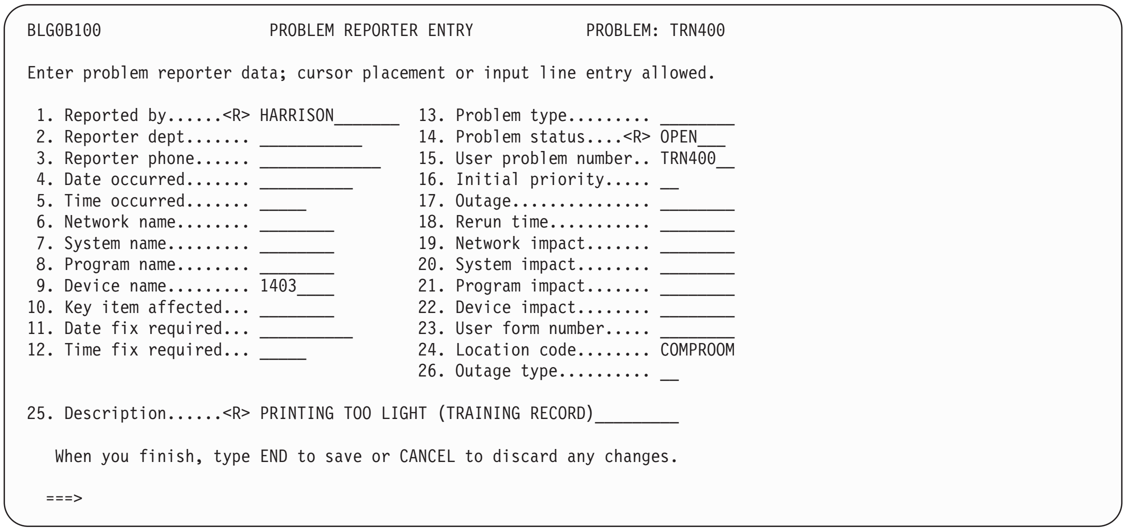
B. Training Records

TRN300



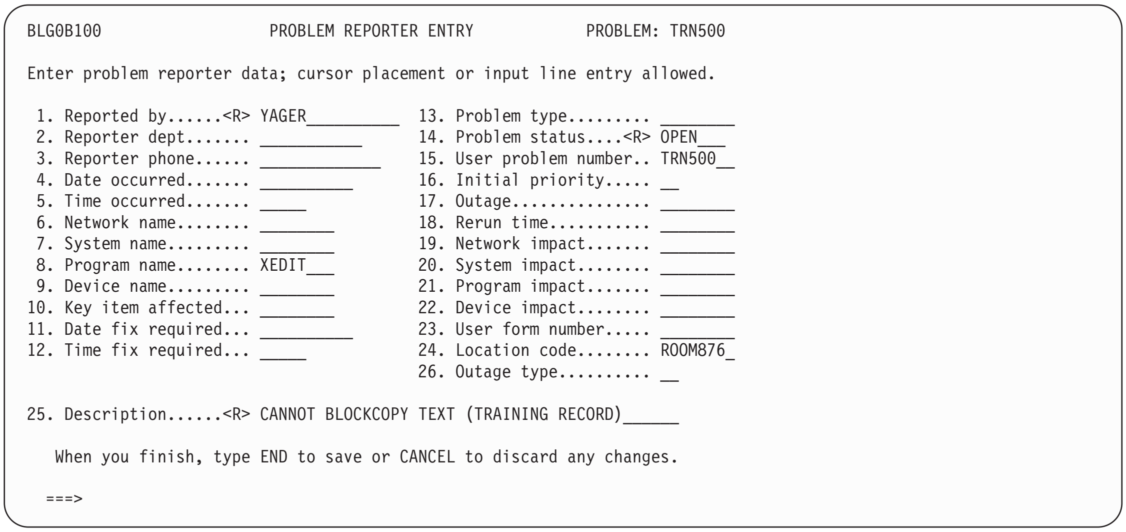
```
BLG0B100                  PROBLEM REPORTER ENTRY            PROBLEM: TRN300

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> HARRISON_______  13. Problem type......... ________
 2. Reporter dept....... ___________      14. Problem status....<R> CLOSED_
 3. Reporter phone...... _____________    15. User problem number.. TRN300__
 4. Date occurred....... __________       16. Initial priority..... __
 5. Time occurred....... _____            17. Outage.............. ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ ________         20. System impact........ ________
 9. Device name......... 6670____         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... __________       23. User form number..... ________
12. Time fix required... _____            24. Location code........ ROOM679_
                                          26. Outage type.......... __

25. Description......<R> NEEDS TONER MAKES STREAKS (TRAINING RECORD)__

   When you finish, type END to save or CANCEL to discard any changes.

  ===>
```

TRN400



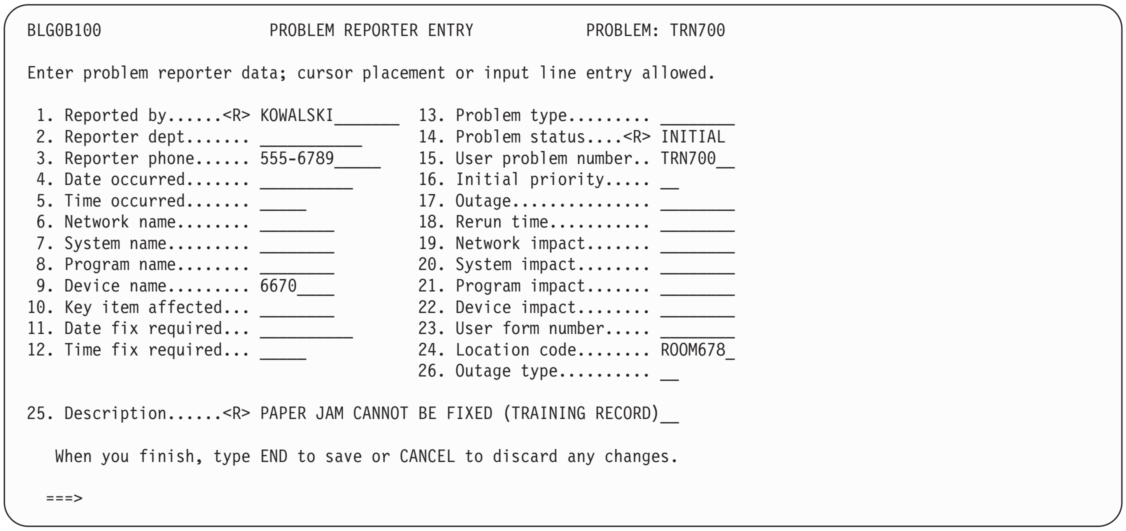
```
BLG0B100                  PROBLEM REPORTER ENTRY            PROBLEM: TRN400

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> HARRISON_______  13. Problem type......... ________
 2. Reporter dept....... ___________      14. Problem status....<R> OPEN___
 3. Reporter phone...... _____________    15. User problem number.. TRN400__
 4. Date occurred....... __________       16. Initial priority..... __
 5. Time occurred....... _____            17. Outage.............. ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ ________         20. System impact........ ________
 9. Device name......... 1403____         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... __________       23. User form number..... ________
12. Time fix required... _____            24. Location code........ COMPROOM
                                          26. Outage type.......... __

25. Description......<R> PRINTING TOO LIGHT (TRAINING RECORD)_________

   When you finish, type END to save or CANCEL to discard any changes.

  ===>
```

TRN500



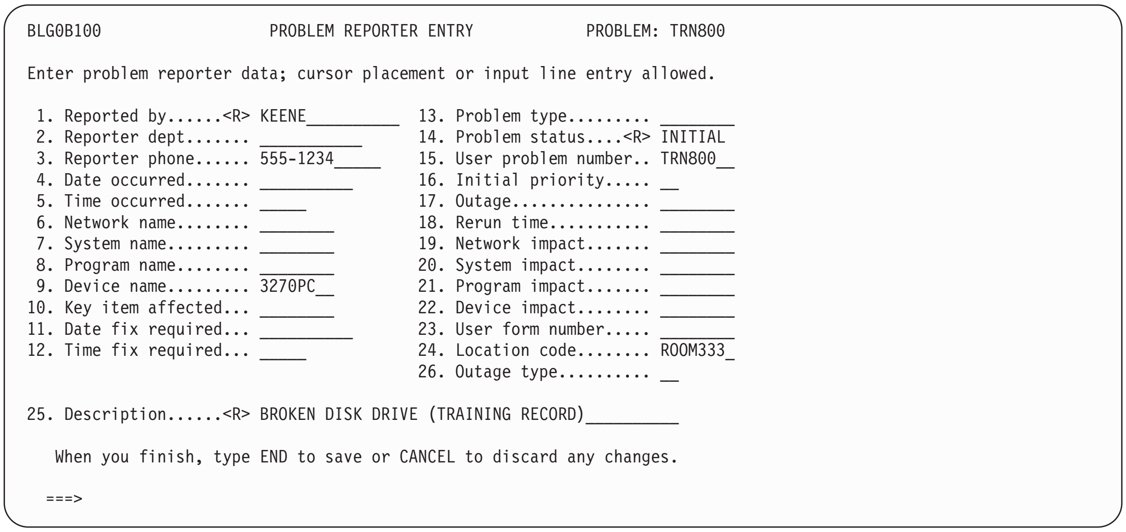
```
BLG0B100                  PROBLEM REPORTER ENTRY            PROBLEM: TRN500

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> YAGER__________  13. Problem type......... ________
 2. Reporter dept....... ___________      14. Problem status....<R> OPEN___
 3. Reporter phone...... _____________    15. User problem number.. TRN500__
 4. Date occurred....... __________       16. Initial priority..... __
 5. Time occurred....... _____            17. Outage............... ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ XEDIT___         20. System impact........ ________
 9. Device name......... ________         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... __________       23. User form number..... ________
12. Time fix required... _____            24. Location code........ ROOM876_
                                          26. Outage type.......... __

25. Description......<R> CANNOT BLOCKCOPY TEXT (TRAINING RECORD)______

   When you finish, type END to save or CANCEL to discard any changes.

  ===>
```

TRN700



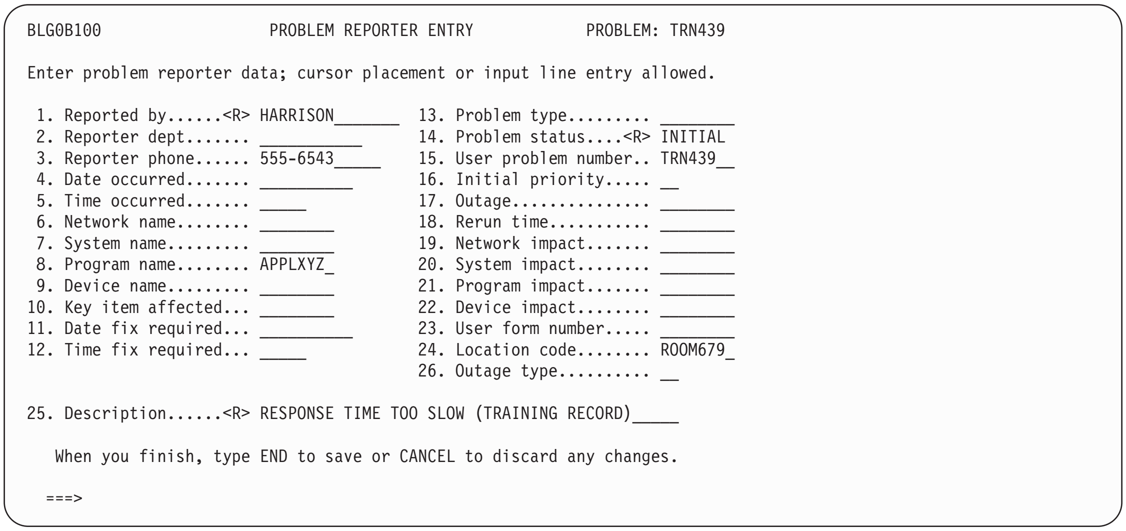
```
BLG0B100                  PROBLEM REPORTER ENTRY            PROBLEM: TRN700

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> KOWALSKI_______  13. Problem type......... ________
 2. Reporter dept....... ___________      14. Problem status....<R> INITIAL
 3. Reporter phone...... 555-6789_____    15. User problem number.. TRN700__
 4. Date occurred....... __________       16. Initial priority..... __
 5. Time occurred....... _____            17. Outage............... ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ ________         20. System impact........ ________
 9. Device name......... 6670____         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... __________       23. User form number..... ________
12. Time fix required... _____            24. Location code........ ROOM678_
                                          26. Outage type.......... __

25. Description......<R> PAPER JAM CANNOT BE FIXED (TRAINING RECORD)__

   When you finish, type END to save or CANCEL to discard any changes.

  ===>
```

TRN800



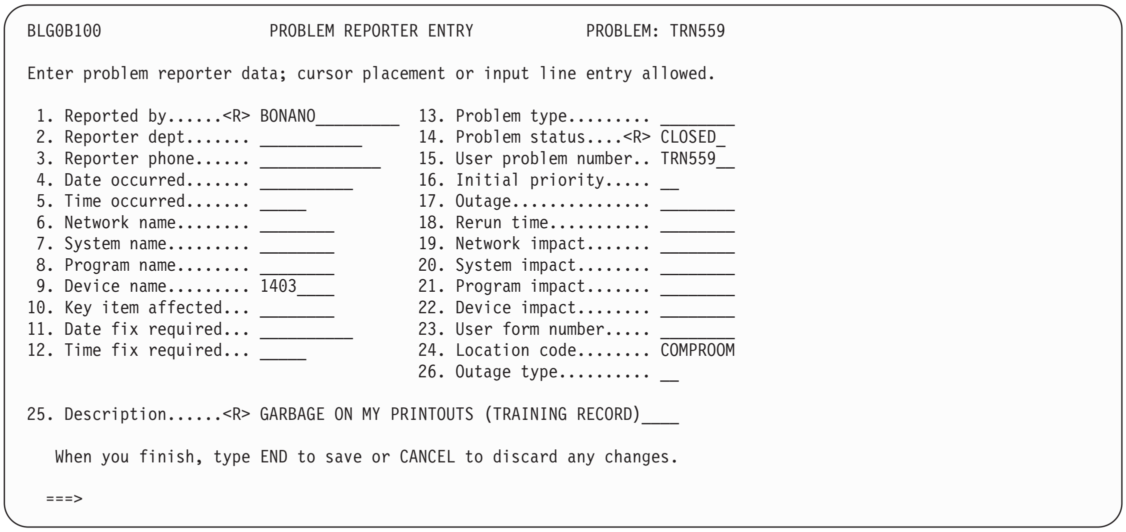
```
BLG0B100                  PROBLEM REPORTER ENTRY           PROBLEM: TRN800

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> KEENE__________  13. Problem type......... ________
 2. Reporter dept....... ___________      14. Problem status....<R> INITIAL
 3. Reporter phone...... 555-1234_____    15. User problem number.. TRN800__
 4. Date occurred....... __________       16. Initial priority..... __
 5. Time occurred....... _____            17. Outage............... ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ ________         20. System impact........ ________
 9. Device name......... 3270PC__         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... __________       23. User form number..... ________
12. Time fix required... _____            24. Location code........ ROOM333_
                                          26. Outage type.......... __

25. Description......<R> BROKEN DISK DRIVE (TRAINING RECORD)__________

   When you finish, type END to save or CANCEL to discard any changes.

  ===>
```

TRN439



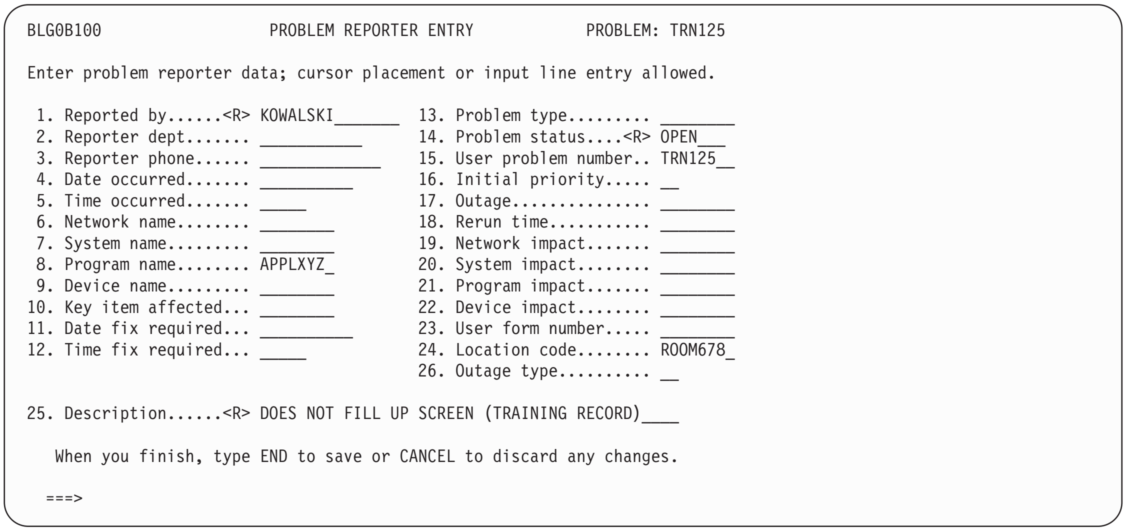
```
BLG0B100                  PROBLEM REPORTER ENTRY           PROBLEM: TRN439

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> HARRISON_______  13. Problem type......... ________
 2. Reporter dept....... ___________      14. Problem status....<R> INITIAL
 3. Reporter phone...... 555-6543_____    15. User problem number.. TRN439__
 4. Date occurred....... __________       16. Initial priority..... __
 5. Time occurred....... _____            17. Outage............... ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ APPLXYZ_         20. System impact........ ________
 9. Device name......... ________         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... __________       23. User form number..... ________
12. Time fix required... _____            24. Location code........ ROOM679_
                                          26. Outage type.......... __

25. Description......<R> RESPONSE TIME TOO SLOW (TRAINING RECORD)_____

   When you finish, type END to save or CANCEL to discard any changes.

  ===>
```

TRN559



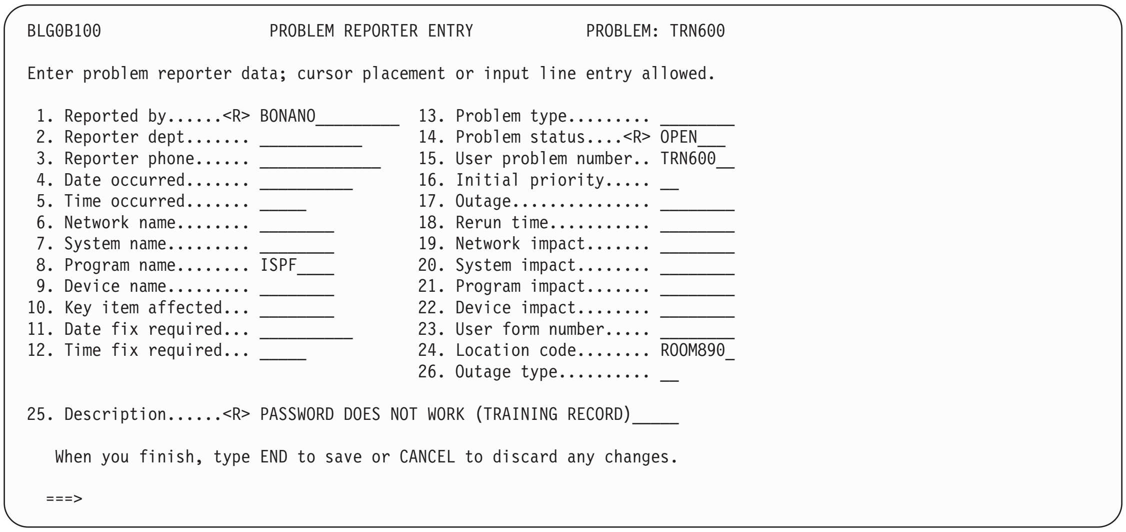
```
BLG0B100                  PROBLEM REPORTER ENTRY            PROBLEM: TRN559

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> BONANO_________  13. Problem type......... ________
 2. Reporter dept....... ___________      14. Problem status....<R> CLOSED_
 3. Reporter phone...... _____________    15. User problem number.. TRN559__
 4. Date occurred....... __________       16. Initial priority..... __
 5. Time occurred....... _____            17. Outage............... ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ ________         20. System impact........ ________
 9. Device name......... 1403____         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... __________       23. User form number..... ________
12. Time fix required... _____            24. Location code........ COMPROOM
                                          26. Outage type.......... __

25. Description......<R> GARBAGE ON MY PRINTOUTS (TRAINING RECORD)____

   When you finish, type END to save or CANCEL to discard any changes.

  ===>
```

TRN125



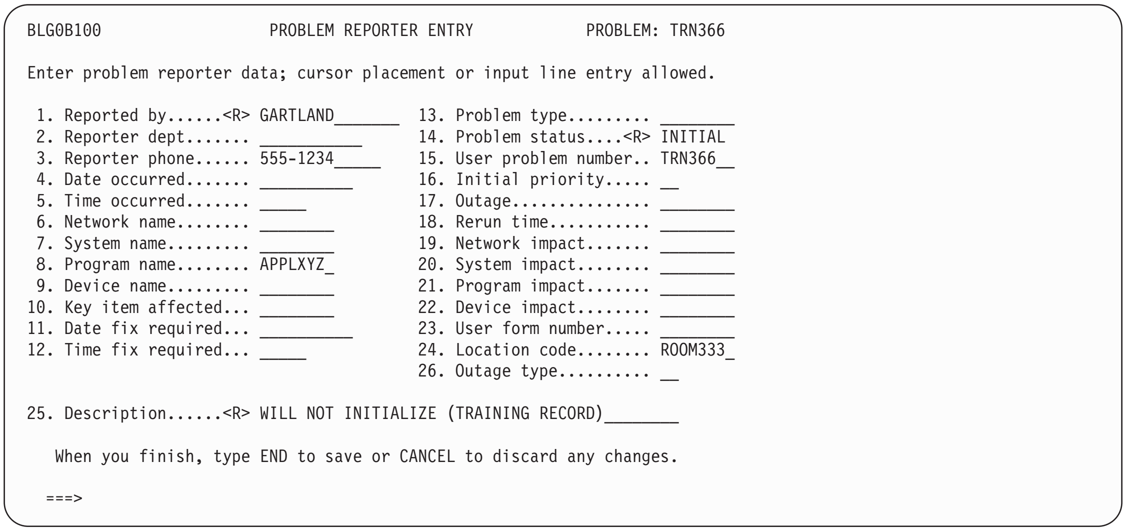
```
BLG0B100                  PROBLEM REPORTER ENTRY            PROBLEM: TRN125

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> KOWALSKI_______  13. Problem type......... ________
 2. Reporter dept....... ___________      14. Problem status....<R> OPEN___
 3. Reporter phone...... _____________    15. User problem number.. TRN125__
 4. Date occurred....... __________       16. Initial priority..... __
 5. Time occurred....... _____            17. Outage............... ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ APPLXYZ_         20. System impact........ ________
 9. Device name......... ________         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... __________       23. User form number..... ________
12. Time fix required... _____            24. Location code........ ROOM678_
                                          26. Outage type.......... __

25. Description......<R> DOES NOT FILL UP SCREEN (TRAINING RECORD)____

   When you finish, type END to save or CANCEL to discard any changes.

  ===>
```

TRN600



```
BLG0B100                  PROBLEM REPORTER ENTRY           PROBLEM: TRN600

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> BONANO_________  13. Problem type......... ________
 2. Reporter dept....... ___________      14. Problem status....<R> OPEN___
 3. Reporter phone...... _____________    15. User problem number.. TRN600__
 4. Date occurred....... __________       16. Initial priority..... __
 5. Time occurred....... _____            17. Outage............... ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ ISPF____         20. System impact........ ________
 9. Device name......... ________         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... __________       23. User form number..... ________
12. Time fix required... _____            24. Location code........ ROOM890_
                                          26. Outage type.......... __

25. Description......<R> PASSWORD DOES NOT WORK (TRAINING RECORD)_____

   When you finish, type END to save or CANCEL to discard any changes.

  ===>
```

TRN366



```
BLG0B100                  PROBLEM REPORTER ENTRY           PROBLEM: TRN366

Enter problem reporter data; cursor placement or input line entry allowed.

 1. Reported by......<R> GARTLAND_______  13. Problem type......... ________
 2. Reporter dept....... ___________      14. Problem status....<R> INITIAL
 3. Reporter phone...... 555-1234_____    15. User problem number.. TRN366__
 4. Date occurred....... __________       16. Initial priority..... __
 5. Time occurred....... _____            17. Outage............... ________
 6. Network name........ ________         18. Rerun time........... ________
 7. System name......... ________         19. Network impact....... ________
 8. Program name........ APPLXYZ_         20. System impact........ ________
 9. Device name......... ________         21. Program impact....... ________
10. Key item affected... ________         22. Device impact........ ________
11. Date fix required... __________       23. User form number..... ________
12. Time fix required... _____            24. Location code........ ROOM333_
                                          26. Outage type.......... __

25. Description......<R> WILL NOT INITIALIZE (TRAINING RECORD)________

   When you finish, type END to save or CANCEL to discard any changes.

  ===>
```

C

Your Organization's Procedures

This chapter can help you to keep track of the local procedures and conventions that your organization follows. You can record this information on the charts provided and refer to them in the future.

Before you start, record the name and telephone number of your program administrator for quick reference.

Local Contact:

Program Administrator:
____

Telephone
Number:____________________________________

Getting Started

Use the procedure below to get started on Tivoli Information Management for z/OS. Record the CLIST set up by your organization on the line provided.

Starting Tivoli Information Management for z/OS:

1. Log on to MVS.

2. Choose ISPF option number:_________________
   or
   Enter CLIST named: ____________________________________

Numbering a Record

Each record must have a unique record ID. Your organization may already have a procedure for assigning ID numbers to records. If so, write that procedure below. To find out what that procedure is, call your program administrator.

Record ID Numbering Procedure:
____

____

____

____

____

Note: If you do not assign an ID to a record, Tivoli Information Management for z/OS automatically assigns a numeric ID.

Data Entry Conventions

Just as your organization may have standard numbering procedures for records, it may also have standard procedures for entering specific kinds of data in the database.

Having a standard method for entering similar kinds of data makes it easier to search for data after it is stored in the database. For example, if you entered someone's name as JOHNSMITH, then next time entered JSMITH, and again entered SMITHJ, it would be difficult to find all records that contained the name JOHN SMITH.

Your organization recognizes the importance of having data entry conventions. Some of the panels on which you enter data have fields that accept data in a specific format and, therefore, do not let you enter data improperly. However, this type of data checking cannot be done on all types of data. Therefore, your organization has established data entry conventions for you to follow when entering data into records. Write the conventions on the next page so you can easily refer to them whenever you enter data.

Data entry conventions:

Names:

____

____

Dates:

____

____

Times:

____

____

Product names:

____

____

Machine types:

____

____

Location codes:

____

____

Department names:

____

____

Other data:

____
____
____
____
____

Deleting a Record

Your organization may have a local procedure that you should follow to delete a record from the database. If so, write that procedure here.

Local procedure for getting a record deleted:

____
____
____
____
____
____
____
____
____
____
____
____
____
____
____

Printing a Record

Your organization may have local printing defaults that are used in printing a record. If so, write those defaults here.

Local printing defaults:

____
____
____
____
____
____
____
____
____
____
____
____
____
____
____

D

Using Notification Management

Notice

The information in this appendix applies to the former methods of notification management that were used in Tivoli Information Management for z/OS. A more current method exists to perform message notification. For information, see "Sending E-mail Messages" on page 95. The information in this chapter is retained as a convenience for those customers who may still be using the former methods of notification management.

This information is for the program administrator responsible for implementing Notification Management. You must understand Terminal Simulator Panels (TSPs), Terminal Simulator EXECs (TSXs), and the Panel Modification Facility (PMF) to perform the tasks described in this chapter. For information about TSPs and TSXs, refer to the *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference*. For information about PMF, refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide*.

Notification Management is a Tivoli Information Management for z/OS function that gives the people you designate a way to monitor the status of a problem, change, or activity record from its creation to its resolution. Designated people are notified by the system for certain conditions related to the record.

Notification Management encompasses two different functions: immediate notification and problem escalation.

Immediate notification sends a message to a specified user, or group of users, every time a problem, change, or activity record is entered or updated. Additionally, you can implement immediate notification for other types of records. The Integration Facility also provides immediate notification. Refer to *Tivoli Information Management for z/OS Integration Facility Guide* for more information.

Escalation processing handles only problem records. Problem escalation updates priorities and sends messages to people identified in RULES records until a problem record is closed. Use problem escalation to alert people of unresolved problems in your data processing organization.

The two functions of Notification Management are separate and are started differently.

Other features of the Notification Management facility that you can use are:

- Use TSXs to perform notification functions. TSXs provide an easy way to construct complex mail messages and also allow for easier maintenance.

- Allow immediate notification mail and escalation messages to be sent using MVS TCP/IP Simple Mail Transfer Protocol (SMTP). Messages routed using SMTP can be processed by a locally installed electronic mail package such as Lotus Notes.
- Allow messages to be sent directly from a user's address space, or added to a BLX-SP queue from which they will subsequently be sent. Placing the messages on a queue offloads notification processing from each user's address space.
- Enable immediate notification processing for records filed through API processing.

You can select which method to use to perform immediate notification—TSXs that use TCP/IP SMTP or TSPs that use TSO Send. The TSX method constructs a mail message using data from a filed record and sends the message using MVS TCP/IP SMTP. The TSP method builds a simple mail message using a filed record ID and sends the message to a TSO user using the TSO send command. You can disable immediate notification, or enable one of these two methods. Or, you can modify either of the sample methods provided to support processing available in your local environment.

See "Disabling Immediate Notification" on page 482 to disable immediate notification. See "Immediate Notification Using TSXs and TCP/IP" to enable TSXs and TCP/IP SMTP, or "Immediate Notification Using TSPs and TSO Send" on page 461 to enable TSPs and TSO send. For information on problem escalation, see "Understanding Problem Escalation" on page 484.

Immediate Notification Using TSXs and TCP/IP

This section describes how to set up immediate notification using TSXs and TCP/IP SMTP. For details on how to enable Notification Management using TSPs and TSO Send, see "Immediate Notification Using TSPs and TSO Send" on page 461. That section describes the former traditional approach of setting up the Notification Management facility through the use of TSPs. Problem escalation is explained in "Understanding Problem Escalation" on page 484. For a discussion of immediate notification with the Integration Facility, refer to *Tivoli Information Management for z/OS Integration Facility Guide*.

Notification Processing with TSXs

Tivoli Information Management for z/OS provides sample TSXs to perform immediate notification. TSXs can provide an easier way to construct a complex mail message than using traditional TSPs. The TSXs provided can:

- Build the notification message when a record is filed.
- Send the message using TCP/IP SMTP processing or add the message to a BLX-SP queue.
- Extract the message from the BLX-SP queue and send the message using TCP/IP SMTP mail.

Tivoli Information Management for z/OS provides sample TSXs that can be invoked when problem, change, and activity records are filed. You can invoke a TSX anywhere in the panel dialogs to initiate notification processing. The sample TSXs provide a way to send a detailed mail message to the assignee of a record when it is filed. You can modify the TSXs to include different data in the message, send the message to someone other than the assignee, or use a different mail mechanism other than TCP/IP SMTP. You can also write your own TSX to notify users at times other than when filing records. For instance, you could write a TSX that would kick off notification processing when the user makes a selection.

By using TSXs, Tivoli Information Management for z/OS can easily construct the mail message using data from the filed record and provide a sample method of sending the mail using MVS TCP/IP SMTP.

An overview of the processing done by TSXs for TCP/IP SMTP mail follows:

- When a record requiring notification is filed, a TSX is invoked.
- The TSX extracts the assignee to notify and other information from the record.
- If an assignee is found, the TSX looks in the USERS record to find any mail addresses defined for that person. If no address is found, a default address is used.

  A user exit, BLGUSERS, accesses the address information in the USERS record.
- The mail message is built.
- The TSX appends TCP/IP SMTP header and trailer information to the message to put it in a format that can be processed through SMTP.
- The TSX calls a user exit, BLGNSYAL, to allocate a SYSOUT data set with a destination of SMTP.
- Once the data set is allocated, a TSX writes the message data to it. User exit BLGNSYFR frees the data set, which serves to route the message to the SMTP server. No TSO services are required, so API-generated records and records generated through interactive use can use this method of immediate notification.
- Alternatively, the TSX can send the message to a BLX-SP queue for later processing. (The default is not to queue the messages.)
- TCP/IP SMTP routes the mail using your installation's mail configuration. Once messages reach this point, Tivoli Information Management for z/OS no longer has control of the message.

Immediate Notification with TSXs

As shipped, immediate notification is performed by a set of TSPs that are called when a problem, change, or activity record is filed. The TSPs use the **Assignee Name** field in the record to determine to whom to send the notification message. If the TSPs cannot determine to whom to send the notification message, the TSPs send it to a default user ID that is specified in the TSPs. You must enable or disable immediate notification before allowing your users to create or update problem, change, or activity records. If you want to use the TSXs that are provided to handle immediate notification processing, you must also change the panels that call the TSPs to call the TSXs instead. The following section describes how to enable immediate notification so that it uses TSXs and TCP/IP SMTP. If you prefer to disable immediate notification, see “Disabling Immediate Notification” on page 482 for complete instructions.

Enabling Immediate Notification with TSXs Using TCP/IP SMTP

Before you can use immediate notification, you must perform the following steps to enable it. The TSXs supplied with Tivoli Information Management for z/OS reside in the SBLMTSX data set. You can use any MVS editor to update them. You should copy any TSX you want to modify to another data set before making modifications.

To run a TSX, you must have the BLGTSX DD allocated as described in the “Starting Tivoli Information Management for z/OS” section of the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference*.

Note: **Integration Facility users**: If you plan to use TSXs, you can generally follow steps 1 through 7, but be sure to use the TSXs defined for the Integration Facility (BTNXCN07 and BTNXPN03) as described in *Tivoli Information Management for z/OS Integration Facility Guide* instead. The examples shown in this section show the use of the BLGTXNOT TSX. You will need to make comparable changes in the Integration Facility TSXs.

1. Change the default mail addresses in the TSX. The sample TSX you would change for immediate notification is BLGTXNOT. Refer to “Changing the Default Mail Address” on page 451 for instructions.
2. Decide if you want to enable immediate or queued processing. With immediate processing, the mail message is constructed and sent when a record is filed, all within the user’s address space. With queued processing, the mail message is constructed when the record is filed, but instead of being sent immediately, it is added to one of three BLX-SP queues and later extracted from the queue and sent. Immediate processing is the default; no action is required to implement it. To implement queued processing, see “Queued Processing” on page 452 for more information.
3. Decide if you want to use TCP/IP for message routing. If you do, define the correct TCP/IP SMTP mail header and trailer information for your environment in the BLGTXNOT TSX. See “Defining TCP/IP SMTP Header and Trailer Information” on page 453 for more information. Also, you should ensure that TCP/IP is installed and that SMTP is configured to be enabled. (For more information on SMTP, refer to the *IBM TCP/IP for MVS Customization and Administration Guide, SC31-7134.*) If you do not want to use TCP/IP SMTP, modify the TSX logic to support the mail processing of your choice.
4. Define the logical record length, SYSOUT class, destination, and *ddname* of the data set dynamically allocated to TCP/IP SMTP to match your local environment. See “Defining the SMTP Data Set” on page 453 for more information.
5. Modify the **’FROM:’** mail line defined in the BLGTXNOT TSX *buildmsg* subroutine to include the mail address to use as the *from* address. See “Modifying the ″From″ Mail Line” on page 453 for more information.
6. Determine the format of the mail message to send and what record data to include in the notification message. This data could be different for each type of record. For instance, you might want to include priority data from problem records but not change records.
7. Using the Panel Modification Facility, change the following control panels to invoke the appropriate TSXs.

BLG1A111	Problem records
BLG1A121	Change records
BLG1A131	Activity records

   To determine which TSX to invoke, refer to “TSP to TSX Equivalent” on page 455 for a list of immediate notification TSXs.
8. If you haven’t already done so, create a USERS record. See “Creating the USERS Record” on page 456 for more information.

9. Perform a simple test to ensure that when a record is filed, it is routed to the proper mail address. If you did not set up the USERS record, the message should be routed to the default address.

Changing the Default Mail Address

The default mail address in the TSXs as shipped by Tivoli must be changed to enable notification messages to be routed to the program administrator (or some other designated person) if an assignee or alias is not found in the USERS record. The default mail address must be changed for each type of record that the notification TSXs process. You can also change the processing, if desired; for example, you might choose not to send a message if the filed record does not contain an assignee.

For immediate notification, change the default mail addresses in the BLGTXNOT TSX, which can be found in the SBLMTSX data set and changed using an MVS editor. The data set(s) that contain the notification TSXs must be allocated to the BLGTSX DD to invoke TSXs for immediate notification. (For information on allocating the BLGTSX DD, refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference*.)

The BLGTXNOT TSX sends the notification message for problem, change, and activity records. Determine the default address to use for each type of record, and modify this TSX to define them. The default address is used when the record does not have an assignee or when the assignee does not have any mail addresses defined in the USERS record.

This example shows where to change the default mail addresses in the BLGTXNOT TSX. The example shows an extract from the sample BLGTXNOT TSX that is shipped with Tivoli Information Management for z/OS. You would, for instance, enter a default address for problem records on the **DEFAULT_ADDRESS_PROBLEM** line. Your own TSX or other similar TSXs (such as those used with the Integration Facility or Escalation processing) may contain variations.

```
/**********************************************************************/
/* Set up constants to be used in the rest of the TSX.                */
/**********************************************************************/
/* Default address to be used if assignee not in USERS record.        */
/* Set up a default for each type of record that this TSX will be     */
/* sending notifications for.                                         */
DEFAULT_ADDRESS_PROBLEM='defaultp';
DEFAULT_ADDRESS_CHANGE='defaultc';
DEFAULT_ADDRESS_ACTIVITY='defaulta';

/* Select the processing method.  IMMEDIATE indicates to send the     */
/* mail via TCP/IP SMTP within this TSX.  QUEUE indicates to queue    */
/* the mail message to the BLX-SP using the QMAIL control line.       */
notification_type='IMMEDIATE';   /* QUEUE or IMMEDIATE                */

/* Define SYSOUT data set attributes and destination information.     */
/* If performing IMMEDIATE Notification, a user exit will use this    */
/* information to allocate a SYSOUT data set with a destination of    */
/* TCP/IP SMTP.  Then the e-mail message will be written to this      */
/* data set by the TSX.                                               */
lrecl=80;                        /* LRECL of SYSOUT data set          */
class='A';                       /* Class of SYSOUT data set          */
node='node';                     /* TCP/IP SMTP Node - used as part of
                                    destination for SYSOUT data set   */
userid='SMTP';                   /* TCP/IP SMTP USERID part of
                                    destination for SYSOUT data set   */

/* Define SMTP header and trailer lines to be included in the         */
/* e-mail message.                                                   */
/* Change the information to match your system environment.          */
hdr.1='HELO mailaddress';
hdr.2='MAIL FROM:<mailaddress>';
hdr.3='RCPT TO:<';
hdr.4='DATA';
hdr.0=4

trl.1='.';
trl.2='QUIT';
trl.0=2;
/********************************************************************/
```

Queued Processing

Immediate processing, where a mail message is constructed and sent when a record is filed, is the default type of mail processing that is specified in the BLGTXNOT TSX shipped with Tivoli Information Management for z/OS. If you would like to reduce the time a user must wait for a record to be filed, you can defer some of the mail processing to a later time without directly impacting the user. BLGTXNOT can build a mail message and add it to a BLX-SP queue. You can alter this logic to minimize the processing performed when the record is filed. For example, you could place just the record ID of the record being filed onto the BLX-SP queue. When the record ID is later pulled off the queue, the mail message could be built and sent at that time.

You can define up to three BLX-SP queues to contain mail messages. You can specify the queue names and limits on the number of items that can be placed onto each queue. For instructions on how to set the MAILQ parameter of the BLX-SP parameters member to specify names of mail queues and queue limits, refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference*. To query the BLX-SP to find out how many mail items exist on the queues or to modify the queue limits, refer to the *Tivoli Information Management for z/OS Operation and Maintenance Reference* for a description of the MAILQ operator command.

To use queued processing, you must modify the BLGTXNOT TSX to specify **QUEUE** as the **notification_type** (an example of BLGTXNOT is shown in the previous example. You must also use one or more TSXs to extract the mail from the queue(s) and send the mail. These TSXs can be run using Tivoli Information Management for z/OS in batch or interactively. Sample TSX BLGTXDEQ shows how to dequeue and send mail using TCP/IP SMTP.

When the shutdown command is issued for the BLX-SP, the system checks to see if there are any mail items on any of the queues. If there are, a message is issued indicating that the shutdown will halt until the queues are empty or the wait time is exceeded.

Use of a message queue will impact the processing of the BLX-SP in two ways:

- The BLX-SP will have more work to do because it will place items on and off the mail queues.
- At shutdown, if the **MAILQWAITTM** parameter is specified in the session-parameters member, processing is suspended until either all of the mail queues are empty or the wait time has been exceeded. If you do not specify MAILQWAITTM, shutdown occurs immediately regardless of the status of the mail queues.

You can, however, specify how long the BLX-SP will wait upon shutdown for mail to clear off of the queues. The wait time is broken into 10 second intervals. If the queues are not cleared by the time you specify in the **MAILQWAITTM** parameter in the BLXPRM startup, the queues are flushed. The mail items are lost and shutdown processing continues.

Therefore, if you need to stop the BLX-SP, you may want to wait until the mail is processed. Once the shutdown starts, there is no way to query the number of items on the queues, change the wait time, or stop the shutdown.

Refer to the *Tivoli Information Management for z/OS Planning and Installation Guide and Reference* for more information on the BLX-SP parameters member and use of the **MAILQWAITTM** parameter.

Defining TCP/IP SMTP Header and Trailer Information

As part of the overall setup to enable the Notification Management facility to route messages using MVS TCP/IP SMTP, you must define the correct SMTP mail header and trailer information for your environment.

The header and trailer information is defined in the BLGTXNOT TSX. (See the code starting at the **HDR.1** line in the example on page 451.) After the BLGTXNOT TSX looks in the USERS record for mail addresses defined to the assignee, it builds the mail message and includes this SMTP header and trailer information. Without this data, the mail messages will not be routed successfully.

For more information on SMTP, refer to the *IBM TCP/IP V3R1 for MVS Customization and Administration Guide, SC31-7134.*

Defining the SMTP Data Set

As part of the immediate notification processing performed by BLGTXNOT, a SYSOUT data set is dynamically allocated to the destination of SMTP. The BLGTXNOT TSX writes the mail message to this data set, which sends the mail to the SMTP server, and frees the data set.

You must ensure that the BLGTXNOT TSX contains the correct data set attributes and destination information for the SYSOUT data set. The logical record length, SYSOUT class, and destination (TCP/IP SMTP node and userid) of the data set dynamically allocated to SMTP should match values required by your local environment.

The example on page 451 shows where these definitions occur in the sample BLGTXNOT TSX.

Modifying the ″From″ Mail Line

The BLGTXNOT TSX must be modified to contain an address indicating where mail messages are coming *from*. This information is included in the mail message.

Refer to the following example to see where you would change the **From mail address** in the BLGTXNOT TSX (see **call addmsg(’From: mailaddress’);**). The **From mail address** can refer to your Tivoli Information Management for z/OS system.

```
/*******************************************************************/
/* Add the message data to the stemmed variable.                   */
/*******************************************************************/
call addmsg('Date:         'date() time());
call addmsg('From:         mailaddress');
if recordaction='CREATE' then
  lastpart='created.'
else
  lastpart='updated.'
call addmsg('Subject:      'recordtype 'record' rnid lastpart);
do b=1 to blg_address.0;
  call addmsg('To:           <'blg_address.b'>');
end;
call addmsg(' ');
call addmsg(recordtype 'record filed in Information Management for z/OS.');
call addmsg(' ');
call addmsg('Record ID:          'rnid);
call addmsg('Reporter/requester: 'reporter);
call addmsg('Date/Time:          'date() time());
call addmsg('Assignee:           'assignee);
call addmsg('Status:             'status);
call addmsg('Description:        'description);
call addmsg('Priority:           'priority);
call addmsg(' ');
call addmsg('Description Text:');
do a=1 to descr_text.0;
  if descr_text.a=' then
    descr_text.a=' '
  call addmsg('    'descr_text.a);
end;
```

Figure 37. Portion of the BLGTXNOT TSX Shipped with Tivoli Information Management for z/OS

Defining the SMTP Data Set for Use with Dequeue Processing

Tivoli Information Management for z/OS provides a TSX, BLGTXDEQ, which shows how to dequeue mail messages from the BLX-SP queue, if message queuing is selected rather than processing in the user's address space. If you elect to queue mail messages, you must also modify the BLGTXDEQ TSX or the TSX(s) you are using to dequeue and process mail to define the SMTP data set attributes and destination. The definition should include the SYSOUT class and destination (TCP/IP SMTP node and userid) for your environment.

Additionally, you must modify the DEQMAIL parameters in the BLGTXDEQ TSX to match the queue name used in your environment.

Changing the Message Format

As part of the setup procedure to enable immediate notification with TSXs and TCP/IP SMTP, you must determine the format of the mail message to send, and what data to include from the record in the notification message. The contents could be different for each type of record, and additional data can be added.

To determine what to include in the mail message, first review the sample BLGTXNOT TSX provided with Tivoli Information Management for z/OS to become familiar with its function and the message content that would normally be provided (see Figure 37). For example, it contains message data such as the record ID, reporter, date, assignee, status, description, and priority. Then, decide what alterations are required for your environment and make the necessary changes in your TSX.

TSP to TSX Equivalent

As part of your setup procedure to enable notification processing with TSXs, you need to modify panels to invoke the correct TSX. Use the following tables to determine what changes to make. For example, modify the BLG1A111 panel to invoke the BLGTSXPE TSX for problem creation notification, rather than the BLGTSPPE TSP.

Table 42. Tivoli Information Management for z/OS Immediate Notification TSPs and TSX Equivalents

TSP name	TSX name	Purpose	Invoked by
BLGTSPPE	BLGTSXPE	Problem create notification	BLG1A111
BLGTSPPU	BLGTSXPU	Problem update notification	BLG1A111
BLGTSPCE	BLGTSXCE	Change create notification	BLG1A121
BLGTSPCU	BLGTSXCU	Change update notification	BLG1A121
BLGTSPAE	BLGTSXAE	Activity create notification	BLG1A131
BLGTSPAU	BLGTSXAU	Activity update notification	BLG1A131

A separate TSX is still called for each record type and record action, but the TSX calls the BLGTXNOT TSX passing the type of record and the record action, as shown in the next table.

Table 43 summarizes other immediate notification TSXs that are shipped with Tivoli Information Management for z/OS.

Table 43. Other Significant Tivoli Information Management for z/OS Immediate Notification TSXs

TSX name	Purpose	Processing	Invoked by
BLGTXNOT	Perform notification for all record types.	Read record just filed and extract information for mail message. Send message through TCP/IP SMTP or queue message to BLX-SP for later processing.	TSXs BLGTSXPE, BLGTSXPU, BLGTSXCE, BLGTSXCU, BLGTSXAE, BLGTSXAU
BLGTXDEQ	Sample TSX that shows dequeuing mail built using BLGTXNOT.	Dequeues the mail and sends it using TCP/IP SMTP.	n/a

Creating the USERS Record

Before you can use immediate notification effectively, you must create a USERS record. Having a USERS record prevents a default user ID from receiving all notification messages. You can have only one USERS record.

The USERS record you create must contain:

- The names (individual users or groups of users, such as departments) that are possible assignees for a problem, change, or activity record.

  You can also specify a distribution list which exists on another mail system, such as Lotus Notes.

  A USERS record can contain up to 19 274 names of assignees or aliases. (Aliases are typically used for Escalation message notification, or with the Integration Facility, where you might have a list of people such as problem reviewers or change committee members who would need to be notified of record changes.)

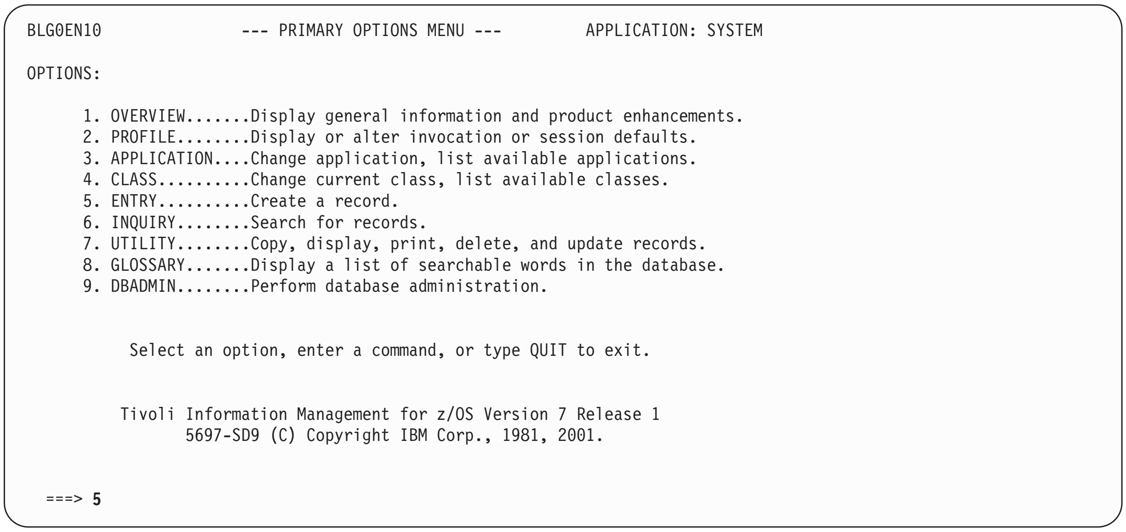
- The mail address associated with each name (this could be an MVS userid, or an electronic mail address such as a Lotus Notes ID or Internet address). If you choose to use TCP/IP SMTP, this address must be capable of being processed by TCP/IP SMTP.

Immediate notification takes the assignee name information from the record being filed, locates it in the USERS record, and sends a notification message to the corresponding mail address or addresses. If it is unable to determine to whom to send the notification message, it sends the message to the default user ID that you specified in the TSX.

The sample immediate notification TSXs assume that the mail addresses are contained in the USERS record. You can choose to use some other method to store the mail addresses and modify the TSX logic accordingly. Depending on your environment, you may be able to make the assignee name itself a mail address in the target mail system, thus eliminating the address look-up step.

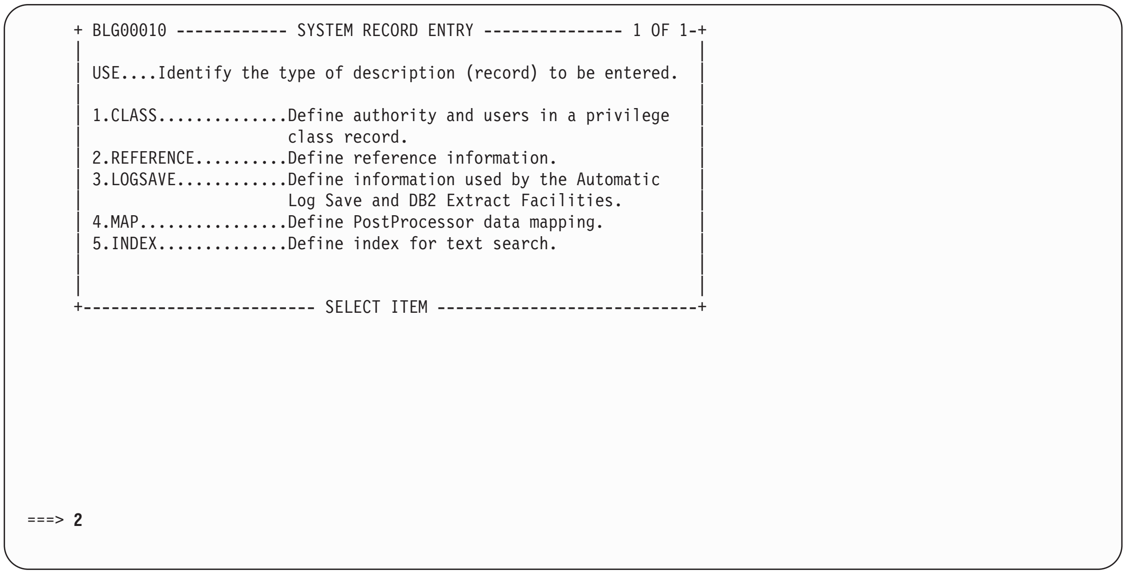
User exit BLGUSERS can be called from a TSX to extract mail addresses from the USERS record. More information about BLGUSERS can be found on page 513.

To create the USERS record, on the Primary Options Menu for the System application, select **Entry**. Type **5** and press **Enter**.



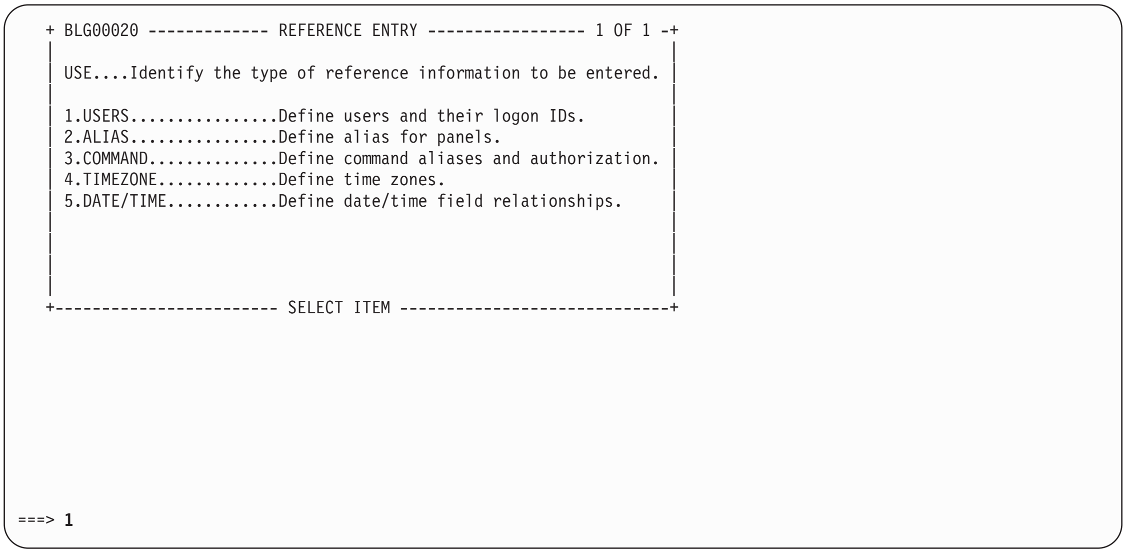
```
BLG0EN10              --- PRIMARY OPTIONS MENU ---         APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


  ===> 5
```

The USERS record is reference information. From this panel, select **Reference**. Type **2** and press **Enter**.



```
       + BLG00010 ------------ SYSTEM RECORD ENTRY --------------- 1 OF 1-+
       |                                                                  |
       | USE....Identify the type of description (record) to be entered.  |
       |                                                                  |
       | 1.CLASS.............Define authority and users in a privilege    |
       |                     class record.                                |
       | 2.REFERENCE.........Define reference information.                |
       | 3.LOGSAVE...........Define information used by the Automatic     |
       |                     Log Save and DB2 Extract Facilities.         |
       | 4.MAP...............Define PostProcessor data mapping.           |
       | 5.INDEX.............Define index for text search.                |
       |                                                                  |
       |                                                                  |
       +------------------------- SELECT ITEM ----------------------------+







===> 2
```

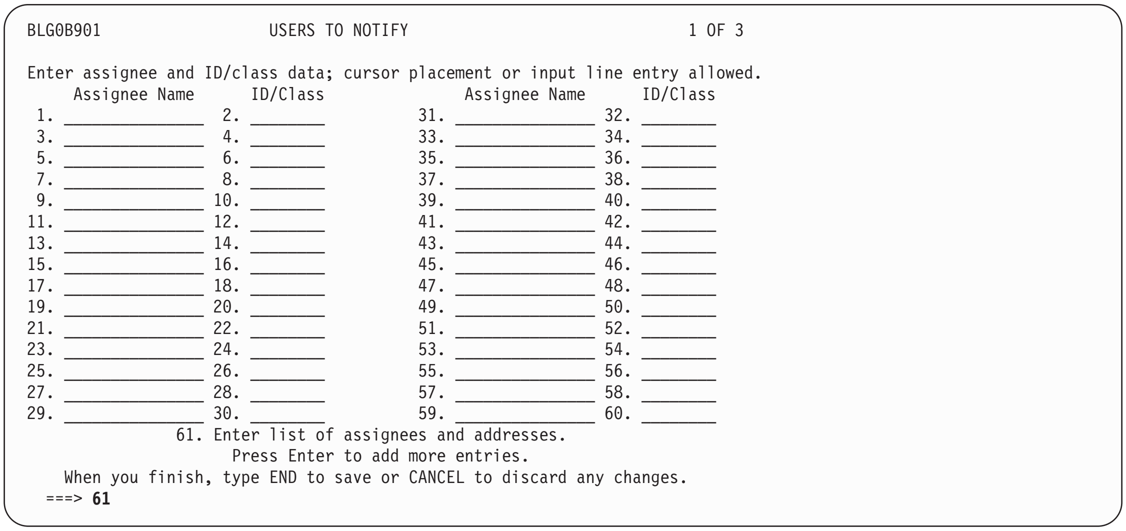
To create the USERS record, type **1** and press **Enter**.



```
  + BLG00020 ------------- REFERENCE ENTRY ----------------- 1 OF 1 -+
  |                                                                  |
  | USE....Identify the type of reference information to be entered. |
  |                                                                  |
  | 1.USERS................Define users and their logon IDs.          |
  | 2.ALIAS................Define alias for panels.                   |
  | 3.COMMAND..............Define command aliases and authorization.  |
  | 4.TIMEZONE.............Define time zones.                         |
  | 5.DATE/TIME............Define date/time field relationships.      |
  |                                                                  |
  |                                                                  |
  |                                                                  |
  |                                                                  |
  +------------------------ SELECT ITEM -----------------------------+









===> 1
```

The USERS record is assigned a record ID of USERS.

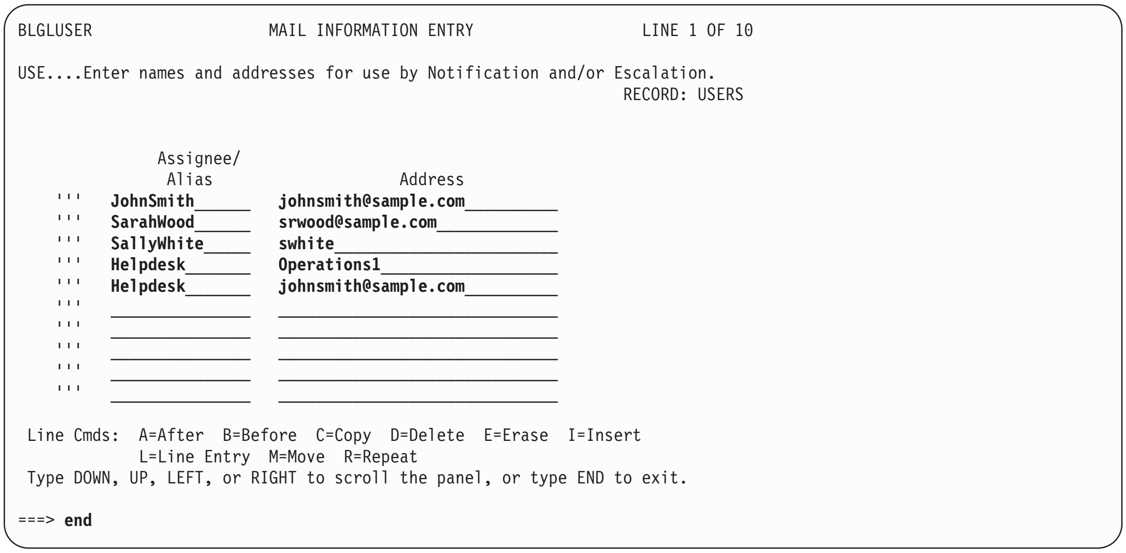
The standard Tivoli Information Management for z/OS USERS TO NOTIFY panel appears. Because many electronic addresses may be too long to fit in the ID/Class column and because user exit BLGUSERS cannot extract addresses entered into the ID/Class fields, rather than use this standard panel, type in option **61** and press **Enter** to get the list processor panel for mail information entry.



```
BLG0B901                 USERS TO NOTIFY                          1 OF 3

Enter assignee and ID/class data; cursor placement or input line entry allowed.
     Assignee Name      ID/Class               Assignee Name      ID/Class
 1. _______________  2. ________         31. _______________ 32. ________
 3. _______________  4. ________         33. _______________ 34. ________
 5. _______________  6. ________         35. _______________ 36. ________
 7. _______________  8. ________         37. _______________ 38. ________
 9. _______________ 10. ________         39. _______________ 40. ________
11. _______________ 12. ________         41. _______________ 42. ________
13. _______________ 14. ________         43. _______________ 44. ________
15. _______________ 16. ________         45. _______________ 46. ________
17. _______________ 18. ________         47. _______________ 48. ________
19. _______________ 20. ________         49. _______________ 50. ________
21. _______________ 22. ________         51. _______________ 52. ________
23. _______________ 24. ________         53. _______________ 54. ________
25. _______________ 26. ________         55. _______________ 56. ________
27. _______________ 28. ________         57. _______________ 58. ________
29. _______________ 30. ________         59. _______________ 60. ________
                61. Enter list of assignees and addresses.
                      Press Enter to add more entries.
    When you finish, type END to save or CANCEL to discard any changes.
  ===> 61
```

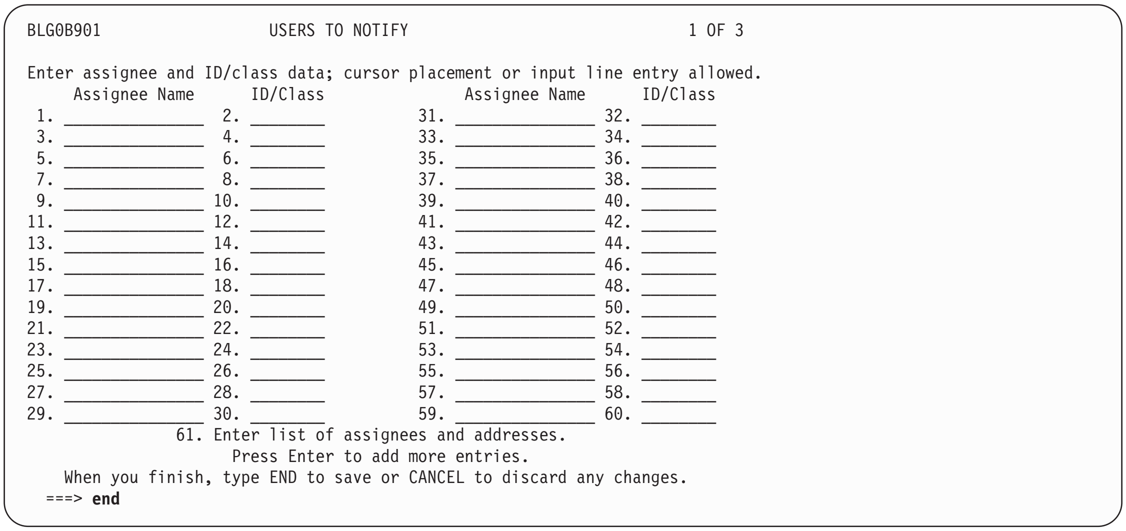
Enter the assignee/alias names and addresses of the individuals or groups who are to receive immediate notification messages. The assignee names can be 1 to 15 alphanumeric positions and can contain #, @, $, &,; and / characters. (You can also enter a privilege class as the assignee name, but you must repeat the entry and type in the address of each person in the privilege class.) The mail address can be 1 to 60 characters. Because the **Address** field can contain 60 characters, only part of the field is shown on the panel. If your data is longer than what you can type on this panel, you can type an **L2** (for Address) in the line command area to the left of the row that you want to update, and press **Enter** to go to the assisted-entry panel where you can enter the complete address. Or, you can use the RIGHT and LEFT scroll commands to scroll the **Address** column to enter more data.

After you enter the data, type **end** and press **Enter** to return to the USERS TO NOTIFY panel.



```
BLGLUSER                   MAIL INFORMATION ENTRY                  LINE 1 OF 10

USE....Enter names and addresses for use by Notification and/or Escalation.
                                                                 RECORD: USERS


               Assignee/
                Alias                    Address
    '''   JohnSmith______   johnsmith@sample.com__________
    '''   SarahWood______   srwood@sample.com_____________
    '''   SallyWhite_____   swhite________________________
    '''   Helpdesk_______   Operations1___________________
    '''   Helpdesk_______   johnsmith@sample.com__________
    '''   _______________   ______________________________
    '''   _______________   ______________________________
    '''   _______________   ______________________________
    '''   _______________   ______________________________
    '''   _______________   ______________________________

 Line Cmds:  A=After  B=Before  C=Copy  D=Delete  E=Erase  I=Insert
             L=Line Entry  M=Move  R=Repeat
 Type DOWN, UP, LEFT, or RIGHT to scroll the panel, or type END to exit.

===> end
```

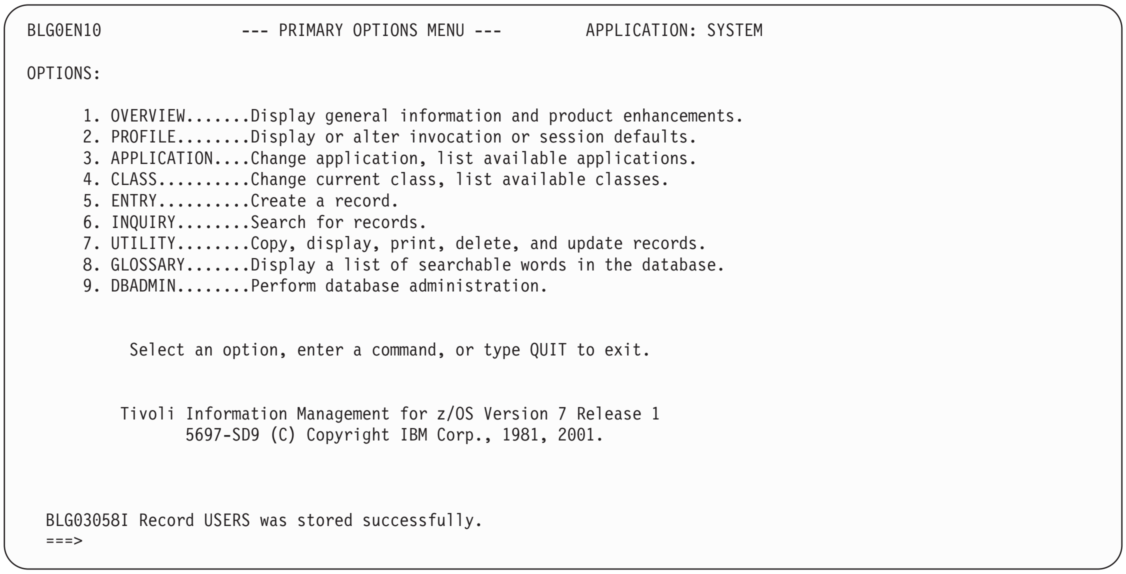
Type **end** again on the USERS TO NOTIFY panel to file your entries.



```
BLG0B901                 USERS TO NOTIFY                          1 OF 3

Enter assignee and ID/class data; cursor placement or input line entry allowed.
     Assignee Name      ID/Class              Assignee Name      ID/Class
 1. _______________  2. ________         31. _______________ 32. ________
 3. _______________  4. ________         33. _______________ 34. ________
 5. _______________  6. ________         35. _______________ 36. ________
 7. _______________  8. ________         37. _______________ 38. ________
 9. _______________ 10. ________         39. _______________ 40. ________
11. _______________ 12. ________         41. _______________ 42. ________
13. _______________ 14. ________         43. _______________ 44. ________
15. _______________ 16. ________         45. _______________ 46. ________
17. _______________ 18. ________         47. _______________ 48. ________
19. _______________ 20. ________         49. _______________ 50. ________
21. _______________ 22. ________         51. _______________ 52. ________
23. _______________ 24. ________         53. _______________ 54. ________
25. _______________ 26. ________         55. _______________ 56. ________
27. _______________ 28. ________         57. _______________ 58. ________
29. _______________ 30. ________         59. _______________ 60. ________
              61. Enter list of assignees and addresses.
                    Press Enter to add more entries.
    When you finish, type END to save or CANCEL to discard any changes.
  ===> end
```

The task of creating a USERS record is finished. A message tells you that the record was stored.

As more people become targets of notification, you must add their names and mail addresses to the USERS record.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                5697-SD9 (C) Copyright IBM Corp., 1981, 2001.



  BLG03058I Record USERS was stored successfully.
  ===>
```

Because you can have only one USERS record, this record cannot be copied and it does not appear on inquiry panel flows. The record can, however, be displayed, updated, printed, or deleted. If you attempt to create another USERS record, you are taken into update mode for the existing USERS record.

Note: The BLGLUSER panel is intended for use with e-mail notification and the use of TSXs. It is not recommended that you mix assignees on the BLG0B901-3 panel(s) with assignees on the BLGLUSER panel. You should use one or the other method of entering the assignee names. If you decide to use TSXs, you should delete the USERS record you previously created using TSPs (if you had one), and create it again using the BLGLUSER panel. To make entry of the new record easier, you should issue the PRINT R USERS command to print the old USERS record first before deleting it.

Immediate Notification Using TSPs and TSO Send

This section describes how to enable immediate notification using the standard TSP approach. For information on how to set up immediate notification using TSXs, refer to “Immediate Notification Using TSXs and TCP/IP” on page 448. You can enable only one of these approaches. If you are using TSXs and TCP/IP SMTP, ignore this section.

The method of immediate notification described in this section is one that is controlled by a set of TSPs that are called when a problem, change, or activity record is filed. The TSPs use the **Assignee Name** field in the record to determine to whom to send the notification message. If the TSPs cannot determine to whom to send the notification message, the TSPs send it to a default user ID that is specified in the TSPs. You must enable or disable immediate notification before allowing your users to create or update problem, change, or activity records. The following describes how to enable the standard TSP method of doing immediate notification. If you prefer to disable immediate notification, see “Disabling Immediate Notification” on page 482 for complete instructions.

Enabling Immediate Notification

Before you can use immediate notification, you must perform the following steps to enable the function:

1. Delete the BRANCH statement from the TSPs marked for immediate notification in the following lists.
2. Change the default user ID in the same TSPs.
3. Create a USERS record.

Depending on the needs of your organization, you may also want to alter some of the TSPs for immediate notification. To enable immediate notification, change the TSPs indicated by an asterisk (*) in the following list.

- The TSPs used for immediate notification in Problem Management are:

  BLGTSPPE*
  Send notification that a problem record was entered.

  BLGNOTPE
  Called by BLGTSPPE when the ID/Class associated with the **Assignee name** field in the problem record is a privilege class.

  BLGTSPPU*
  Send notification that a problem record was updated.

  BLGNOTPU
  Called by BLGTSPPU when the ID/Class associated with the **Assignee name** field in the problem record is a privilege class.

- The TSPs used for immediate notification in Change Management are:

  BLGTSPCE*
  Send notification that a change record was entered.

  BLGNOTCE
  Called by BLGTSPCE when the ID/Class associated with the **Assignee name** field in the change record is a privilege class.

  BLGTSPCU*
  Send notification that a change record was updated.

  BLGNOTCU
  Called by BLGTSPCU when the ID/Class associated with the **Assignee name** field in the change record is a privilege class.

- The TSPs used for immediate notification of activity records are:

  BLGTSPAE*
  Send notification that an activity record was entered.

  BLGNOTAE
  Called by BLGTSPAE when the ID/Class associated with the **Assignee name** field in the activity record is a privilege class.

  BLGTSPAU*
  Send notification that an activity record was updated.

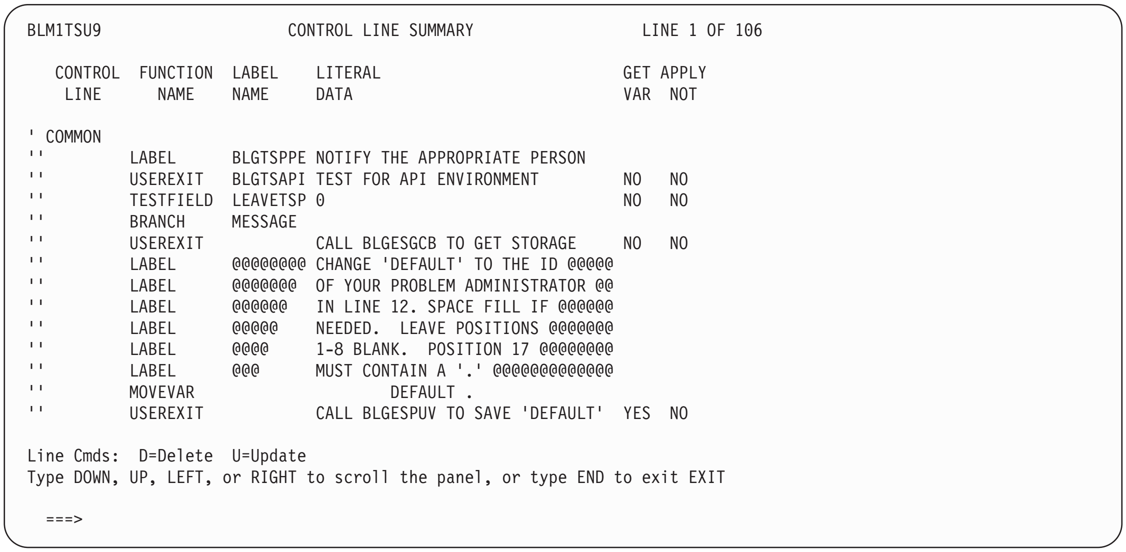
  BLGNOTAU
  Called by BLGTSPAU when the ID/Class associated with the **Assignee name** field in the activity record is a privilege class.

Deleting the BRANCH Statements

Using PMF, delete the **BRANCH** control line that branches to the **LABEL** control line called **MESSAGE** in the following TSPs:

- For Problem records, change TSPs BLGTSPPE and BLGTSPPU
- For Change records, change TSPs BLGTSPCE and BLGTSPCU
- For Activity records, change TSPs BLGTSPAE and BLGTSPAU

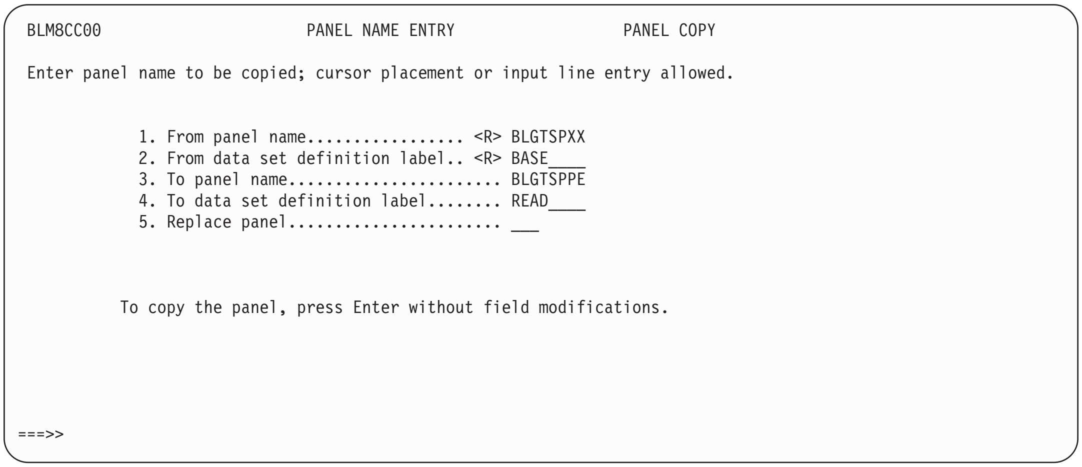
For example, this panel shows the first 14 lines of BLGTSPPE. The fifth line is the **BRANCH** control line you delete.



```
BLM1TSU9                    CONTROL LINE SUMMARY                  LINE 1 OF 106

   CONTROL  FUNCTION  LABEL    LITERAL                           GET APPLY
    LINE      NAME    NAME     DATA                              VAR  NOT

' COMMON
''         LABEL      BLGTSPPE NOTIFY THE APPROPRIATE PERSON
''         USEREXIT   BLGTSAPI TEST FOR API ENVIRONMENT          NO   NO
''         TESTFIELD  LEAVETSP 0                                 NO   NO
''         BRANCH     MESSAGE
''         USEREXIT            CALL BLGESGCB TO GET STORAGE      NO   NO
''         LABEL      @@@@@@@@ CHANGE 'DEFAULT' TO THE ID @@@@@
''         LABEL      @@@@@@@  OF YOUR PROBLEM ADMINISTRATOR @@
''         LABEL      @@@@@@   IN LINE 12. SPACE FILL IF @@@@@@
''         LABEL      @@@@@    NEEDED.  LEAVE POSITIONS @@@@@@@
''         LABEL      @@@@     1-8 BLANK.  POSITION 17 @@@@@@@@
''         LABEL      @@@      MUST CONTAIN A '.' @@@@@@@@@@@@@
''         MOVEVAR                     DEFAULT .
''         USEREXIT            CALL BLGESPUV TO SAVE 'DEFAULT'  YES  NO

Line Cmds:  D=Delete  U=Update
Type DOWN, UP, LEFT, or RIGHT to scroll the panel, or type END to exit EXIT

  ===>
```

The notification TSPs are changed for Version 7.1 to take advantage of the enhanced e-mail message notification. A copy of TSP BLGTSPPE as it appeared prior to these changes has been shipped as TSP BLGTSPXX. Before you change BLGTSPPE, follow the steps below to copy BLGTSPPE into your modified panel data set from TSP BLGTSPXX. The changes for the other notification TSPs are similar.

1. On BLG0EN20, the Primary Options Menu, type **9** on the command line to go into PMF and press Enter.
2. On BLM8C000, type **3** and press Enter to copy a panel.
3. On BLM8CC00, the Panel Name Entry panel, type BLGTSPXX in the **From panel name** field and type the name of your *base* panel data set in the **From data set definition label** field. Type BLGTSPPE in the **To panel name** field and type the name of your read panel data set in the **To data set definition label** field. Press Enter twice to cause panel BLGTSPXX to be copied to your read panel data set. You will receive a message confirming that the panel was copied.



```
 BLM8CC00                      PANEL NAME ENTRY                 PANEL COPY

 Enter panel name to be copied; cursor placement or input line entry allowed.


            1. From panel name................ <R> BLGTSPXX
            2. From data set definition label.. <R> BASE____
            3. To panel name...................... BLGTSPPE
            4. To data set definition label........ READ____
            5. Replace panel...................... ___



          To copy the panel, press Enter without field modifications.





===>>
```

Changing the Default User ID

In the same TSPs you just changed, you must also change the default user ID. The default user ID in the TSPs as shipped by Tivoli is **DEFAULT** (see the MOVEVAR control line in the previous screen). You must change **DEFAULT** to the MVS user ID of the program administrator who is responsible for notification management. A comment in each of these TSPs helps you identify where to make the changes.

The notification message is sent to the default ID if any one of these conditions is true:

- The database has no USERS record.
- The problem, change, or activity record has no assignee name.
- The assignee name in the problem, change, or activity record is not in the USERS record.
- The assignee name is found in the USERS record, but the **ID/Class** field for that name is blank.
- The assignee name identifies a privilege class record, but the user filing the problem, change, or activity record does not have privilege class display authority.

When the **ID/Class** field for the specified assignee name is found in the USERS record, it is identified either as a privilege class or as an individual.

- If the **ID/Class** is a privilege class record, a message is sent to each of the first 20 users listed in the privilege class record. (TSO SEND restricts you to sending messages to only 20 users.)
- If the **ID/Class** field is not a privilege class record, it is assumed to be an individual, and the notification message is sent to that user ID.

Note: If an assignee name is listed more than once in the USERS record, the notification message is sent to the **ID/Class** field associated with the last occurrence of that name in the USERS record.

Creating the USERS Record

Before you can use immediate notification effectively, you must create a USERS record. Having a USERS record prevents a default user ID from receiving all notification messages. You can have only one USERS record in the database.

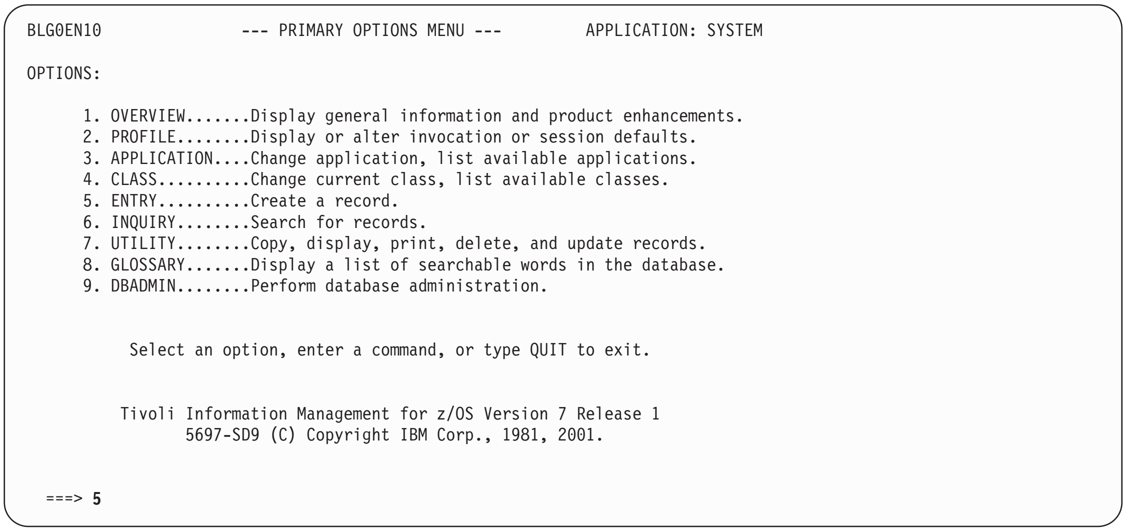
The privilege class is used when the assignee name represents a group. If you use privilege classes to secure your data, the people creating and updating problem, change, or activity records must have display authority for privilege class records.

The USERS record you create must contain:

- The names (individual users or groups of users, such as departments) that are possible assignees for a problem, change, or activity record. A USERS record can contain up to 90 names.
- The MVS user ID or privilege class associated with each name.

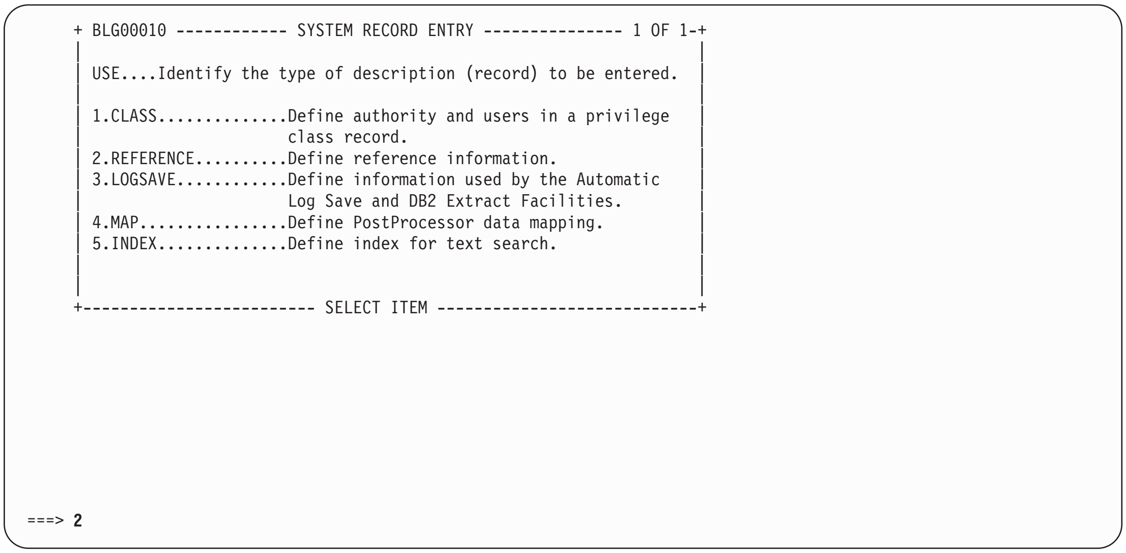
Immediate notification takes the assignee name information from the record being filed, locates it in the USERS record, and sends a notification message to the corresponding MVS user ID, or to the first 20 eligible users in the privilege class. If it is unable to determine to whom to send the notification message, it sends the message to the default user ID that you specified in the TSP.

On the Primary Options Menu for the System application, select **Entry**. Type **5** and press **Enter**.



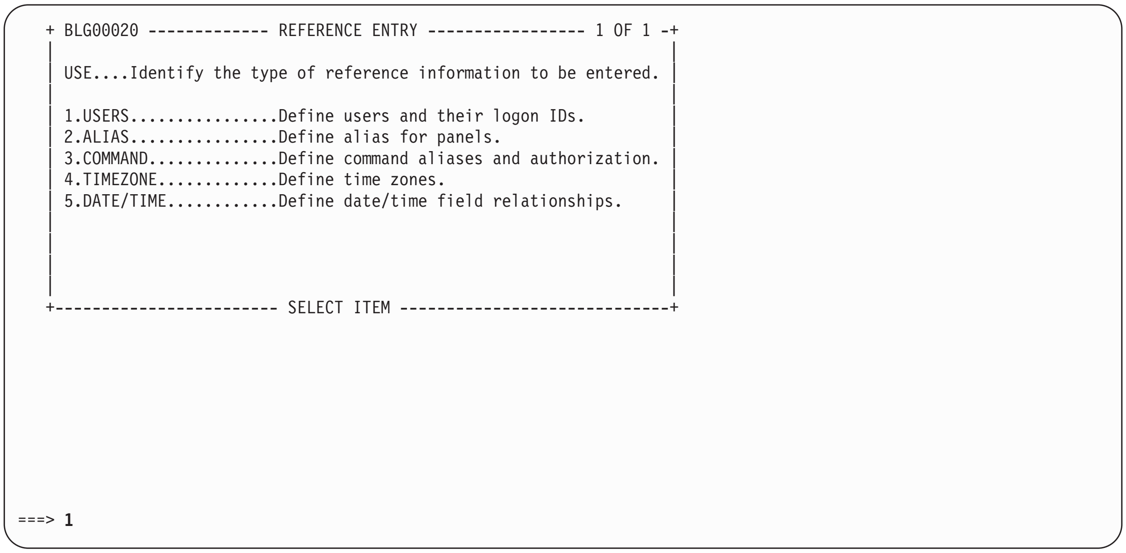
```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


  ===> 5
```

The USERS record is reference information. From this panel, select **Reference**. Type **2** and press **Enter**.



```
      + BLG00010 ------------ SYSTEM RECORD ENTRY --------------- 1 OF 1-+
      |                                                                  |
      | USE....Identify the type of description (record) to be entered.  |
      |                                                                  |
      | 1.CLASS.............Define authority and users in a privilege    |
      |                     class record.                                |
      | 2.REFERENCE.........Define reference information.                |
      | 3.LOGSAVE...........Define information used by the Automatic     |
      |                     Log Save and DB2 Extract Facilities.         |
      | 4.MAP...............Define PostProcessor data mapping.           |
      | 5.INDEX.............Define index for text search.                |
      |                                                                  |
      |                                                                  |
      +------------------------- SELECT ITEM ----------------------------+








 ===> 2
```

To create the USERS record, select **Users**. Type **1** and press **Enter**.



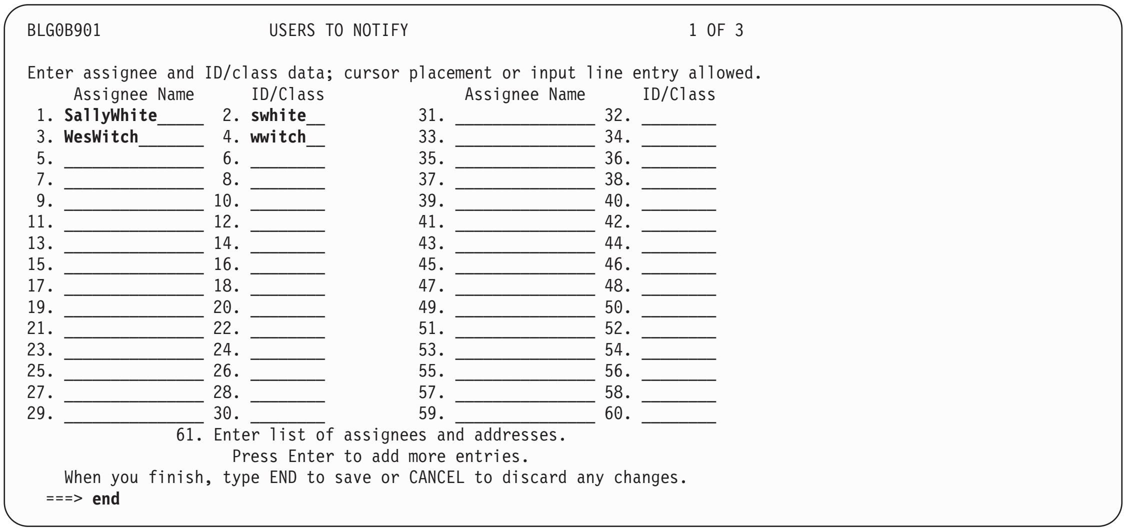
```
    + BLG00020 ------------- REFERENCE ENTRY ----------------- 1 OF 1 -+
    |                                                                  |
    | USE....Identify the type of reference information to be entered. |
    |                                                                  |
    | 1.USERS...............Define users and their logon IDs.          |
    | 2.ALIAS...............Define alias for panels.                   |
    | 3.COMMAND.............Define command aliases and authorization.  |
    | 4.TIMEZONE............Define time zones.                         |
    | 5.DATE/TIME...........Define date/time field relationships.      |
    |                                                                  |
    |                                                                  |
    |                                                                  |
    |                                                                  |
    +------------------------ SELECT ITEM -----------------------------+








===> 1
```

The USERS record is assigned a record ID of USERS.

Enter the assignee names and user IDs or privilege classes of all individuals or groups whom you want to receive immediate notification messages.

You can include 30 names on this panel. If you want to make additional entries, press **Enter** after completing this panel, and a second entry panel appears. A third panel appears if you press **Enter** again. The USERS record can contain a maximum of 90 names as shipped.

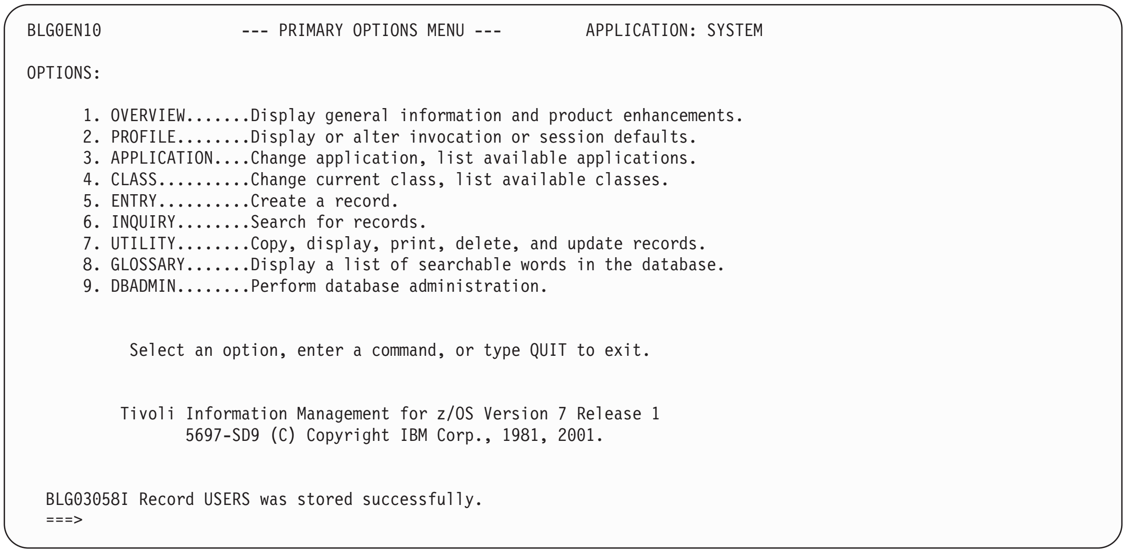
After you enter all of the assignee names and ID/Classes for your organization, type **end** and press **Enter** to file the record.



```
BLG0B901                USERS TO NOTIFY                           1 OF 3

Enter assignee and ID/class data; cursor placement or input line entry allowed.
     Assignee Name      ID/Class              Assignee Name      ID/Class
 1. SallyWhite_____  2. swhite__         31. ______________ 32. ________
 3. WesWitch_______  4. wwitch__         33. ______________ 34. ________
 5. ______________   6. ________         35. ______________ 36. ________
 7. ______________   8. ________         37. ______________ 38. ________
 9. ______________  10. ________         39. ______________ 40. ________
11. ______________  12. ________         41. ______________ 42. ________
13. ______________  14. ________         43. ______________ 44. ________
15. ______________  16. ________         45. ______________ 46. ________
17. ______________  18. ________         47. ______________ 48. ________
19. ______________  20. ________         49. ______________ 50. ________
21. ______________  22. ________         51. ______________ 52. ________
23. ______________  24. ________         53. ______________ 54. ________
25. ______________  26. ________         55. ______________ 56. ________
27. ______________  28. ________         57. ______________ 58. ________
29. ______________  30. ________         59. ______________ 60. ________
               61. Enter list of assignees and addresses.
                     Press Enter to add more entries.
    When you finish, type END to save or CANCEL to discard any changes.
  ===> end
```

Note: Option 61 is used for entering data for the TSX method of immediate notification using TCP/IP SMTP. For instructions on using option 61 and TSXs, refer instead to “Creating the USERS Record” on page 456 for information.

The task of creating a USERS record is finished. A message tells you that the record was stored.



```
BLG0EN10              --- PRIMARY OPTIONS MENU ---        APPLICATION: SYSTEM

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. DBADMIN........Perform database administration.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                5697-SD9 (C) Copyright IBM Corp., 1981, 2001.


  BLG03058I Record USERS was stored successfully.
  ===>
```

Because you can have only one USERS record, this record cannot be copied and it does not appear on inquiry panel flows. The record can, however, be displayed, updated, printed, or deleted. If you attempt to create another USERS record, you are taken into update mode for the existing USERS record. You can, however, expand the USERS record to hold additional names. The next section tells you how to do this.

Expanding the USERS Record

Your organization may need more than the 90 names you can list in the USERS record. You can expand the USERS record to include additional names.

Assume you need to add 60 additional names. To do this, you must create a new, unique s-word for every field you add. Because each name includes two fields, **Assignee name** and **ID/Class**, you must add a total of 120 new s-words to the dictionary to add 60 new names to a USERS record. Refer to the *Tivoli Information Management for z/OS Panel Modification Facility Guide* for complete instructions on adding new s-words to the dictionary.

You must modify 3 panels to allow these additional 60 names. Those panels are:

- BLG0B903
- BLG0L903
- BLG1A197

You must create 4 new panels to allow these additional 60 names:

1. Copy panel **BLG0B903** twice.
   a. Name the first copy **BLG0B904**.
   b. Name the second copy **BLG0B905**.
2. Copy panel **BLG0L903** twice.
   a. Name the first copy **BLG0L904**.
   b. Name the second copy **BLG0L905**.

Change the fields on the panels as indicated in Table 44.

Table 44. Changes Required to Null Reply Control Information

	Null Target Field		**Function Code Index Field**	
On Panel:	**Change From:**	**Change to:**	**Change From:**	**Change to:**
BLG0B903	BLG1A911	BLG0B904	(no change)	(No change)
BLG0B904	BLG1A911	BLG0B905	(no change)	(No change)
BLG0B905	BLG1A911	(no change)	(no change)	(no change)
BLG0L903	(blank)	BLG0L904	000D	000A
BLG0L904	(blank)	BLG0L905	000D	000A
BLG0L905	(blank)	(no change)	000D	(no change)

For each of the 4 new panels you created, select **Summary** on panel BLM8CU70. This allows you to update each field and change the s-words. Every field on panels BLG0B904 and BLG0B905 must have a unique s-word. Use these same s-words on panels BLG0L904 and BLG0L905. Each panel has 60 fields, one for the **ID/Class** field and one for the **Assignee name**.

You must also update control panel BLG1A197 and insert 120 new ADD control lines. For each new control line, update the **Structured word index** field with one of the new s-words you defined in the dictionary. Also, update the **Prefix index** field as follows:

- Assignee name control lines: **056B**
- ID/Class control lines: **056A**

When you finish creating these control lines, file the panel. You must delete the current USERS record so that the updated control panel (BLG1A197) will initialize the new fields.

Test your changes. If your changes do not work, verify that the s-words and prefix indexes are correct in the BLG1A197 control panel. Also check to be sure that the **Null target** fields on the BLG0B903, BLG0B904, and BLG0B905 appear as shown in Table 44.

Changing the Notification Message

The TSPs and the notification text in each follows:

Table 45. Notification Message Text

TSP Name	Message Text
BLGNOTAE	Activity record–*rnid* created
BLGNOTAU	Activity record–*rnid* updated
BLGTSPAE	Activity record–*rnid* created
BLGTSPAU	Activity record–*rnid* updated
BLGNOTCE	Change record–*rnid* created
BLGNOTCU	Change record–*rnid* updated
BLGTSPCE	Change record–*rnid* created
BLGTSPCU	Change record–*rnid* updated
BLGNOTPE	Problem record–*rnid* created
BLGNOTPU	Problem record–*rnid* updated
BLGTSPPE	Problem record–*rnid* created
BLGTSPPU	Problem record–*rnid* updated

Three control panels support immediate notification:

- BLG1A111 — Problem record
- BLG1A121 — Change record
- BLG1A131 — Activity record.

These control panels call the required TSP when a record is filed. You need not change the control panels if you change either the text of the notification message or the message recipient.

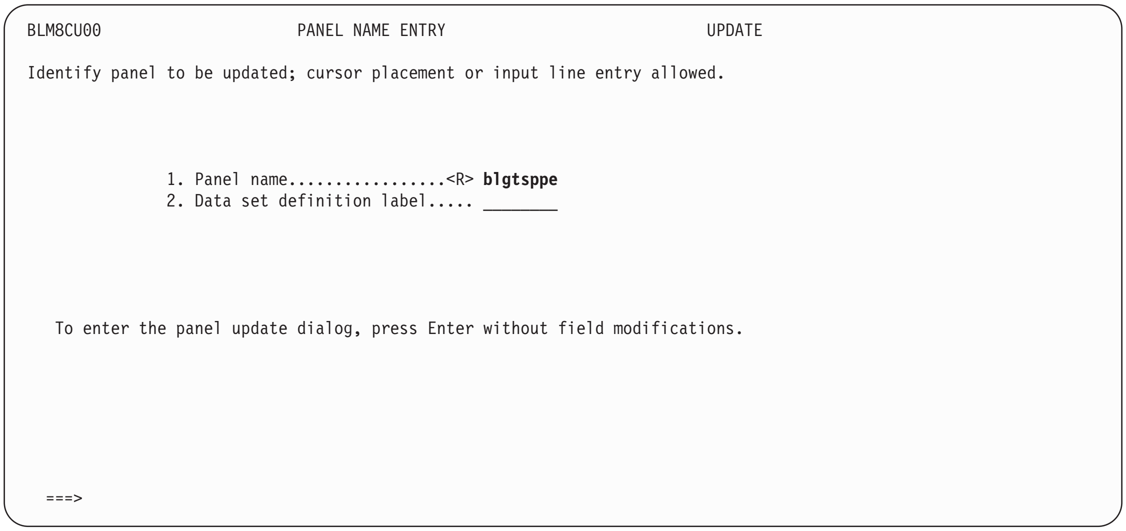
You build the message text with a series of MOVEVAR control lines. You can change the message text in the TSPs by modifying these control lines.

The TSPs that generate the messages follow the naming convention BLGTSP*xy* and BLGNOT*xy*, where *x* is **P** for problem records, **C** for change records, and **A** for activity records. The *y* character is **E** for newly entered records or **U** for updated records.

This example uses TSP BLGTSPPE, for newly entered problem records, to demonstrate how to modify the notification message to include the **Description** field. This example works only when the assignee name is not specified, the assignee name is not in the USERS record (the default ID is used), or the assignee name is an individual. If you wish to include the **Description** field in the notification message when the assignee name is a privilege class, TSP BLGNOTPE must be modified as TSP BLGTSPPE is modified in this example.

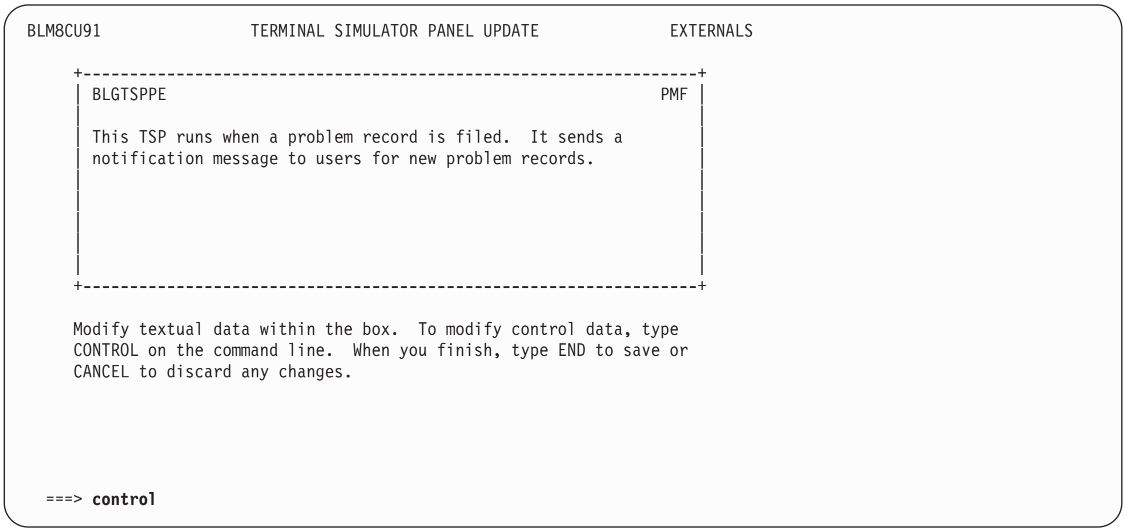
Select **PMF** from the Management application Primary Options Menu (BLG0EN20). Then select **Panel update** from the Panel Modification Facility options panel (BLM8C000). The Panel Name Entry panel shown here appears.

In the **Panel name** field, type **blgtsppe** and press **Enter** twice.



```
BLM8CU00                    PANEL NAME ENTRY                           UPDATE

Identify panel to be updated; cursor placement or input line entry allowed.




               1. Panel name.................<R> blgtsppe
               2. Data set definition label..... ________





   To enter the panel update dialog, press Enter without field modifications.








  ===>
```

On the control line of panel BLM8CU91, type `control` and press **Enter**.



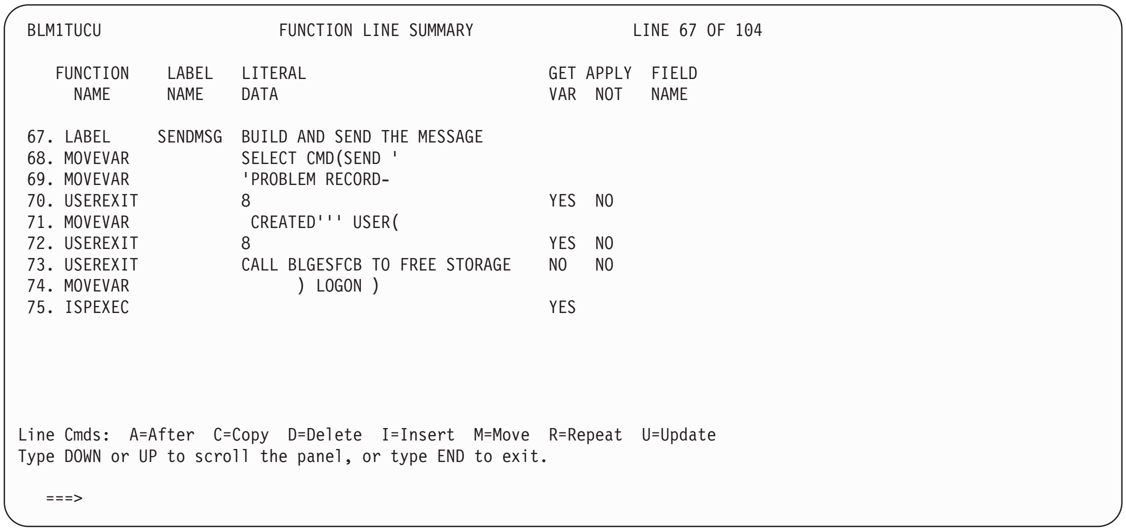
```
BLM8CU91               TERMINAL SIMULATOR PANEL UPDATE             EXTERNALS

     +-------------------------------------------------------------------+
     | BLGTSPPE                                                      PMF |
     |                                                                   |
     | This TSP runs when a problem record is filed.  It sends a         |
     | notification message to users for new problem records.            |
     |                                                                   |
     |                                                                   |
     |                                                                   |
     |                                                                   |
     |                                                                   |
     +-------------------------------------------------------------------+

     Modify textual data within the box.  To modify control data, type
     CONTROL on the command line.  When you finish, type END to save or
     CANCEL to discard any changes.





  ===> control
```

The control lines for the TSPs appear. This TSP is large and does a lot of processing, but the actual notification message is generated with only 17 control lines. These control lines start at line 68. If you have deleted the BRANCH statement to enable immediate notification, these control lines start at line 67.

Note: The starting line might be different if you have made other modifications to this TSP or if you are using an earlier version of Tivoli Information Management for z/OS.

The 17 lines begin with a LABEL command. In this case, the LABEL command is a comment line. The comment is BUILD AND SEND THE MESSAGE. The 8 control lines that follow this LABEL line are those that build the notification message. Two of those 8 lines must not be changed. They are the USEREXIT commands with an 8 in the **Literal data** field. ***Do not*** modify these lines; these are the lines that move the record ID and user ID into the message. In this example, line 70 and line 72 must not be changed.

These are control lines that create notification messages. (The next three BLM1TUCU panel illustrations do not show all the data that appears on the panel. They also assume you have deleted the BRANCH statement to enable immediate notification.)



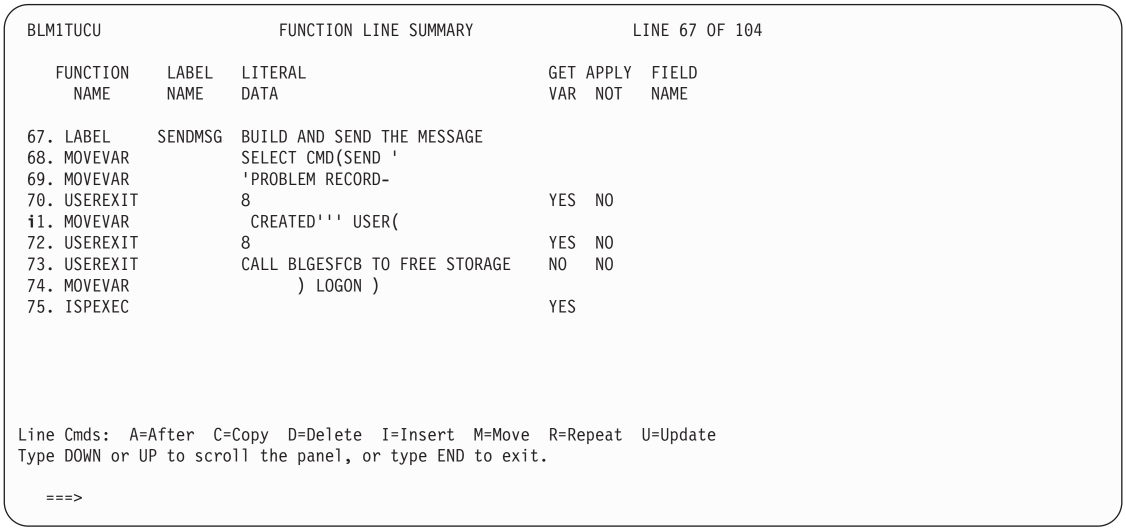
```
 BLM1TUCU                  FUNCTION LINE SUMMARY                LINE 67 OF 104

    FUNCTION    LABEL   LITERAL                          GET APPLY  FIELD
      NAME      NAME    DATA                             VAR  NOT   NAME

 67. LABEL     SENDMSG  BUILD AND SEND THE MESSAGE
 68. MOVEVAR            SELECT CMD(SEND '
 69. MOVEVAR            'PROBLEM RECORD-
 70. USEREXIT           8                                YES  NO
 71. MOVEVAR             CREATED''' USER(
 72. USEREXIT           8                                YES  NO
 73. USEREXIT           CALL BLGESFCB TO FREE STORAGE    NO   NO
 74. MOVEVAR                  ) LOGON )
 75. ISPEXEC                                             YES




Line Cmds:  A=After  C=Copy  D=Delete  I=Insert  M=Move  R=Repeat  U=Update
Type DOWN or UP to scroll the panel, or type END to exit.

   ===>
```

The notification message is built by moving all the parts of the message into the variable data area and then processing the data in the variable data area as a TSO SEND command with the ISPEXEC command line.

The first MOVEVAR command (line 68) that was created has the **Replace Data** field set to YES to clear the variable data area and to insert the new data. Any additional MOVEVAR commands you add must have the **Replace Data** field set to **NO**. The data that is associated with the additional MOVEVAR command lines is concatenated to the data in the first MOVEVAR command line.

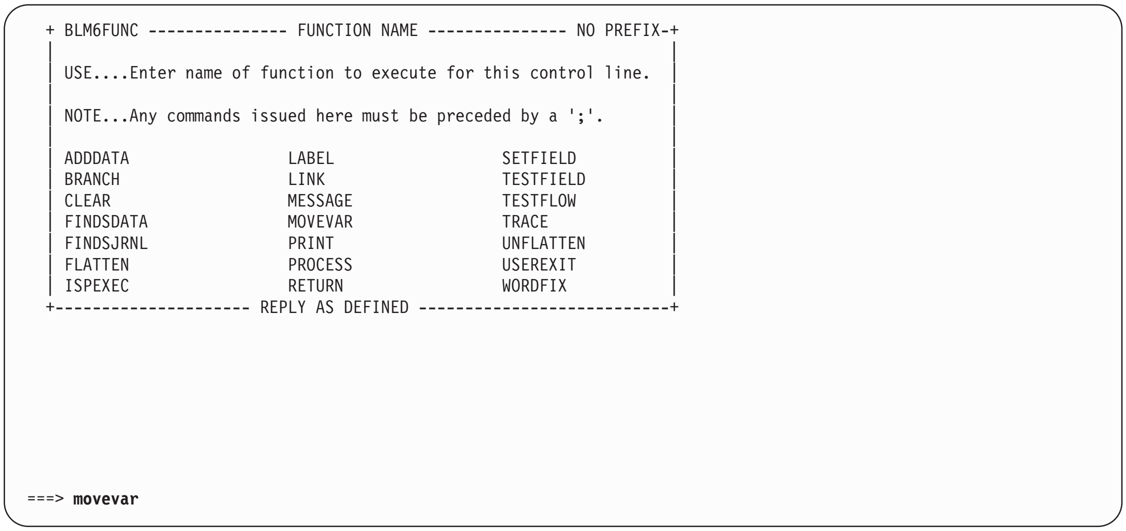
The actual text that appears on the screen as the notification message is the data between the two sets of triple quotation marks Any data you add must be between these sets of quotation marks. Any data you add will be in the case in which you enter it. To prevent interfering with the USEREXITs and other parts of the message, the description data is concatenated at the end of the message. That is, the description field follows the word CREATED in line 71.

Line 71 must be broken into two separate control lines, and the additional commands you add must be inserted between them. Split the original **MOVEVAR** control line at the closing set of triple quotation marks. In the line command area of line 71 use the **I** (INSERT) command.



```
 BLM1TUCU                  FUNCTION LINE SUMMARY                LINE 67 OF 104

    FUNCTION    LABEL   LITERAL                          GET APPLY  FIELD
      NAME      NAME    DATA                             VAR  NOT   NAME

 67. LABEL     SENDMSG  BUILD AND SEND THE MESSAGE
 68. MOVEVAR            SELECT CMD(SEND '
 69. MOVEVAR            'PROBLEM RECORD-
 70. USEREXIT           8                                YES  NO
 i1. MOVEVAR             CREATED''' USER(
 72. USEREXIT           8                                YES  NO
 73. USEREXIT           CALL BLGESFCB TO FREE STORAGE    NO   NO
 74. MOVEVAR                  ) LOGON )
 75. ISPEXEC                                             YES




Line Cmds:  A=After  C=Copy  D=Delete  I=Insert  M=Move  R=Repeat  U=Update
Type DOWN or UP to scroll the panel, or type END to exit.

  ===>
```

Create a MOVEVAR control line.

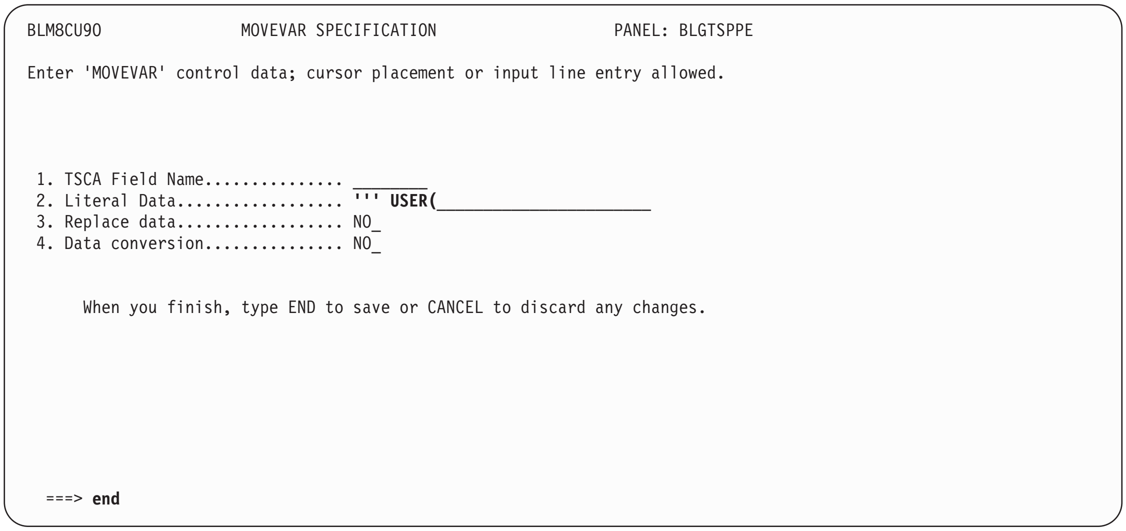


```
  + BLM6FUNC --------------- FUNCTION NAME --------------- NO PREFIX-+
  |                                                                  |
  | USE....Enter name of function to execute for this control line.  |
  |                                                                  |
  | NOTE...Any commands issued here must be preceded by a ';'.       |
  |                                                                  |
  | ADDDATA                LABEL                  SETFIELD           |
  | BRANCH                 LINK                   TESTFIELD          |
  | CLEAR                  MESSAGE                TESTFLOW           |
  | FINDSDATA              MOVEVAR                TRACE              |
  | FINDSJRNL              PRINT                  UNFLATTEN          |
  | FLATTEN                PROCESS                USEREXIT           |
  | ISPEXEC                RETURN                 WORDFIX            |
  +--------------------- REPLY AS DEFINED ---------------------------+







===> movevar
```

After you enter all of the assignee names and ID/Classes for your organization, type **end** and press **Enter** to file the record.

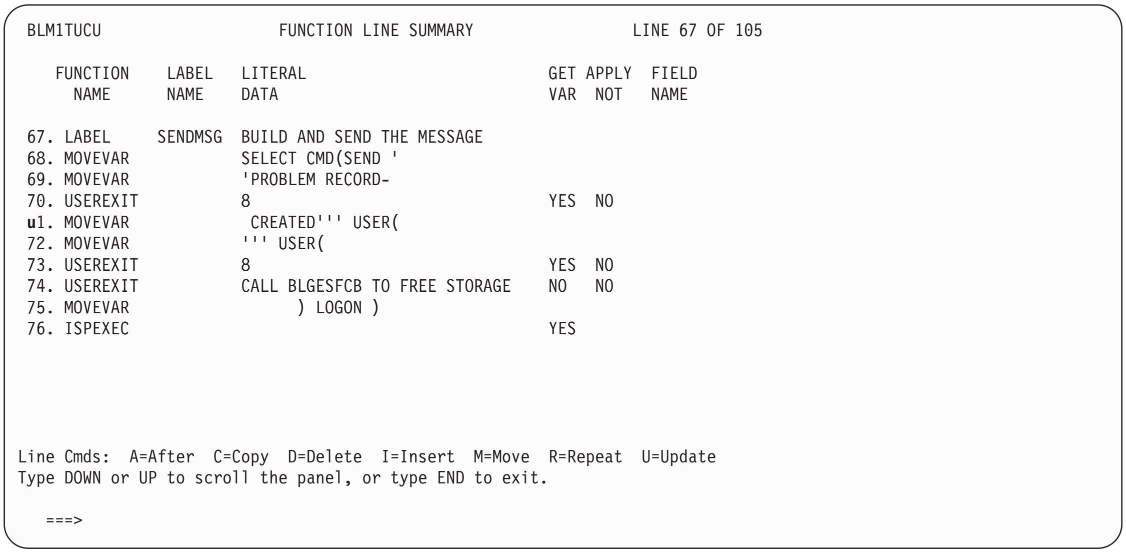
Type **''' USER(** in the **Literal data** field. Then, on the command line, type **end** and press **Enter** to leave the panel.

Note: Be aware that the data will be kept in the case in which you enter the data. The panels which follow reflect the term *USER* rather than *user*.



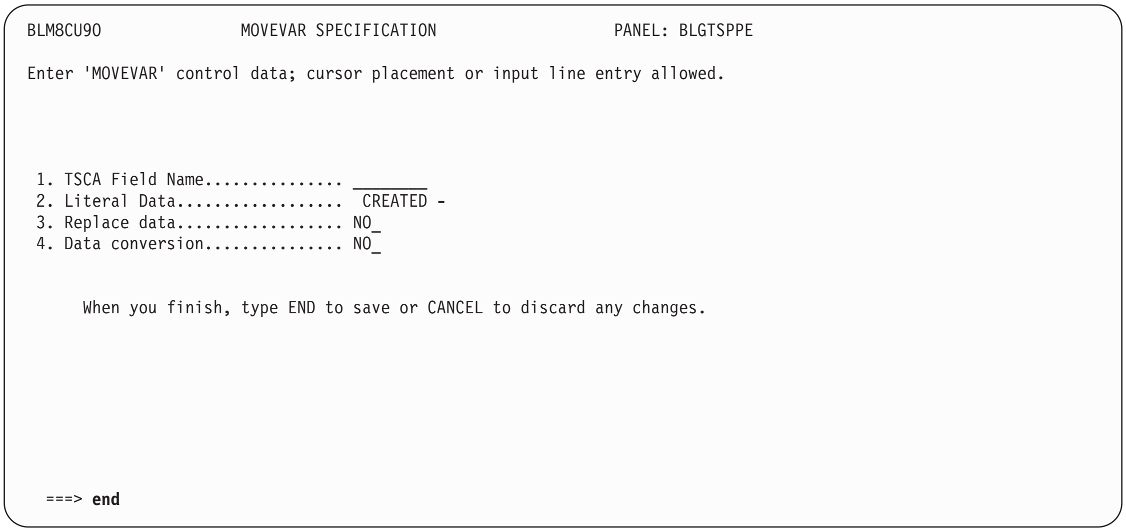
```
BLM8CU90               MOVEVAR SPECIFICATION                 PANEL: BLGTSPPE

Enter 'MOVEVAR' control data; cursor placement or input line entry allowed.




 1. TSCA Field Name.............. ________
 2. Literal Data................. ''' USER(_______________________
 3. Replace data................. NO_
 4. Data conversion.............. NO_


      When you finish, type END to save or CANCEL to discard any changes.








  ===> end
```

Now, remove that information from the original MOVEVAR control line. In the line command area of line 71 use the **U** (UPDATE) command.



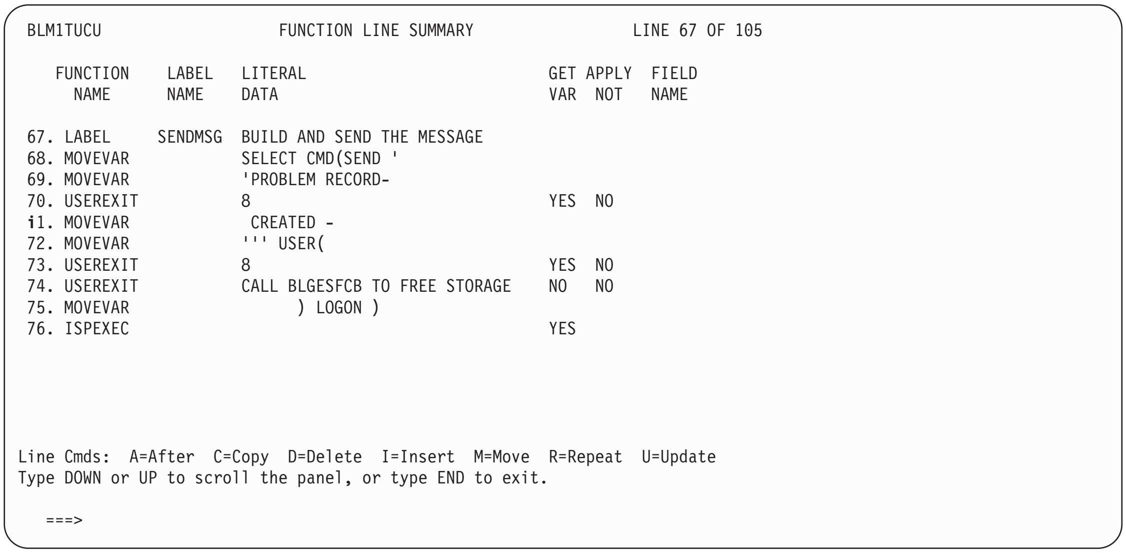
```
 BLM1TUCU                  FUNCTION LINE SUMMARY                LINE 67 OF 105

    FUNCTION    LABEL   LITERAL                         GET APPLY  FIELD
      NAME      NAME    DATA                            VAR  NOT   NAME

 67. LABEL     SENDMSG  BUILD AND SEND THE MESSAGE
 68. MOVEVAR            SELECT CMD(SEND '
 69. MOVEVAR            'PROBLEM RECORD-
 70. USEREXIT           8                               YES  NO
 u1. MOVEVAR             CREATED''' USER(
 72. MOVEVAR            ''' USER(
 73. USEREXIT           8                               YES  NO
 74. USEREXIT           CALL BLGESFCB TO FREE STORAGE   NO   NO
 75. MOVEVAR                  ) LOGON )
 76. ISPEXEC                                            YES




Line Cmds:  A=After  C=Copy  D=Delete  I=Insert  M=Move  R=Repeat  U=Update
Type DOWN or UP to scroll the panel, or type END to exit.

  ===>
```

In the **Literal data** field, blank out the data after the word CREATED, and add a dash as shown. On the command line, type **end** and press **Enter** to leave the panel.



```
 BLM8CU9O              MOVEVAR SPECIFICATION                  PANEL: BLGTSPPE

 Enter 'MOVEVAR' control data; cursor placement or input line entry allowed.




  1. TSCA Field Name............... ________
  2. Literal Data.................  CREATED -
  3. Replace data.................. NO_
  4. Data conversion............... NO_


       When you finish, type END to save or CANCEL to discard any changes.







   ===> end
```

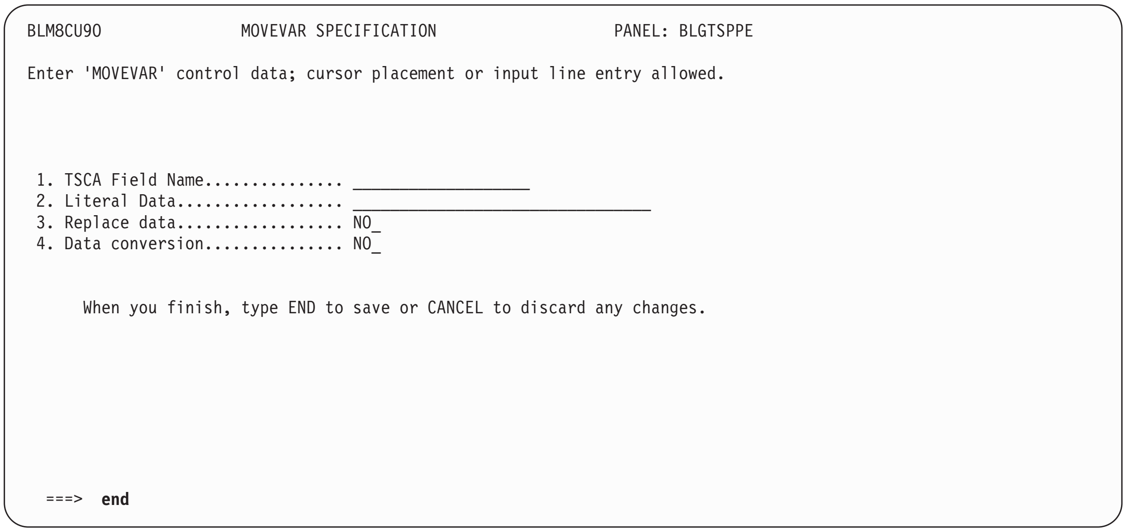
To add the **Description** field data to the notification message, in the line command area of line 71, use the **I** (INSERT) command to add two new control lines.



```
BLM1TUCU                   FUNCTION LINE SUMMARY                LINE 67 OF 105

   FUNCTION    LABEL   LITERAL                          GET APPLY  FIELD
     NAME      NAME    DATA                             VAR  NOT   NAME

67. LABEL     SENDMSG  BUILD AND SEND THE MESSAGE
68. MOVEVAR            SELECT CMD(SEND '
69. MOVEVAR            'PROBLEM RECORD-
70. USEREXIT           8                                YES  NO
i1. MOVEVAR             CREATED -
72. MOVEVAR            ''' USER(
73. USEREXIT           8                                YES  NO
74. USEREXIT           CALL BLGESFCB TO FREE STORAGE    NO   NO
75. MOVEVAR                  ) LOGON )
76. ISPEXEC                                             YES




Line Cmds:  A=After  C=Copy  D=Delete  I=Insert  M=Move  R=Repeat  U=Update
Type DOWN or UP to scroll the panel, or type END to exit.

   ===>
```

The first control line to create is a MOVEVAR control line. This line places a blank in the variable data area. Be sure the **Replace data** field is set to NO.

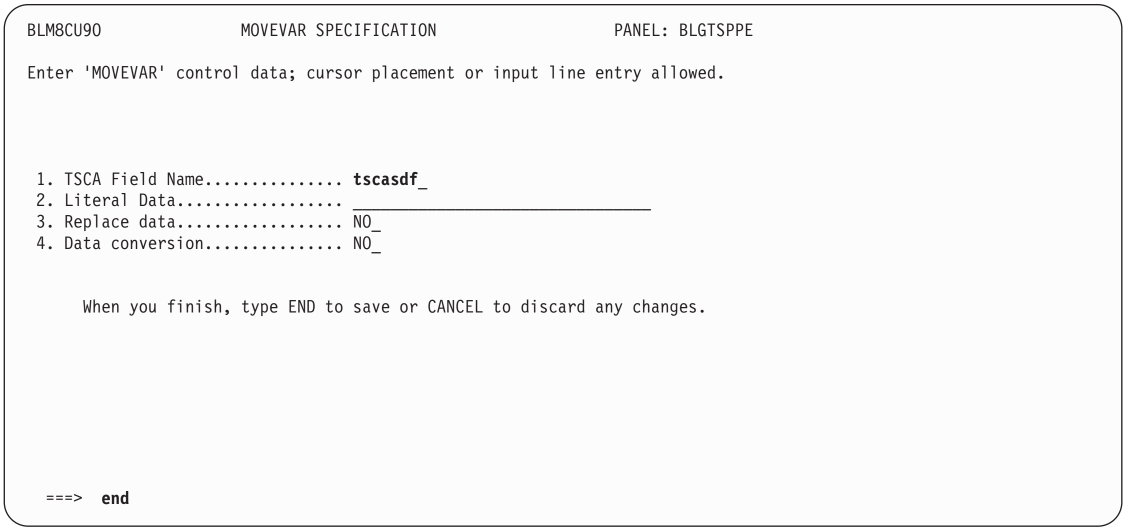
On the command line, type **end** and press **Enter** to leave this panel.



```
BLM8CU90              MOVEVAR SPECIFICATION                 PANEL: BLGTSPPE

Enter 'MOVEVAR' control data; cursor placement or input line entry allowed.




 1. TSCA Field Name............... ___________________
 2. Literal Data.................. ________________________________
 3. Replace data.................. NO_
 4. Data conversion............... NO_


      When you finish, type END to save or CANCEL to discard any changes.







   ===>  end
```

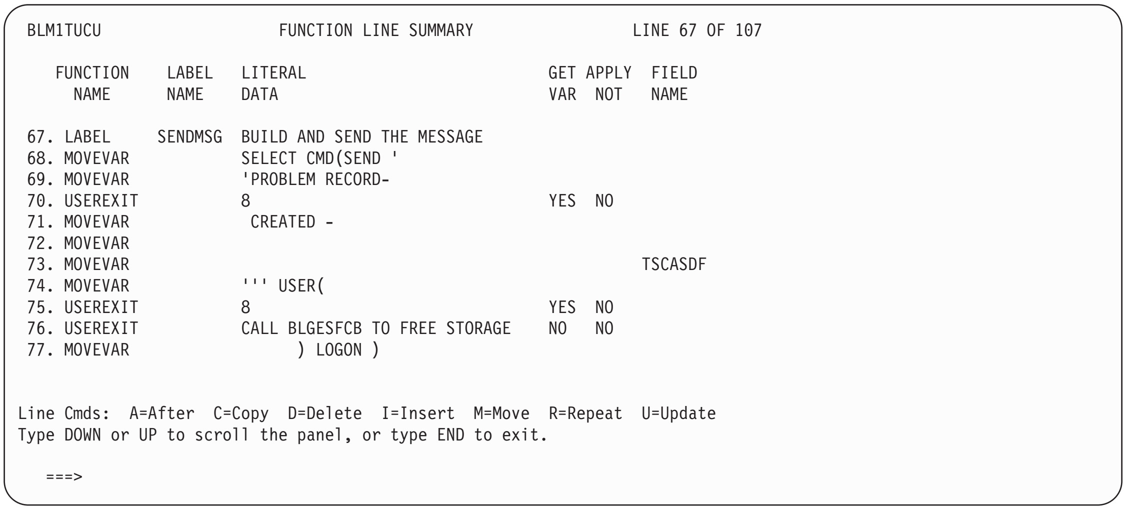
The second control line to create is a MOVEVAR control line. This line takes the **Description** field from the TSCA field TSCASDF and concatenates it to the end of the data in the variable data area. Type **tscasdf** in the **TSCA field name** field. Be sure the **Replace data** field is set to **NO**.

On the command line, type **end** and press **Enter** to leave this panel.



```
BLM8CU9O              MOVEVAR SPECIFICATION                  PANEL: BLGTSPPE

Enter 'MOVEVAR' control data; cursor placement or input line entry allowed.




 1. TSCA Field Name............... tscasdf_
 2. Literal Data.................. ________________________________
 3. Replace data.................. NO_
 4. Data conversion............... NO_


      When you finish, type END to save or CANCEL to discard any changes.








  ===>  end
```

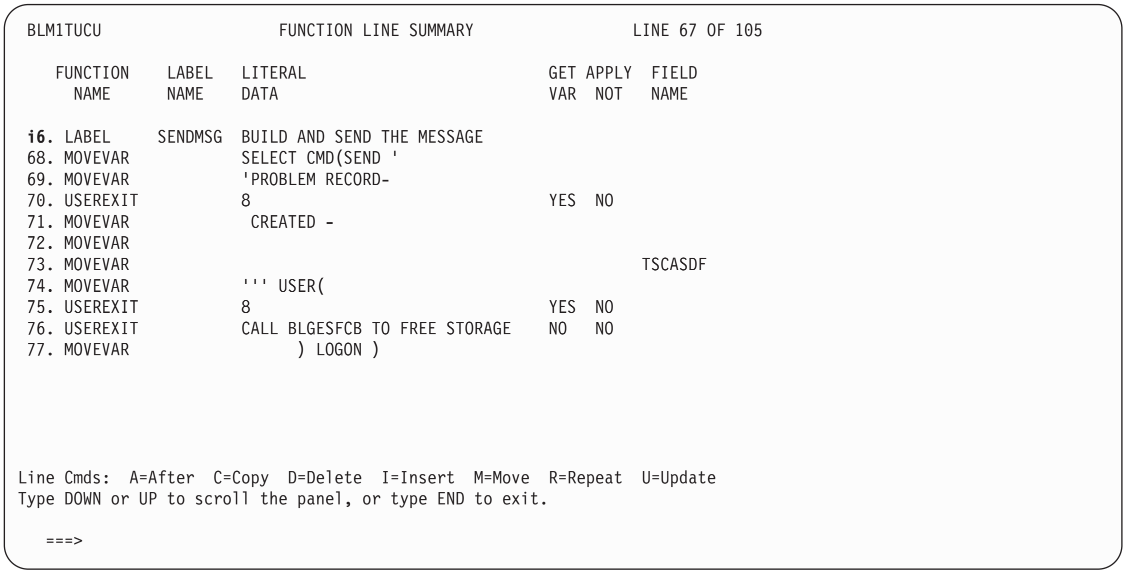
The revised TSP looks like this.



```
BLM1TUCU                  FUNCTION LINE SUMMARY                 LINE 67 OF 107

   FUNCTION    LABEL   LITERAL                         GET APPLY  FIELD
     NAME      NAME    DATA                            VAR  NOT   NAME

67. LABEL     SENDMSG  BUILD AND SEND THE MESSAGE
68. MOVEVAR            SELECT CMD(SEND '
69. MOVEVAR            'PROBLEM RECORD-
70. USEREXIT           8                               YES  NO
71. MOVEVAR             CREATED -
72. MOVEVAR
73. MOVEVAR                                                       TSCASDF
74. MOVEVAR            ''' USER(
75. USEREXIT           8                               YES  NO
76. USEREXIT           CALL BLGESFCB TO FREE STORAGE   NO   NO
77. MOVEVAR                  ) LOGON )


Line Cmds:  A=After  C=Copy  D=Delete  I=Insert  M=Move  R=Repeat  U=Update
Type DOWN or UP to scroll the panel, or type END to exit.

  ===>
```

Now, insert several other control lines to load the **TSCA** field **TSCASDF** with the **Description** from the problem record.

In the line command area of line 67, use the **I** (INSERT) command to add six new control lines.

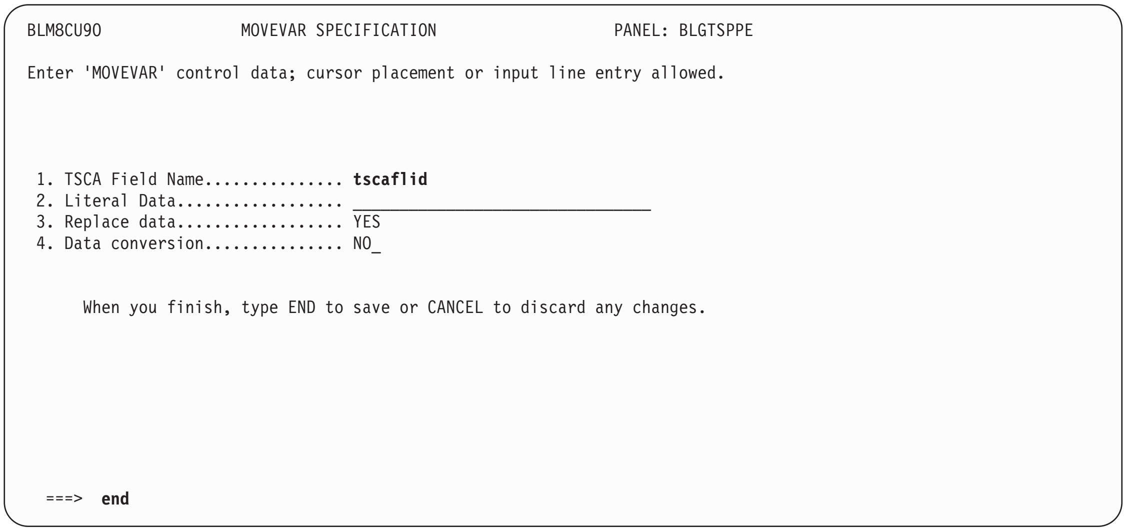


```
 BLM1TUCU                 FUNCTION LINE SUMMARY                LINE 67 OF 105

    FUNCTION    LABEL   LITERAL                         GET APPLY  FIELD
      NAME      NAME    DATA                            VAR  NOT   NAME

 i6. LABEL     SENDMSG  BUILD AND SEND THE MESSAGE
 68. MOVEVAR            SELECT CMD(SEND '
 69. MOVEVAR            'PROBLEM RECORD-
 70. USEREXIT           8                               YES  NO
 71. MOVEVAR             CREATED -
 72. MOVEVAR
 73. MOVEVAR                                                      TSCASDF
 74. MOVEVAR            ''' USER(
 75. USEREXIT           8                               YES  NO
 76. USEREXIT           CALL BLGESFCB TO FREE STORAGE   NO   NO
 77. MOVEVAR                  ) LOGON )





Line Cmds:  A=After  C=Copy  D=Delete  I=Insert  M=Move  R=Repeat  U=Update
Type DOWN or UP to scroll the panel, or type END to exit.

   ===>
```

The first control line to create is a MOVEVAR control line.

Type **tscalfid** in the **TSCA field name** field. Be sure the **Replace data** field is set to **YES**.

On the command line, type **end** and press **Enter** to leave this panel.

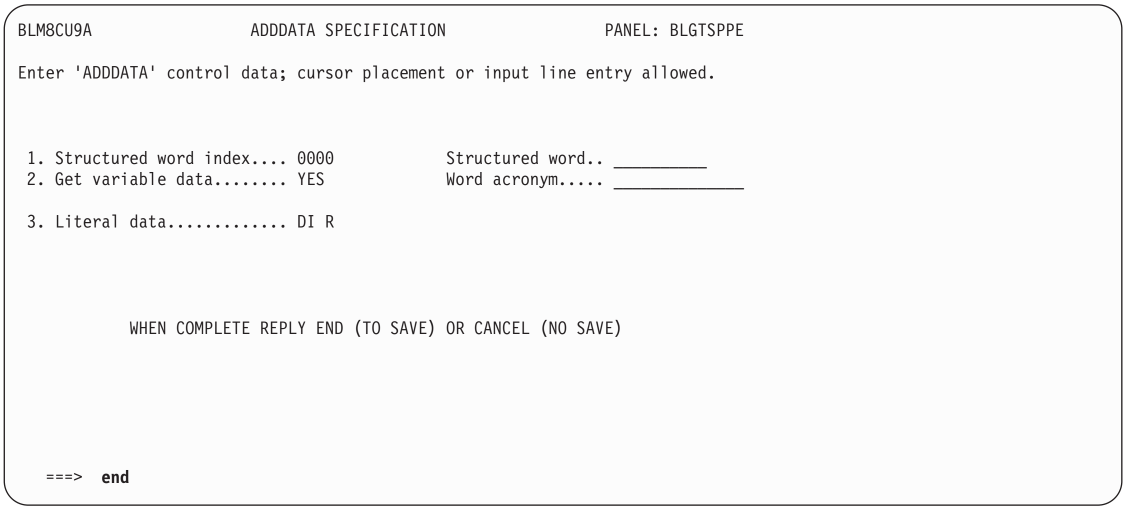


```
 BLM8CU90              MOVEVAR SPECIFICATION                PANEL: BLGTSPPE

 Enter 'MOVEVAR' control data; cursor placement or input line entry allowed.




  1. TSCA Field Name............... tscaflid
  2. Literal Data.................. ________________________________
  3. Replace data.................. YES
  4. Data conversion............... NO_


       When you finish, type END to save or CANCEL to discard any changes.








   ===>  end
```

The second control line to create is an ADDDATA control line.

Type **DI R** in the **Literal data** field. Be sure the **Get variable data** field is set to **YES**.

On the command line, type **end** and press **Enter** to leave this panel.

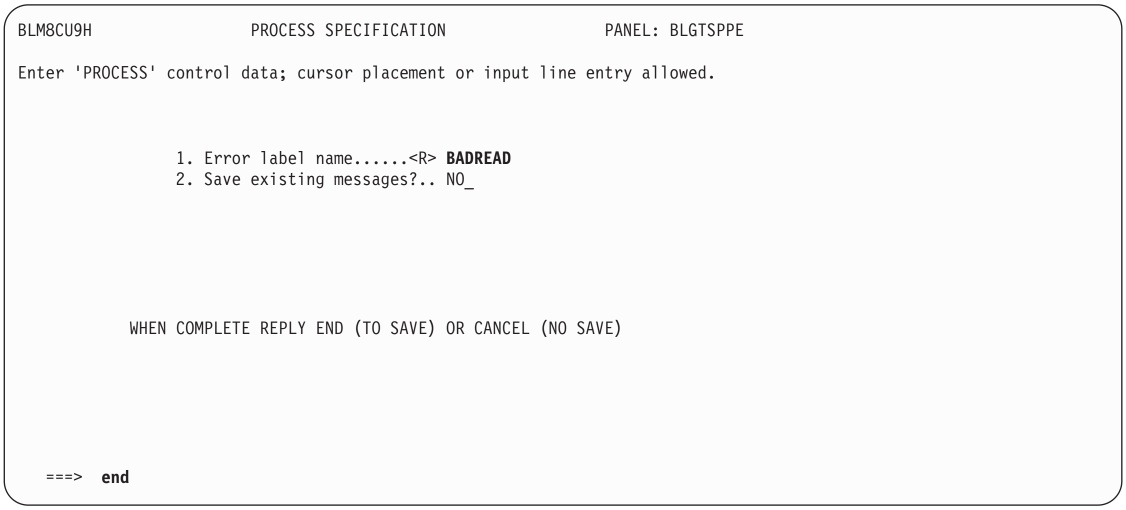


```
BLM8CU9A                ADDDATA SPECIFICATION                PANEL: BLGTSPPE

Enter 'ADDDATA' control data; cursor placement or input line entry allowed.



 1. Structured word index.... 0000           Structured word.. __________
 2. Get variable data........ YES            Word acronym..... ______________

 3. Literal data............. DI R




            WHEN COMPLETE REPLY END (TO SAVE) OR CANCEL (NO SAVE)






   ===>  end
```

The third control line to create is a PROCESS control line. This line will cause the last record filed to be displayed.

Type **BADREAD** in the **Error label name** field.

On the command line, type **end** and press **Enter** to leave this panel.

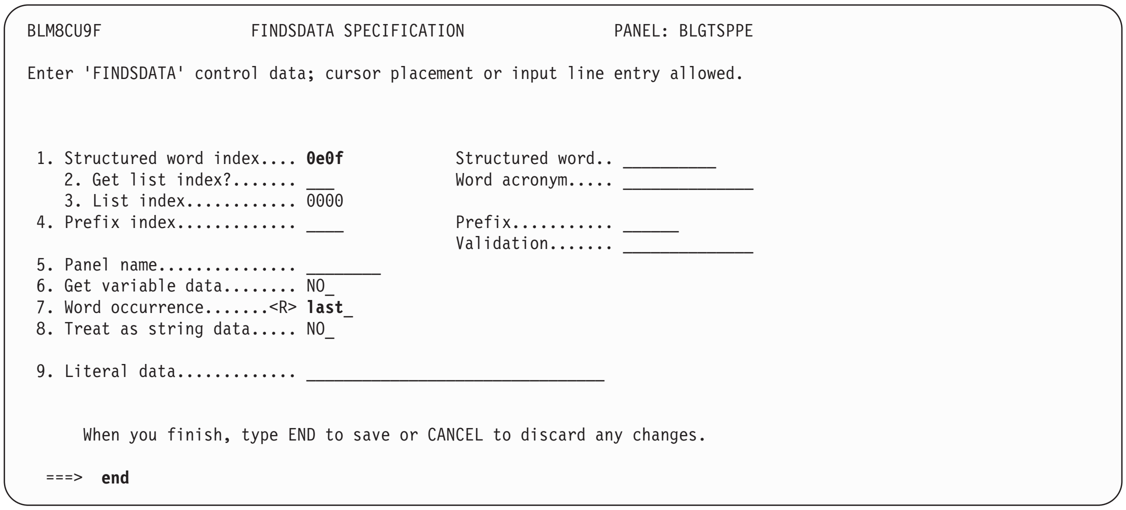


```
BLM8CU9H                PROCESS SPECIFICATION                PANEL: BLGTSPPE

Enter 'PROCESS' control data; cursor placement or input line entry allowed.



                1. Error label name......<R> BADREAD
                2. Save existing messages?.. NO_






            WHEN COMPLETE REPLY END (TO SAVE) OR CANCEL (NO SAVE)






   ===>  end
```

The fourth control line to create is a FINDSDATA control line. The s-word index associated with the **Description** field is S0E0F.

Type **0e0f** in the **Structured word index** field and type **last** in the **Word occurrence** field.

On the command line, type **end** and press **Enter** to leave this panel.



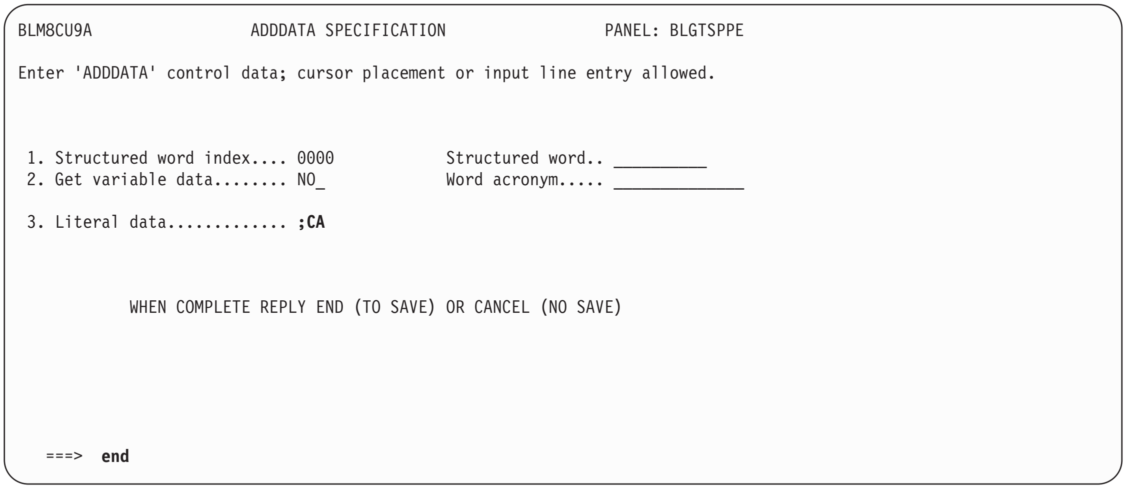
```
BLM8CU9F               FINDSDATA SPECIFICATION               PANEL: BLGTSPPE

Enter 'FINDSDATA' control data; cursor placement or input line entry allowed.



 1. Structured word index.... 0e0f           Structured word.. _________
    2. Get list index?....... ___            Word acronym..... _____________
    3. List index............ 0000
 4. Prefix index............. ____           Prefix........... ______
                                             Validation....... _____________
 5. Panel name............... ________
 6. Get variable data........ NO_
 7. Word occurrence.......<R> last_
 8. Treat as string data..... NO_

 9. Literal data............. ________________________________


      When you finish, type END to save or CANCEL to discard any changes.

  ===>  end
```

Note: It is important to keep the **Description** field a required field in problem, change, and activity entry and update panels, so that the newly added FINDSDATA command always finds data to add to the notification message.

The fifth control line to create is an ADDDATA control line.

Type **;ca** in the **Literal data** field. Be sure the **Get variable data** field is set to **NO**.

On the command line, type **end** and press **Enter** to leave this panel.

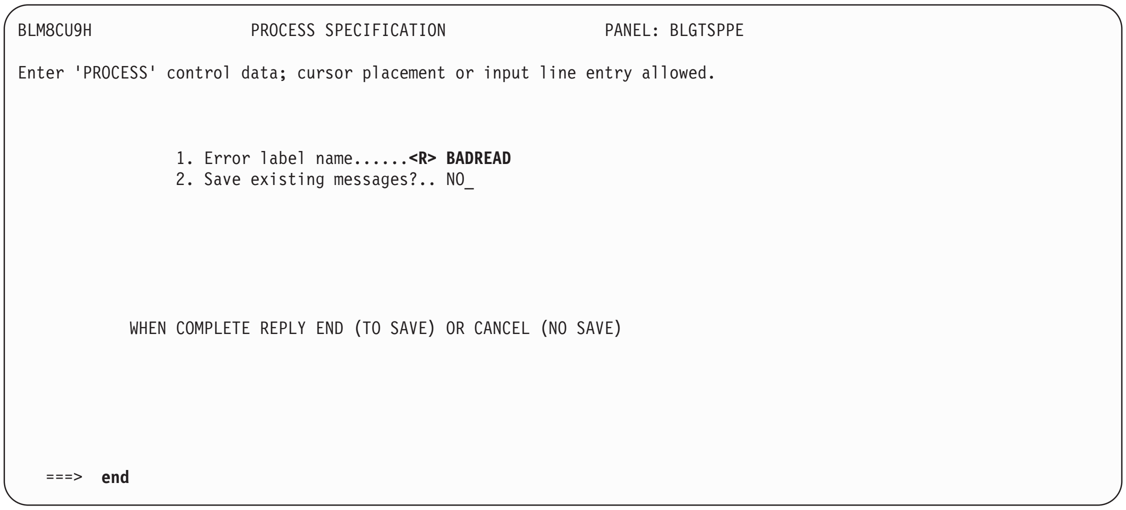


```
BLM8CU9A                ADDDATA SPECIFICATION                PANEL: BLGTSPPE

Enter 'ADDDATA' control data; cursor placement or input line entry allowed.



 1. Structured word index.... 0000           Structured word.. _________
 2. Get variable data........ NO_            Word acronym..... _____________

 3. Literal data............. ;CA



           WHEN COMPLETE REPLY END (TO SAVE) OR CANCEL (NO SAVE)





  ===>  end
```

The sixth control line to create is a PROCESS control line.

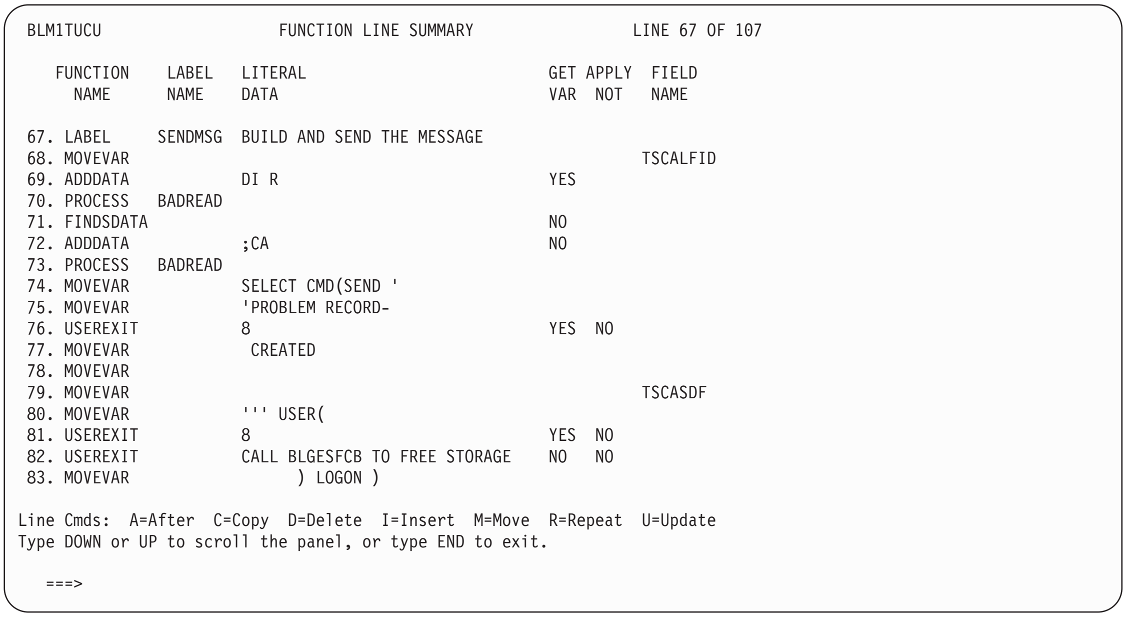
Type **badread** in the **Error label name** field.

On the command line, type **end** and press **Enter** to leave this panel.



```
BLM8CU9H                 PROCESS SPECIFICATION                PANEL: BLGTSPPE

Enter 'PROCESS' control data; cursor placement or input line entry allowed.



                 1. Error label name......<R> BADREAD
                 2. Save existing messages?.. NO_






            WHEN COMPLETE REPLY END (TO SAVE) OR CANCEL (NO SAVE)







   ===>  end
```

The final revised TSP now looks like this.



```
 BLM1TUCU                   FUNCTION LINE SUMMARY                 LINE 67 OF 107

    FUNCTION    LABEL   LITERAL                          GET APPLY  FIELD
      NAME      NAME    DATA                             VAR  NOT   NAME

 67. LABEL     SENDMSG  BUILD AND SEND THE MESSAGE
 68. MOVEVAR                                                       TSCALFID
 69. ADDDATA            DI R                             YES
 70. PROCESS   BADREAD
 71. FINDSDATA                                           NO
 72. ADDDATA            ;CA                              NO
 73. PROCESS   BADREAD
 74. MOVEVAR            SELECT CMD(SEND '
 75. MOVEVAR            'PROBLEM RECORD-
 76. USEREXIT           8                                YES  NO
 77. MOVEVAR             CREATED
 78. MOVEVAR
 79. MOVEVAR                                                       TSCASDF
 80. MOVEVAR            ''' USER(
 81. USEREXIT           8                                YES  NO
 82. USEREXIT           CALL BLGESFCB TO FREE STORAGE    NO   NO
 83. MOVEVAR                  ) LOGON )

Line Cmds:  A=After  C=Copy  D=Delete  I=Insert  M=Move  R=Repeat  U=Update
Type DOWN or UP to scroll the panel, or type END to exit.

   ===>
```

Assuming that your user ID is DON and that your ID is the default, you can use this IRC on the Primary Options Menu to build a problem record after you modify BLGTSPPE:

5,1,1,=,14,open,25,THIS IS A TEST OF THE NOTIFICATION MESSAGE,,9

Because you do not specify an assignee name and because DON is not the name of a privilege class, the message goes to your default user ID, DON. The resulting notification message is:

’PROBLEM RECORD-00000051 CREATED - THIS IS A TEST OF THE NOTIFICATION MESSAGE’ DON

See “Notification Management User Exits” on page 499 for complete descriptions of the user exits called by the immediate notification TSPs. Refer to the *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference* for complete information on creating control lines in a TSP.

Disabling Immediate Notification

You can disable immediate notification for all records if you do not want to use it. If you want to use immediate notification for only one type of record, but not for the others, you can do that, too.

Update the control panels by removing the two sets of multiple test groups that end with the 001B function code so that immediate notification messages are no longer sent. These are the control panels that require updating:

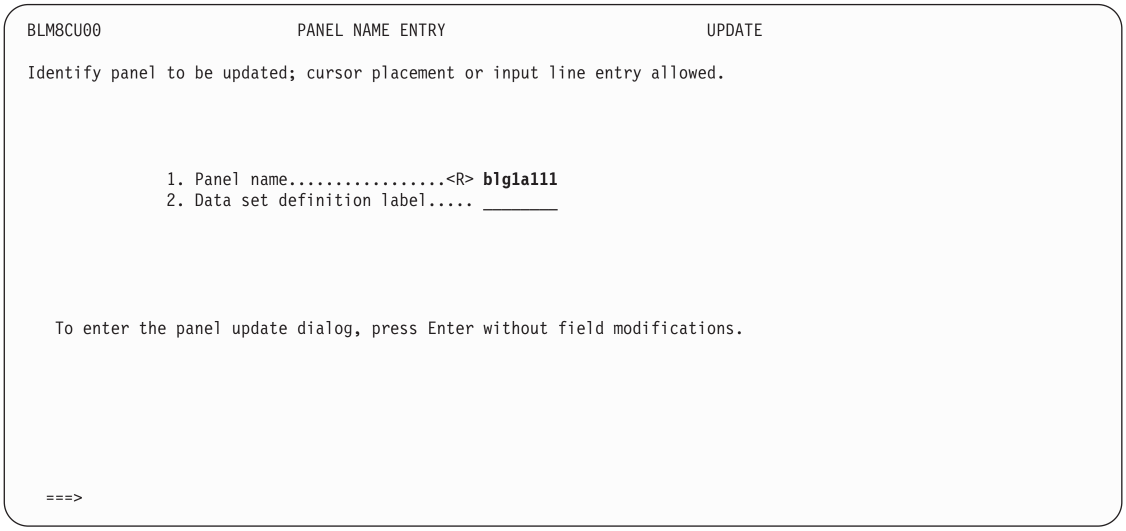
Problem records
BLG1A111

Change records
BLG1A121

Activity records
BLG1A131

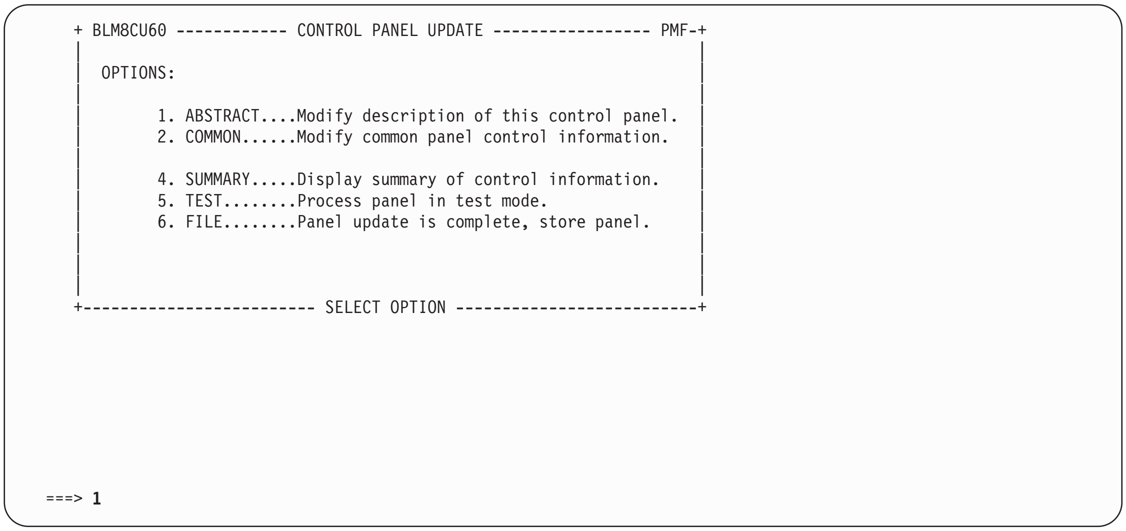
The following example shows how you can disable notification for Problem Management. Disabling Change and Activity notification follows a similar sequence. To begin, select PMF from the Management application Primary Options Menu; then select **Panel update** from the Panel Modification Facility options panel.

To update the control panel for Problem Management, type **blg1a111** in the **Panel name** field and press **Enter** twice.



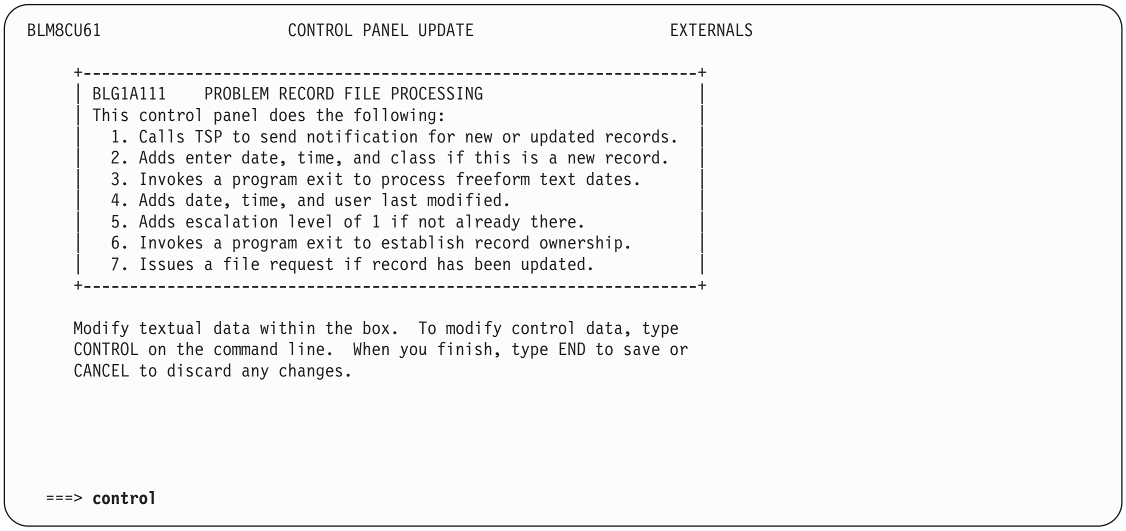
```
BLM8CU00                   PANEL NAME ENTRY                          UPDATE

Identify panel to be updated; cursor placement or input line entry allowed.




               1. Panel name.................<R> blg1a111
               2. Data set definition label..... ________





   To enter the panel update dialog, press Enter without field modifications.







  ===>
```

Select **Abstract** to display details about the control panel. Type **1** and press **Enter**.



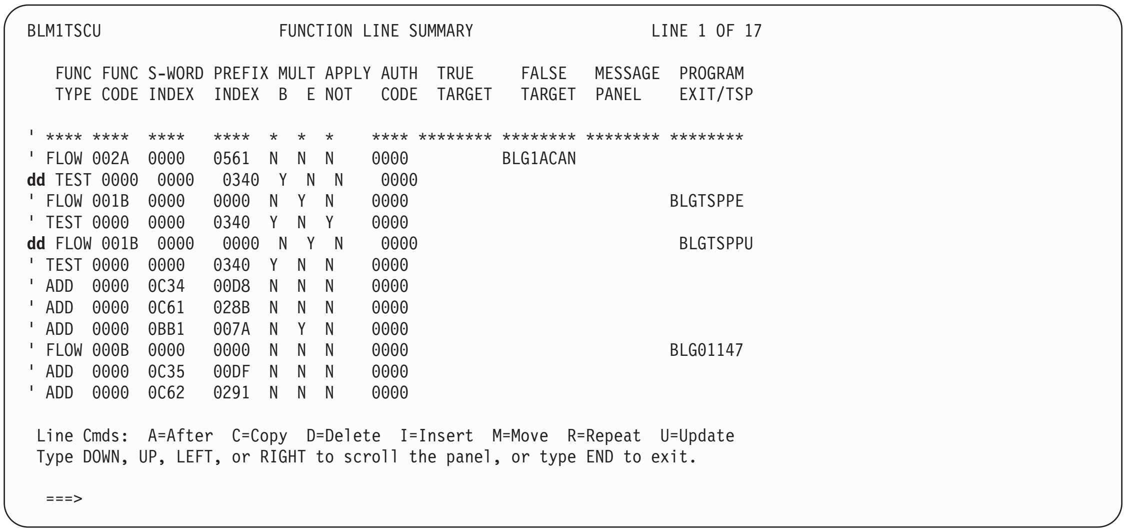
```
    + BLM8CU60 ------------ CONTROL PANEL UPDATE ----------------- PMF-+
    |                                                                  |
    |  OPTIONS:                                                        |
    |                                                                  |
    |        1. ABSTRACT....Modify description of this control panel.  |
    |        2. COMMON......Modify common panel control information.   |
    |                                                                  |
    |        4. SUMMARY.....Display summary of control information.    |
    |        5. TEST........Process panel in test mode.                |
    |        6. FILE........Panel update is complete, store panel.     |
    |                                                                  |
    |                                                                  |
    |                                                                  |
    +------------------------- SELECT OPTION --------------------------+







 ===> 1
```

On the command line of the Control Panel Update abstract description panel, type **control** and press **Enter**.



```
BLM8CU61                   CONTROL PANEL UPDATE                    EXTERNALS

     +-----------------------------------------------------------------+
     | BLG1A111    PROBLEM RECORD FILE PROCESSING                      |
     | This control panel does the following:                          |
     |   1. Calls TSP to send notification for new or updated records. |
     |   2. Adds enter date, time, and class if this is a new record.  |
     |   3. Invokes a program exit to process freeform text dates.     |
     |   4. Adds date, time, and user last modified.                   |
     |   5. Adds escalation level of 1 if not already there.           |
     |   6. Invokes a program exit to establish record ownership.      |
     |   7. Issues a file request if record has been updated.          |
     +-----------------------------------------------------------------+

     Modify textual data within the box.  To modify control data, type
     CONTROL on the command line.  When you finish, type END to save or
     CANCEL to discard any changes.





 ===> control
```

On the Function Line Summary panel, use the **D** (DELETE) line command to delete the two sets of multiple test groups that end with the 001B function code. Each test group includes a TEST and FLOW function type. Type **dd** as shown and press **Enter**.



```
BLM1TSCU                   FUNCTION LINE SUMMARY                   LINE 1 OF 17

   FUNC FUNC S-WORD PREFIX MULT APPLY AUTH  TRUE     FALSE   MESSAGE  PROGRAM
   TYPE CODE INDEX  INDEX  B  E NOT   CODE  TARGET   TARGET  PANEL    EXIT/TSP

' **** ****  ****   ****  *  *  *    **** ******** ******** ******** ********
' FLOW 002A  0000   0561  N  N  N    0000          BLG1ACAN
dd TEST 0000  0000   0340  Y  N  N    0000
' FLOW 001B  0000   0000  N  Y  N    0000                            BLGTSPPE
' TEST 0000  0000   0340  Y  N  Y    0000
dd FLOW 001B  0000   0000  N  Y  N    0000                            BLGTSPPU
' TEST 0000  0000   0340  Y  N  N    0000
' ADD  0000  0C34   00D8  N  N  N    0000
' ADD  0000  0C61   028B  N  N  N    0000
' ADD  0000  0BB1   007A  N  Y  N    0000
' FLOW 000B  0000   0000  N  N  N    0000                            BLG01147
' ADD  0000  0C35   00DF  N  N  N    0000
' ADD  0000  0C62   0291  N  N  N    0000

 Line Cmds:  A=After  C=Copy  D=Delete  I=Insert  M=Move  R=Repeat  U=Update
 Type DOWN, UP, LEFT, or RIGHT to scroll the panel, or type END to exit.

  ===>
```

If you had enabled immediate notification using TSXs and TCP/IP SMTP, the names **BLGTSPPE** and **BLGTSPPU** in the Function Line Summary panel would be changed to **BLGTSXPE** and **BLGTSXPU**.

File your changes, and copy the panel into a read panel data set.

Understanding Problem Escalation

You can use the problem escalation facility of Tivoli Information Management for z/OS to inform interested users of a problem's status and its priority until the problem is closed. Escalation processing runs in batch mode and at the intervals you set. The escalation of problem records is controlled by a list of criteria that you define in RULES records, a record type defined specifically for problem escalation.

The escalation facility uses the RULES records criteria to create a search argument that identifies problem records that you want escalated. For each problem that is identified, Tivoli Information Management for z/OS sends an alert message to one or more user IDs or mail addresses, as specified in the RULES record. The **Current priority** field and the **Escalation level** field of the problem record are updated each time a message is sent.

Escalation messages can be processed using MVS TCP/IP SMTP mail. To take advantage of this function, you must build the escalation message using the existing *ddname* or *dsname* processing as described later in this section.

Implementing Problem Escalation

Before you can implement escalation processing, you need to understand the process and perform the following tasks:

1. Read this section for an overview of the problem escalation process and the implementation tasks.

2. Establish your installation's escalation criteria within the problem management process. If you want to use escalation criteria that are different from those shipped by Tivoli, use PMF to update the RULES record panels.

3. Define the RULES records for each set of criteria that you want to monitor.
4. Create the JCL to run the escalation batch job and set the interval at which this job is to be run. Some sample JCL, called BLMESCAL, is provided for you in the SBLMSAMP data set.

Understanding the Escalation Process

The escalation facility uses RULES records to create a search argument to identify problem records to escalate. For each problem that is escalated, an alert message is sent to one or more user IDs specified in the RULES record. You can also use each RULES record user ID field as an “alias” that is defined along with one or more mail addresses in the USERS record. The alias can be a privilege class name or a name of your choice to identify one or more people that should be sent an escalation message. If you use this method, the escalation message is sent to each address defined for the alias. **Current Priority** and **Escalation Level** fields in the problem record are also updated.

The following steps outline the processing performed by TSP BLGESCAL. See “Notification Management User Exits” on page 499 for information about the escalation control block (ESCB) and the user exits for notification management.

1. Search for all RULES records and obtain a search results list. For complete information about creating RULES records, see “Creating Rules Records” on page 487.
2. Select each RULES record, one at a time from the search results list, and do the following:
   a. Move selected information from the RULES record to the ESCB.
   b. Move the search criteria from the RULES record to the ESCB.
   c. Store the escalation command template in the ESCB. The template contains the command and an outline of the format used to send the escalation alert message.
   d. Build a search argument to find all nonclosed[1] problem records in the database.
   e. Determine which of the escalation criteria exist in the RULES record. Add them to the search argument. For example, if Key item affected is one of the search criteria and the RULES record contains an entry of REPORTS in the **Key item affected** field, then COMK/REPORTS is added to the search argument (COMK/ is the p-word for Key item affected). If the **Key item affected** field in the RULES record is blank, this p-word is not added to the search argument.
   f. Issue a SEARCH command to get a list of all nonclosed problem records that are candidates for escalation according to this RULES record.
   g. Process each problem record:
      1) Subtract the **Date** and **Time** fields in the problem record from the current date and time. The result is the calculated duration. The escalation level in the problem record determines whether the Date entered and **Time entered** fields or the **Date last altered** and **Time last altered** fields are used for the subtraction operation.
      2) Compare the calculated duration against the duration in the RULES record to determine whether to issue an alert. The escalation level in the problem record determines whether the level 1, 2, or 3 duration should be used.

D. Using Notification Management

[1] A nonclosed problem is a problem with any status other than closed.

3) Build the alert message, using either the data in the variable data area for the Terminal Simulator Communications Area (TSCA) or a specified message panel.

4) Substitute the alert message and the appropriate information from the ESCB in the escalation command template.

5) Send the alert message.

6) If the **Priority adjustment** field in the RULES record is not 0, update the **Current priority** field in the problem record.

7) Update the Escalation level in the problem record.

8) File the record.

Note: When problem records are updated because of escalation processing, immediate notification messages are *not* sent.

9) Processing is complete for one problem record. Repeat steps 2.g.1 through 2.g.8 until all problem records in the search results list are processed.

h. Processing is complete for one RULES record. Repeat steps 2.a through 2.g until all RULES records in the search results list created in step 1 have been processed.

3. Processing is complete. Print the messages on the chain and exit from Tivoli Information Management for z/OS.

The Escalation TSPs and TSXs

The BLGESCAL TSP controls the escalation facility. This TSP calls other TSPs and TSXs to perform different escalation functions. The following table names each TSP and its purpose. The processing steps in this table refer to “Understanding the Escalation Process” on page 485.

TSP Name	Purpose of the TSP	Processing Step
BLGESCAL	Escalation facility control.	1, 2.h, 3
BLGESC01	Finds the escalation criteria in a RULES record and stores the criteria in the ESCB.	2.a, 2.b, 2.d, 2.e, 2.f
BLGESC02	Controls the problem record loop area.	2.g.8, 2.g.9
BLGESC03	Gets data from the RULES record and stores it in the ESCB.	2.a
BLGESC04	(Continuation of BLGESC03.) Also stores the notification command template in the ESCB.	2.a, 2.c
BLGESC05	Determines the escalation level of a problem record.	N/A
BLGESC06	Performs processing for a problem record based on the escalation level.	2.g.1 through 2.g.7
BLGESC07	Builds a notification message.	2.g.3

In addition to these TSPs, a TSX is available for escalation processing:

TSX Name	Purpose of the TSX
BLGESCX6	Sends the escalation message via TCP/IP SMTP

If you elect to use the BLGESCX6 TSX, you must modify some escalation TSPs. See "Using the BLGESCX6 TSX with TCP/IP SMTP" on page 497 for more information.

Creating Rules Records

Familiarize yourself with the escalation criteria defined in the escalation facility and decide whether they fit your installation's needs. You must create your RULES records before escalation processing can be done. The security of your RULES records is controlled by the privilege class records in your database. The Privilege Class Record panels include authority for RULES records.

The criteria used in the escalation facility as shipped from Tivoli uses the following fields in the RULES records to find matching fields in problem records:

- **Initial priority**
- **Problem type**
- **Location code**
- **Key item affected**.

Create a RULES record for each set of unique search criteria that you want escalated. Refer to the *Tivoli Information Management for z/OS User's Guide* for information about creating records.

To establish your site's escalation criteria, make a list of the possible combinations of escalation criteria that are valid for your site. For example, perhaps your location has the following:

- Four initial priority values: 01, 02, 03, and 04
- Two problem types: Hardware and Software
- One location: A10
- Two key items affected: TAPE01 and DISK03B

Because every problem reported has a location code of A10, you need not use the **Location code** field as part of your search argument. Therefore, this field is not an escalation criterion for your installation.

Continuing with this example, you come up with a list of 16 RULES records that include every possible situation that can occur. Table 46 lists these RULES records.

Table 46. Example Rules Records

Rules Record Number	Initial Priority	Key Item Affected	Problem Type
1	01	TAPE01	Hardware
2	01	TAPE01	Software
3	01	DISK03B	Hardware
4	01	DISK03B	Software
5	02	TAPE01	Hardware
6	02	TAPE01	Software
7	02	DISK03B	Hardware

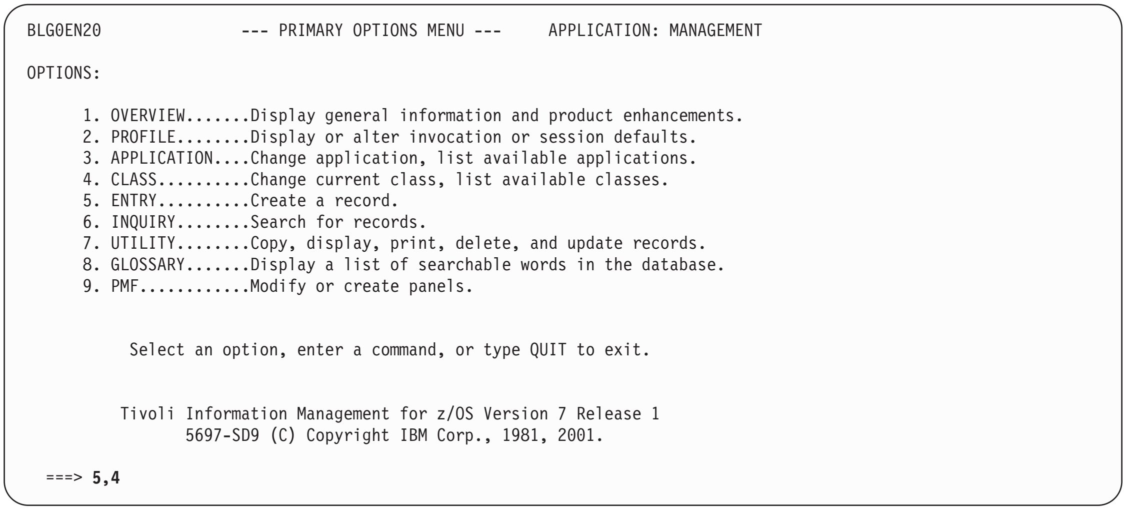
Table 46. Example Rules Records (continued)

Rules Record Number	Initial Priority	Key Item Affected	Problem Type
8	02	DISK03B	Software
9	03	TAPE01	Hardware
10	03	TAPE01	Software
11	03	DISK03B	Hardware
12	03	DISK03B	Software
13	04	TAPE01	Hardware
14	04	TAPE01	Software
15	04	DISK03B	Hardware
16	04	DISK03B	Software

Scanning this list, you find that some combinations cannot logically occur in your installation. For example, you never open software problems with an initial priority of 01. Therefore, you do not need to create RULES records numbered 2 and 4.

Suppose you have assigned only one person to monitor all problems with the key item TAPE01. You, therefore, need to create only one RULES record for the key item affected TAPE01, leaving the **Initial priority** and **Problem type** fields blank. In this example, you do not need to create RULES records numbered 1, 2, 5, 6, 9, 10, 13, and 14. The one RULES record that you need to create uses the **Key item affected** field as the only escalation criterion.

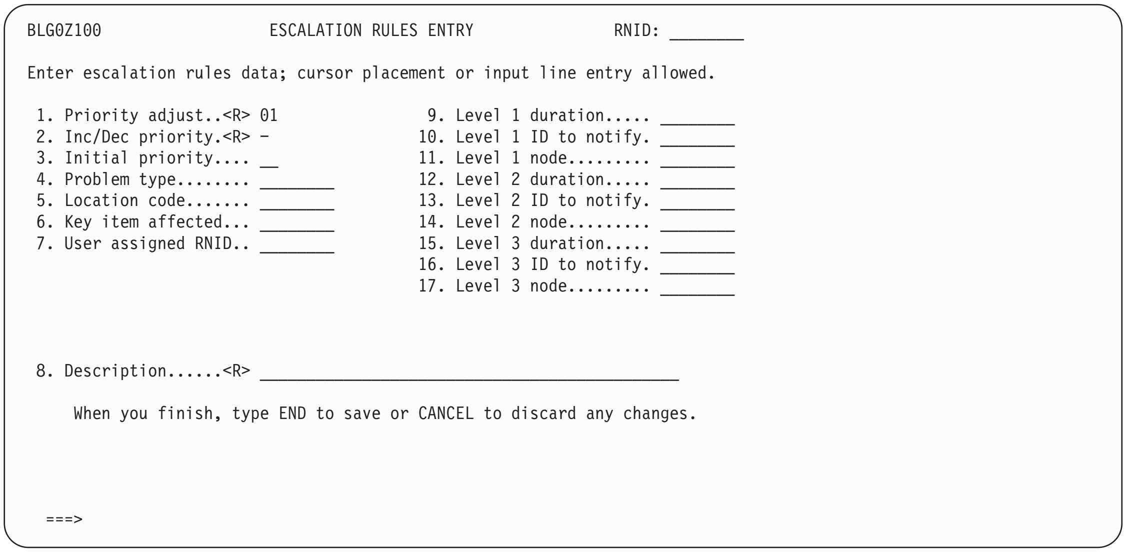
You can add, delete, or modify RULES records at any time. A RULES record that has no escalation criteria results in *all* nonclosed problem records being candidates for escalation. Rules records are Management application records, so be sure to start on Primary Options Menu, BLG0EN20, for the Management application. To get to the Escalation Rules Entry panel, type `5,4` and press Enter.



```
BLG0EN20              --- PRIMARY OPTIONS MENU ---     APPLICATION: MANAGEMENT

OPTIONS:

      1. OVERVIEW.......Display general information and product enhancements.
      2. PROFILE........Display or alter invocation or session defaults.
      3. APPLICATION....Change application, list available applications.
      4. CLASS..........Change current class, list available classes.
      5. ENTRY..........Create a record.
      6. INQUIRY........Search for records.
      7. UTILITY........Copy, display, print, delete, and update records.
      8. GLOSSARY.......Display a list of searchable words in the database.
      9. PMF............Modify or create panels.


           Select an option, enter a command, or type QUIT to exit.


          Tivoli Information Management for z/OS Version 7 Release 1
                 5697-SD9 (C) Copyright IBM Corp., 1981, 2001.

  ===> 5,4
```

The Escalation Rules Entry panel contains several required fields. Some of these fields are primed. The **Current priority** field is primed with a value of 01, and the **Inc/Dec priority** field is primed with a minus sign (-). You can modify both fields.

When the criteria in this RULES record cause a problem record to be escalated, the current priority in the problem record is adjusted based on the values in these fields. The values shown cause the **Priority adjust** field in the problem record to be decremented (because of the minus sign) by 1.



```
BLG0Z100                ESCALATION RULES ENTRY              RNID: ________

Enter escalation rules data; cursor placement or input line entry allowed.

 1. Priority adjust..<R> 01              9. Level 1 duration..... _______
 2. Inc/Dec priority.<R> -              10. Level 1 ID to notify. _______
 3. Initial priority.... __             11. Level 1 node......... _______
 4. Problem type........ ________       12. Level 2 duration..... _______
 5. Location code....... ________       13. Level 2 ID to notify. _______
 6. Key item affected... ________       14. Level 2 node......... _______
 7. User assigned RNID.. ________       15. Level 3 duration..... _______
                                        16. Level 3 ID to notify. _______
                                        17. Level 3 node......... _______



 8. Description......<R> _____________________________________________

     When you finish, type END to save or CANCEL to discard any changes.




  ===>
```

The Level 1, 2, and 3 IDs can be the actual userids on the Level 1, 2, and 3 nodes or they can be aliases as defined in the USERS record. If they are aliases, do not use the node fields.

Understanding the Escalation Search Argument

The escalation facility initially searches for all RULES records. It then builds a search argument for all nonclosed problems that match the criteria found in the first RULES record escalation criteria fields. For example, if the **Initial priority** field in your first RULES record has a value of 03, the search argument finds all nonclosed problems with an initial priority of 03. If the **Problem type** field also contains the entry SOFTWARE, the search argument finds all nonclosed problems with an initial priority of 03 and a problem type of software. This logic is continued for the **Location code** field. When the escalation function finishes processing the problem records that require escalation based on the first RULES record, it continues to the next RULES record and processes the appropriate problem records based on the next RULES record criteria. This process continues until all of the RULES records are used.

Table 47 illustrates how a search argument is built.

Note: The SUSPEND command is issued first so that when you are finished processing the problem records, you return to the next RULES record in the list.

Table 47. Building the Search Argument

Initial Priority	Problem Type	Location Code	Key Item Affected	Search Argument
3				SUS,6,1,SE +¬STAC/CLOSED PRII/3
3	software			SUS,6,1,SE +¬STAC/CLOSED PRII/3 TYPE/SOFTWARE
3	software	A10		SUS,6,1,SE +¬STAC/CLOSED PRII/3 TYPE/SOFTWARE LOCC/A10

Table 47. Building the Search Argument (continued)

Initial Priority	Problem Type	Location Code	Key Item Affected	Search Argument
3	software	A10	DISK03B	SUS,6,1,SE +¬STAC/CLOSED PRII/3 TYPE/SOFTWARE LOCC/A10 COMK/DISK03B

All problem records have an **Escalation level** field that is not seen by the user. An escalation level of 1 is added to a problem record the first time the record is filed. The **Escalation level** field is incremented by 1 each time escalation occurs for this record (up to a maximum value of 4) to enable the escalation processor to determine which levels of escalation have been performed.

If you try to escalate a problem record that was created in your database on some version before 4.1 and the record has not been subsequently updated under 4.1 or a later version, the **Escalation level** field is added to the record the first time the record is updated.

The BLGESCAL TSP processes the RULES records one at a time, locating all of the nonclosed problem records that match the criteria set in each RULES record. Then BLGESCAL determines which records to escalate. The contents of the **Escalation level** field in each problem record determines which levels of duration, node, user ID, date set, and time set to use in the processing. If a record that meets the criteria does not contain an escalation level, an escalation level of 1 is assumed.

Next, the BLGESCAL TSP sends the escalation message, updates the **Current priority** field in the problem record according to the values in the RULES record, and increments the number of the escalation level in the problem record. The maximum escalation level is 4. When this level is reached, it remains at 4 until the problem record is closed.

If the duration value in the RULES record for the current escalation level is blank, the **Escalation level** field is incremented; however, the priority remains unchanged, and no message is sent.

Escalating Problems

Table 48 illustrates how two different RULES records cause problem records to be escalated. These records illustrate how the escalation level and how long the problem has been active determine whether a problem is escalated and whether an alert message is sent.

Table 48. Values for Two Rules Records

Rules Record Field	Rules Record 1	Rules Record 2
Priority adjust	1	1
Inc/Dec Priority	-	+
Initial priority	20	01
Problem type	SWBATCH	
Location code		DALLAS
Key item affected	ANYITEM	ANYITEM
Level 1 duration	00:01:00	00:12:00
Level 1 ID to notify	USER01	USER01
Level 1 node	ABC01	DEF01
Level 2 duration	00:06:00	

Table 48. Values for Two Rules Records (continued)

Rules Record Field	Rules Record 1	Rules Record 2
Level 2 ID to notify	USER02	
Level 2 node	ABC01	
Level 3 duration	07:00:00	01:00:00
Level 3 ID to notify	USER03	USER03
Level 3 node	ABC02	DEF01

Rules Record 1

In this example:

- When a problem with a type of SWBATCH, an initial priority of 20, and a key item affected of ANYITEM has existed for at least one hour and has not been closed, USER01 at ABC01 is sent an alert message, and the **Current priority** field in the problem record is reduced by 1. The escalation level in the record is raised to 2.
- When the same problem has existed for at least 6 hours and has not been closed, USER01 and USER02 at ABC01 are sent a message, and the **Current priority** field in the problem record is reduced by 1. The escalation level in the record is raised to 3.
- When the same problem has existed for at least 7 days and has not been closed, USER01 and USER02 at ABC01 and USER03 at ABC02 are sent a message, and the **Current priority** field in the problem record is reduced by 1. The escalation level in the record is raised to 4.
- Alert messages are sent to all three users and the **Current priority** field is lowered by 1 every 7 days after that (until it reaches 1, where it stops) until the problem record is closed. If the problem record is updated during this time, alert messages are sent every 7 days from when the record was last updated and filed. The escalation level in the problem record, once it reaches 4, does not change.

Rules Record 2

In this example:

- When a problem with a location of DALLAS and an initial priority of 01 has existed for at least 12 hours and has not been closed, USER01 at DEF01 is sent an alert message, and the **Current priority** field in the problem record is raised by 1. The escalation level in the record is raised to 2.
- The next time the escalation batch job runs, Tivoli Information Management for z/OS looks at the **Level 2 duration** field in the RULES record and finds a blank. This tells the system that no escalation message needs to be sent, and the current priority is not raised. The escalation level in the problem record is raised to 3.
- The system checks the **Level 3 duration** field on succeeding cycles, and, when the problem has existed for one day, a message is sent to USER03 at DEF01, as well as to USER01 at DEF01. The **Current priority** field in the record is increased by 1, and the escalation level is raised to 4.
- Assuming that no other updates are made to the problem record, alert messages are sent to both USER01 and USER03, and the **Current priority** field in the problem record is incremented by 1 every day after that until the problem record is closed. If the record is updated, alert messages are sent every day afterward, based on when the record was last updated and filed.

Note: When a record is updated, the escalation clock restarts. For example, if the duration at escalation level 3 is 5 hours and the record is updated 4 hours after the last message was sent, the next message is sent when 9 hours have passed. If the record is not updated again, another message is sent in 5 hours.

This example assumes you are not using the BLGESCX6 TSX to route escalation messages using TCP/IP SMTP.

Sending the Alert Message

Table 49 shows which users are notified at each escalation level.

Table 49. Users Notified at Each Escalation Level

Escalation Level	Duration Used	Date/Time Used	Messages Sent to
1	Level 1 duration	Date/time entered	Level 1 user ID or address
2	Level 2 duration	Date/time entered	Level 1 and 2 user IDs or addresses
3	Level 3 duration	Date/Time entered	Level 1, 2, and 3 user IDs or addresses
4	Level 3 duration	Date/Time last altered	Level 1, 2, and 3 user IDs or addresses

Note: If you want only one user notified in any RULES record, specify that user as a level 3 user, leave levels 1 and 2 blank, and complete only the **Level 3 duration**, **Level 3 ID to notify**, and **Level 3 node** fields. You only need **Level 3 node** if your **Level 3 ID to notify** is a userid. If it is an alias, you do not need to specify anything in the **Level 3 node**. If the level 3 fields are blank, the escalation messages are sent only twice. After the level 1 and level 2 user IDs are notified, no other messages are sent.

Starting Escalation Processing

Problem escalation is designed to be started through a batch job submitted on a periodic basis to MVS. This job starts the BLGESCAL TSP that reads all the RULES records. If you have five different RULES records, the TSP performs five different searches against the nonclosed problem records in the database.

Creating the JCL for the Escalation Batch Job

BLMESCAL in the sample library (SBLMSAMP) contains JCL you can use to start the escalation batch job. Modify BLMESCAL as the prologue directs. The user ID you specify in the JCL must be the TSO logon ID dedicated to the Problem Escalation Facility. The user ID must be in the MASTER privilege class because that class has the authority to update all problem records, regardless of which class is the owning privilege class.

Be sure the JOB card matches the defined standards for your installation. You may need a JOBCAT card if the data sets are not in the master catalog. Also, you may not require a STEPLIB if the Tivoli Information Management for z/OS load modules are in the system link pack area.

Tivoli Information Management for z/OS creates a data set called &ZPREFIX; &ZPREFIX is the ISPF variable set when you enter PROFILE PREFIX(user ID). If a value cannot be found for &ZPREFIX, a data set called &ZUSER.$ESTEMP$ is created. Set the variable by using the TSO profile prefix command.

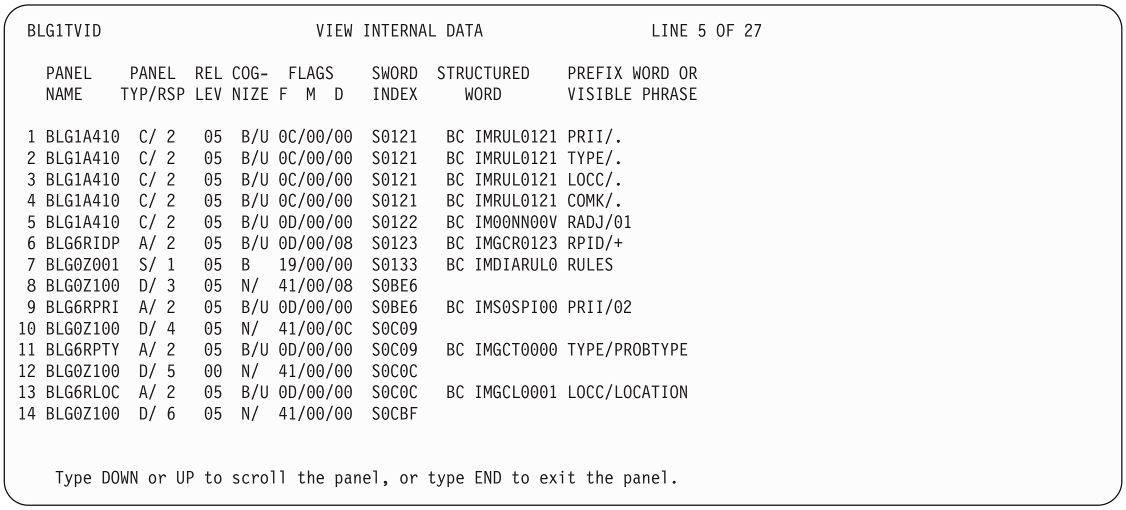
Tailoring the Escalation Facility

An s-word index of 0121 in a RULES record indicates that the prefix collected with it is used as escalation criteria. Escalation processing looks at the p-word associated with each 0121 s-word index and scans the rest of the RULES record for other occurrences of that prefix. If the prefix has data, that data is used in the search argument for escalation processing.

The following example illustrates how an escalation search argument is built.

Note: The line numbers in this example are for reference only.

This panel shows an internal view of a RULES record. The s-word index of 0121 for lines 1 through 4 shows the prefixes that are used as escalation criteria.



```
 BLG1TVID                       VIEW INTERNAL DATA                  LINE 5 OF 27

   PANEL    PANEL  REL COG-  FLAGS    SWORD  STRUCTURED    PREFIX WORD OR
   NAME    TYP/RSP LEV NIZE F  M  D   INDEX     WORD       VISIBLE PHRASE

 1 BLG1A410  C/ 2   05  B/U 0C/00/00  S0121   BC IMRUL0121 PRII/.
 2 BLG1A410  C/ 2   05  B/U 0C/00/00  S0121   BC IMRUL0121 TYPE/.
 3 BLG1A410  C/ 2   05  B/U 0C/00/00  S0121   BC IMRUL0121 LOCC/.
 4 BLG1A410  C/ 2   05  B/U 0C/00/00  S0121   BC IMRUL0121 COMK/.
 5 BLG1A410  C/ 2   05  B/U 0D/00/00  S0122   BC IM00NN00V RADJ/01
 6 BLG6RIDP  A/ 2   05  B/U 0D/00/08  S0123   BC IMGCR0123 RPID/+
 7 BLG0Z001  S/ 1   05  B   19/00/00  S0133   BC IMDIARUL0 RULES
 8 BLG0Z100  D/ 3   05  N/  41/00/08  S0BE6
 9 BLG6RPRI  A/ 2   05  B/U 0D/00/00  S0BE6   BC IMS0SPI00 PRII/02
10 BLG0Z100  D/ 4   05  N/  41/00/0C  S0C09
11 BLG6RPTY  A/ 2   05  B/U 0D/00/00  S0C09   BC IMGCT0000 TYPE/PROBTYPE
12 BLG0Z100  D/ 5   00  N/  41/00/00  S0C0C
13 BLG6RLOC  A/ 2   05  B/U 0D/00/00  S0C0C   BC IMGCL0001 LOCC/LOCATION
14 BLG0Z100  D/ 6   05  N/  41/00/00  S0CBF


    Type DOWN or UP to scroll the panel, or type END to exit the panel.
```

In this example, the TSP first looks in the current record for an occurrence of the prefix PRII/. It finds PRII/02 (line 9) and adds it to the search argument. The TSP then adds TYPE/PROBTYPE (line 11) because of line 2, and it adds LOCC/LOCATION (line 13) because of line 3 to the search argument. Because of line 4, it looks for COMK/ next. No COMK/ p-word has an associated value, so this escalation criteria is not a part of the search argument in this RULES record.

Changing the Escalation Criteria

The escalation facility as shipped from Tivoli uses the following fields in the RULES records:

- Initial priority
- Problem type
- Location code
- Key item affected

To change the escalation criteria:

- Add the p-word (for example, PERH/) to the dictionary, if it is not already there.
- Update control panel BLG1A410 to include the ADD control lines with the s-word index 0121 and the required p-word index for the criteria you want to include in your search argument.
- Update the Escalation Rules panel (BLG0Z100) to reflect the same criteria. For example, if you add location code as an escalation criterion, add the **Location Code** field to panel BLG0Z100.

  You can also delete criteria in the same way. Although this task is optional if you delete criteria, we recommend that you make the modifications to prevent any confusion about which escalation criteria are used.
- Modify the Summary, Display, Inquiry, Utility, and Delete panels to reflect your changes.
- Modify the RFTs that print RULES records.

Changing the Escalation Command Template

The command used to send an escalation alert message is XMIT. If you are using the BLGESCX6 TSX, the alert message is sent through TCP/IP SMTP mail. If your installation uses a different command or uses a different format, change the escalation command template that is defined by a MOVEVAR control line in TSP BLGESC04 and saved in the ESCB.

The format of the command template is:

```
XMIT (&NODE/&ID)  MSGDDNAME(&DDN) NOLOG
```

It can contain one or more of the following variables:

- &NODE (Node of the message recipient)
- &ID (User ID of the message recipient)
- &DDN (ddname)
- &DSN (Data set name of a data set to be sent)
- &MSG (Text of the message)

These variables are the only ones for which you can make substitutions. You cannot specify ISPF or system variables in the command template.

Three sample commands follow:

```
SEND '&MSG.' USER(&ID.) LOGON

XMIT &NODE./&ID.  MSGDDNAME('&DDN.') NOLOG

XMIT &NODE./&ID.  DA('&DSN.') NOLOG
```

Note: Using &DSN with XMIT can degrade performance of the escalation facility. Because XMIT frees the data set automatically when it is finished, the escalation code must re-allocate the data set every time a message is sent.

If you want to use TCP/IP SMTP mail rather than XMIT to send messages, change the command template to:

```
&ID MSGDDNAME(&DDN)
```

&ID. The mail alias from the RULES record

&DDN.
The *ddname* to which the escalation data set is written

When the Escalation Facility determines that it is time to send a message, the command template is extracted from the ESCB. All variables are filled in from the RULES record, the TSCA variable data area, and, if specified, other parameters on the call to user exit BLGESNOT in TSP BLGESC06.

Follow all variables with a period (.). If a variable in the template contains a period, two periods must be present (..). For example, if you use USERID.NODE, your command template must contain &ID.&NODE.

Variables that are not specified are not extracted from the RULES record to be put into the command template. For example, using the previous SEND command template sends the message to a user at the node on which the escalation job is running, ignoring the node specified in the RULES record.

Specify only the variables that your SEND command will use.

Changing the Escalation Message

The escalation notification message shipped with the Escalation Facility reads as follows:

`ALERT: REC=`*record id*, `PRI=`*current priority*, `STAT=`*current status*, `DESC=`*problem description*.

The BLGESC07 TSP builds this message. The BLGESNOT user exit assembles the string of data to be sent to the ISPEXEC module and substitutes the message into the position of the &DSN. or &MSG. variable in the escalation command template. You can modify the escalation message to meet your requirements either by changing TSP BLGESC07 or by building a message panel including the message you want to send.

You can specify a message panel and extract the message from it to create a simpler one, but you are limited to inserting only one piece of variable data into the message. If your message has inserted data, it is processed in the same manner as in the TSP MESSAGE function, and it requires the same input. Refer to the *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference* for more information about the MESSAGE function. Decide which variable you want to insert, such as the RNID, and do the following:

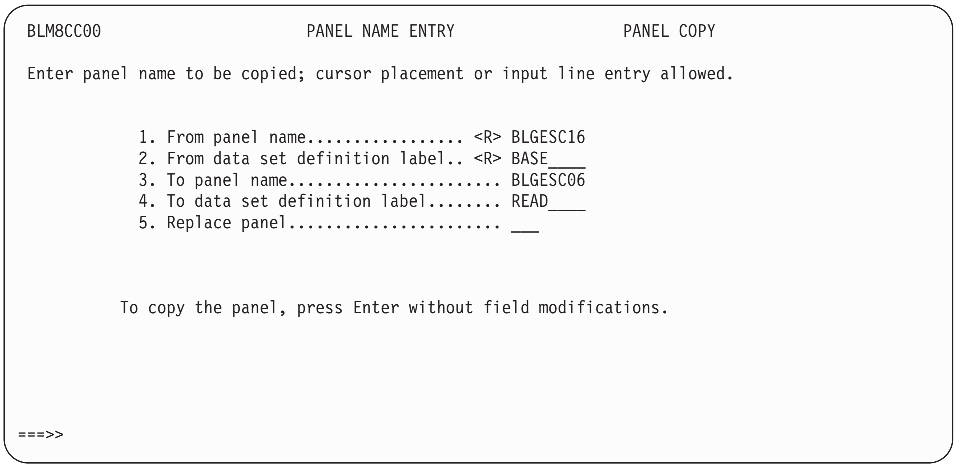
1. Create a message panel containing the new message.
2. Change TSP BLGESC06 by removing its LINK to TSP BLGESC07.

Note: The notification TSPs are changed for Version 7.1 to take advantage of the enhanced e-mail message notification. A copy of TSP BLGESC06 as it appeared prior to these changes has been shipped as TSP BLGESC16. Before you change BLGESC06, follow the steps below to copy BLGESC06 into your modified panel data set from TSP BLGESC16.

a. On BLG0EN20, the Primary Options Menu, type **9** on the command line to go into PMF and press Enter.

b. On BLM8C000, type **3** and press Enter to copy a panel.

c. On BLM8CC00, the Panel Name Entry panel, type BLGESC16 in the **From panel name** field and type the name of your *base* panel data set in the **From data set definition label** field. Type BLGESC06 in the **To panel name** field and type the name of your read panel data set in the **To data set definition label** field. Press Enter twice to cause panel BLGESC16 to be copied to your read panel data set. You will receive a message confirming that the panel was copied.



```
BLM8CC00                     PANEL NAME ENTRY                  PANEL COPY

Enter panel name to be copied; cursor placement or input line entry allowed.


            1. From panel name................. <R> BLGESC16
            2. From data set definition label.. <R> BASE____
            3. To panel name....................... BLGESC06
            4. To data set definition label........ READ____
            5. Replace panel....................... ___



          To copy the panel, press Enter without field modifications.






===>>
```

3. Update the USEREXIT line in BLGESC07 that calls BLGESNOT to include the message panel name in the **Panel name** field.

Using the BLGESCX6 TSX with TCP/IP SMTP

Tivoli Information Management for z/OS provides a TSX that can send escalation notification messages through TCP/IP SMTP rather than through XMIT, using the same logic as the immediate notification TSXs. The BLGESCX6 TSX can be used if you choose to create the escalation message using the existing *ddname* option. You can modify your escalation processing to build the message as desired and send it through any mail processing available on MVS.

If you use the standard XMIT method to send messages, user IDs and nodes are stored in RULES records that define the search criteria for problems to be escalated, and the people to alert at each escalation level. If you use TCP/IP SMTP, you specify aliases in the user ID fields and nothing in the node fields. The aliases represent the actual mail addresses which are in the USERS record. To use the escalation TSX, you must define a new message skeleton that includes a mail alias. The mail alias is stored in the Level 1, 2, and 3 ID fields of the RULES record. TSX BLGESCX6 extracts mail addresses that are defined for the mail alias from the USERS record. You must define the mail aliases and addresses in the USERS record, or customize escalation processing to determine the mail address using an alternative method.

To use the BLGESCX6 TSX, you must first perform the following steps:

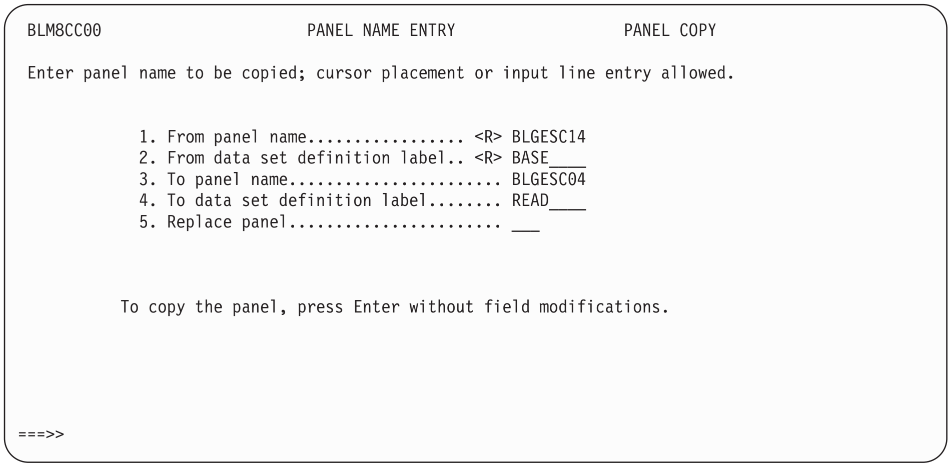
- Store a *mail alias* in the Level 1, 2, and 3 ID fields of the RULES record, instead of the userids and nodes. Then, define the mail aliases in the USERS record in the ASSIGNEE NAME/ALIAS field on BLGLUSER, the Mail Information Entry panel, and include their corresponding mail addresses.
- Modify TSP BLGESC04 to define the message skeleton.

  ```
  &ID;. MSGDDNAME(&DDN)
  ```

  where *&ID*;. is the mail alias obtained from the RULES record, and *&DDN*;. is the *ddname* of a data set containing the escalation message.

  Note: A copy of TSP BLGESC04 as it appeared prior to e-mail notification processing changes in Version 7.1 has been shipped as BLGESC14. Before modifying TSP BLGESC04, use the following steps to copy BLGESC14 to BLGESC04.

  1. On BLG0EN20, the Primary Options Menu, type **9** on the command line to go into PMF and press Enter.
  2. On BLM8C000, type **3** and press Enter to copy a panel.
  3. On BLM8CC00, the Panel Name Entry panel, type BLGESC14 in the **From panel name** field and type the name of your *base* panel data set in the **From data set definition label** field. Type BLGESC04 in the **To panel name** field and type the name of your read panel data set in the **To data set definition label** field. Press Enter twice to cause panel BLGESC14 to be copied to your read panel data set. You will receive a message confirming that the panel was copied.



```
BLM8CC00                     PANEL NAME ENTRY                 PANEL COPY

Enter panel name to be copied; cursor placement or input line entry allowed.


           1. From panel name................. <R> BLGESC14
           2. From data set definition label.. <R> BASE____
           3. To panel name....................... BLGESC04
           4. To data set definition label........ READ____
           5. Replace panel....................... ___



         To copy the panel, press Enter without field modifications.






===>>
```

- Modify TSP BLGESC06. This TSP calls other TSPs and user exits to build the escalation notification message. It also puts the message in a temporary data set, and issues the ISPF command built by the BLGESNOT user exit. To modify BLGESC06:
  - On the calls to user exit BLGESNOT, specify NOCMD in the USER DATA field on panel BLM8CU9P, Data Field Specification. Specifying NOCMD directs the user exit to substitute for the variables in the message skeleton but not build an ISPF command.
  - After the calls to BLGESNOT, replace the ISPEXEC control lines with LINKs to TSX BLGESCX6.

Note: Before modifying TSP BLGESC06, you may need to copy BLGESC16 to BLGESC06. For a description of this process, see page 497.

- Modify TSX BLGESCX6. TSXs supplied with Tivoli Information Management for z/OS reside in the SBLMTSX data set. You should copy the TSX to another data set before making modifications. TSXs that Tivoli Information Management for z/OS invokes must be allocated to the BLGTSX DD.
  - Define the correct TCP/IP SMTP mail header and trailer information for your environment. See “Defining TCP/IP SMTP Header and Trailer Information” on page 453 for more information.
  - Define the logical record length, SYSOUT class, destination, and *ddname* of the data set dynamically allocated to TCP/IP SMTP to match your local environment. See “Defining the SMTP Data Set” on page 453 for more information.
  - Modify the 'From:' mail address. See “Modifying the ″From″ Mail Line” on page 453 for more information.

Notification Management User Exits

The user exits described in this section are intended for use by the escalation facility. Eight of these user exits, BLGESFCB, BLGESGCB, BLGESGET, BLGESPUT, BLGESPUV, BLGNSYAL, BLGNSYFR, and BLGUSERS are also used by immediate notification. After you understand these exits and how they work together, you can use them for your own purposes.

Two possible input types are fields in the TSCA that are expected to be set before the user exit can run and data that is entered when the USEREXIT control line is created. In the first case, the input is described by using the TSCA field name. In the second case, the field name on the USEREXIT panel is specified. The exception to this is the variable data area, which does not have a TSCA name.

Note: User exits which require that information be passed in any field, other than the variable data area (VDA), cannot be run from a TSX. Refer to the Environment section of each of the user exits in this chapter to determine whether or not the user exit can be run from a TSX.

The Escalation Control Block

The escalation control block (ESCB) holds anchor information about a RULES record while a batch escalation job is running.

Table 50 describes the contents of the control block; it also includes the field lengths and the offset of each field into the record. This information is provided for your use in diagnosing problems in the escalation facility. These fields must not be changed.

Table 50. Escalation Control Block Fields

Field Number	Description	Field Length (bytes)	Hex Offset
	Escalation control block	2BE	0
1	ESCB acronym	4	0
2	Rules record RNID	8	4

Table 50. Escalation Control Block Fields (continued)

Field Number	Description	Field Length (bytes)	Hex Offset
3	Reserved	2	C
4	Priority adjust amount	2	E
5	Adjust direction (+ or -)	1	10
6	Reserved	7	11
7	Level 1 duration	8	18
8	Level 1 ID to notify	8	20
9	Level 1 node	8	28
10	Level 2 duration	8	30
11	Level 2 ID to notify	8	38
12	Level 2 node	8	40
13	Level 3 duration	8	48
14	Level 3 ID to notify	8	50
15	Level 3 node	8	58
16	Form of command skeleton	50	60
17	The current date	8	B0
18	The current time	5	B8
19	Reserved	3	BD
	Everything that follows this field is used as escalation criteria and can be repeated a maximum of 30 times	1FE	C0
20	Length of prefix	1	C0
21	Actual prefix	10	C1

BLGESADD–Increment Counter

Adds 1 to the value in TSCAUFLD. This counter has leading zeros in its format.

Input TSCAUFLD as set by a SETFIELD control line

TSCAIPTR containing a pointer to a valid ESCB.

Output
TSCAUFLD is incremented by 1. Possible return and reason codes are listed in Table 51.

Environment
TSP only.

Table 51. BLGESADD Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
4	4	A nonnumeric value was specified in TSCAUFLD.

The BLGESC01 TSP contains an example of this user exit.

BLGESCCL–Escalation Cleanup

Deletes the escalation load modules and frees the temporary data set.

Input TSCAIPTR containing a pointer to a valid ESCB.

Output
Escalation load modules are deleted, and the temporary data set is freed. Possible return and reason codes are listed in Table 52.

Environment
TSP only.

Table 52. BLGESCCL Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
8	4	ESCB pointer is not valid.
8	8	Internal error was detected.
8	C	Error occurred freeing data set. Data set was not freed.

The BLGESCAL TSP contains an example of this user exit.

BLGESCKE–Check Escalation

Checks to see if an escalation job is in process.

Input None.

Output
Return and reason codes, as listed in Table 53.

Environment
TSP only.

Table 53. BLGESCKE Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Escalation is in process.
4	4	Escalation is not in process.

The BLGNOTPU TSP contains an example of using this user exit.

BLGESCLR–Clear Control Block

Resets the ESCB fields to binary zeroes.

Input TSCAIPTR containing a pointer to a valid ESCB.

Output
ESCB is cleared. Possible return and reason codes are listed in Table 54 on page 502.

Environment
TSP only.

Table 54. BLGESCLR Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
8	4	ESCB pointer is not valid.

The BLGESC03 and BLGESC04 TSPs contain examples of using this user exit.

BLGESDAT–Date and Time

Puts the system date and time into the TSCA variable area.

Input The TSCA containing the contents of the **Get variable data** field in the Flag Field Specification panel of the USEREXIT control line.

YES The date and time is appended to the existing variable data.

NO The date and time overlay the existing variable data.

Output
TSCAVDAL containing the length of date and time data in the variable data area; variable data area containing the date in the YY/MM/DD format and the time in the HH:MM format. Return and reason codes are listed in Table 55.

Environment
TSP only.

Table 55. BLGESDAT Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.

The BLGESC06 TSP contains an example of this user exit.

BLGESDUR–Duration

Locates the date and time fields and determines whether the duration (1, 2, or 3 depending on TSCAUFLD) specified by the current RULES record is met for notifying a user ID. A problem record must be in update mode, and the escalation control block must contain the appropriate duration.

Input TSCAIPTR containing a pointer to a valid escalation control block.

TSCAUFLD containing the level of current escalation.

The **Structured word index** field in the Data Field Specification panel of the USEREXIT control line to use for comparison for escalation levels 1 to 3 (the **Date entered** field, as shipped for TSP BLGESC06).

The **New Structured word index** field in the Data Field Specification panel of the USEREXIT control line to use for comparing escalation levels (the **Time entered** field, as shipped for TSP BLGESC06).

Output
Return and reason codes as listed in Table 56 on page 503.

Environment
TSP only.

Table 56. BLGESDUR Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Time to escalate this problem record.
4	0	Not time to escalate this problem record.
4	4	Duration is blank in the RULES record.
4	8	Date or time was not specified in the problem record.
8	4	ESCB pointer is not valid.
8	8	Internal error was detected.
8	C	Date or time conversion error.
8	10	Level number in TSCAUFLD is not valid.

The BLGESC06 TSP contains an example of this user exit.

BLGESFCB–Free Control Block

Frees the escalation control block. This must be the last user exit processed in the escalation function.

Input TSCAIPTR containing a pointer to a valid escalation control block.

Output
TSCAIPTR is set to 0. Possible return and reason codes are listed in Table 57.

Environment
TSP only.

Table 57. BLGESFCB Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
8	4	ESCB pointer is not valid.
8	8	Internal error was detected.

The following TSPs contain examples of using this user exit:
BLGESCAL
BLGNOTAE
BLGNOTAU
BLGNOTCE
BLGNOTCU

BLGESGCB–Get Control Block Storage

Gets storage for the ESCB and anchors it in the TSCA. This user exit must be called before any other escalation user exit is called.

Input None.

Output

TSCAIPTR containing a pointer to a valid ESCB. Possible return and reason codes are listed in Table 58.

Environment

TSP only.

Table 58. BLGESGCB Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Escalation control block was set up successfully.
8	8	Internal error was detected.

The BLGESCAL TSP contains an example of this user exit.

BLGESGET–Get Control Block Field

Gets a field stored by BLGESPUT, BLGESPUV, or a user-written user exit from the ESCB and stores it in the TSCA variable data area.

Input TSCAIPTR containing a pointer to a valid ESCB.

The **Literal/Test data** field in the Data Field Specification panel of the USEREXIT control line calling this exit and containing the hex length of the data to be moved.

The **New data** field in the Data Field Specification panel of the USEREXIT control line calling this exit and containing the 1- to 7-byte hex offset within the escalation control block of the data to be moved.

The **Get variable data** field.

YES To append the data to the end of the variable data area.

NO To replace the data in the variable data area.

Output

TSCAVDAL containing the length of the data in the variable data area. The variable data area is updated with data from the ESCB. Possible return and reason codes are listed in Table 59.

Environment

TSP only.

Table 59. BLGESGET Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
4	4	The length specified in the **Literal/Test data** field is not valid.
4	8	The offset specified in the **New data** field is not valid.
8	4	ESCB pointer is not valid.

The following TSPs contain examples of this user exit:

BLGESC01
BLGNOTAE
BLGNOTAU
BLGNOTCE
BLGNOTCU
BLGESC02

BLGESINI–Initialize

Loads the escalation user exits and allocates a temporary data set that you can use with the BLGESNOT user exit to send the escalation message.

Input TSCAIPTR containing a pointer to a valid ESCB.

Output
Environment for the escalation function.

Environment
TSP only.

Possible return and reason codes are listed in Table 60.

Table 60. BLGESINI Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
8	4	ESCB pointer is not valid.
8	8	Internal error was detected.
8	C	An error occurred while allocating data set.

The BLGESCAL TSP contains an example of this user exit.

BLGESLVL–Level Increment

Increases the escalation level by 1 in the current problem record ESCB, if possible.

Input TSCAIPTR containing a pointer to a valid ESCB.

Output
Escalation level incremented by 1 up to a maximum of 4. No processing occurs if the record has no escalation level or an escalation level of 4. Possible return and reason codes are listed in Table 61.

Environment
TSP only.

Table 61. BLGESLVL Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
4	4	No escalation level field was found in the problem record.
8	8	Internal error was detected.

The BLGESC06 TSP contains an example of this user exit.

BLGESNOT–Notify

Builds the command that notifies a user about a problem record. The command is built and stored in the TSCA variable data area, overlaying what was there. The message comes from one of the following:

- The message line in an input message panel
- Text in the input variable data area

If NOCMD is specified in the User data field on BLM8CU9P, the Data Field Specification panel, the BLGESNOT user exit does not build an ISPF message; rather, it performs a substitution for the variables in the message skeleton.

To send a message from a message panel:

Input TSCAIPTR containing a pointer to a valid ESCB.

TSCAUFLD containing the level of current escalation set by a SETFIELD control line or the BLGESADD user exit.

The **Panel name** field in the Data Field Specification panel of the USEREXIT control line calling this user exit and containing the message panel name from which to extract a message.

TSCAVDAL containing the length of the send command.

Output
TSCAVDAL containing the length of the send command.

Variable data area containing the send command in a format that can be passed directly to ISPEXEC.

Environment
TSP only.

To send a message from a string of data:

Input TSCAIPTR containing a pointer to a valid ESCB.

TSCAUFLD containing the level of current escalation set by a SETFIELD control line in a TSP or the BLGESADD user exit.

TSCAVDAL containing the length of the message to be sent.

Variable data area containing the notification message if no message panel is used.

Note: The message must be created by any combination of a MOVEVAR control line, the BLGESGET user exit, and a user-written user-exit routine.

Output
TSCAVDAL containing the length of the send command.

The variable data area containing the send command in a format that can be passed directly to ISPEXEC.

Environment
TSP only.

Possible return and reason codes are listed in Table 62.

Table 62. BLGESNOT Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Command was built successfully.
4	4	TSCAUFLD does not contain the current level of escalation.
8	4	ESCB pointer is not valid.
8	8	Internal error was detected.
8	C	Error occurred in processing data set.
8	10	Message text is supposed to come from the variable data area, but variable data length (TSCAVDAL) is zero.
8	14	Message text is supposed to come from a message panel, but the panel cannot be loaded.
8	18	Message text is supposed to come from a message panel, but the panel has no text.
8	1C	Panel specified for message text is not a message panel.
8	20	Command exceeds length of variable data area.

The BLGESC06 TSP contains an example of this user exit.

BLGESPRI–Priority Update

Increments the value associated with an input prefix by the priority adjust amount in the escalation control block. The record must be in update mode.

Input TSCAIPTR containing a pointer to a valid ESCB.

The **Literal/Test data** field in the Data Field Specification panel of the USEREXIT control line to be escalated (the **Current Priority** field as shipped for TSP BLGESC06).

Output

The specified field in the current record is incremented by the specified amount.

Environment

TSP only.

Possible return and reason codes are listed in Table 63.

Table 63. BLGESPRI Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Priority was adjusted successfully.
0	4	Priority adjustment amount equals 0.
0	8	Priority reached maximum (01 if the priority is being decremented, 99 if the priority is being incremented); priority is not adjusted.

Table 63. BLGESPRI Return and Reason Codes (continued)

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
4	4	Current priority field (the prefix specified in the **Literal/Test data** field) is not in the problem record; priority is not adjusted.
4	8	Rules record contains a priority adjustment amount that is not valid; priority is not adjusted.
4	C	The prefix in the **Literal/Test data** field is not valid, or the data in the problem record for that prefix is greater than two characters.
4	10	The data in the problem record associated with the prefix in the **Literal/Test data** field is nonnumeric.
8	4	ESCB pointer is not valid.
8	8	Internal error was detected.

The BLGESC06 TSP contains an example of this user exit.

BLGESPUT–Put TSCA Data in Control Block

Gets data from the TSCA and stores it in the escalation control block. This data can be retrieved by user-exit BLGESGET or a user-written exit routine.

Input TSCAIPTR containing a pointer to a valid ESCB.

TSCASDF containing data (collected by a FINDSDATA control line) to be moved into the ESCB.

The **New data** field in the Data Field Specification panel of the USEREXIT control line containing the 1- to 7-byte hex offset within the ESCB where the data is to be moved.

Output

ESCB updated with the data from the TSCA at the specified offset. Possible return and reason codes are listed in Table 64.

Environment

TSP only.

Table 64. BLGESPUT Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful move.
4	4	The length specified in TSCASDFL is not valid.
4	8	The offset specified in the **New data** field is not valid.
8	4	ESCB pointer is not valid.

The following TSPs contain examples of this user exit:

BLGESC03
BLGNOTAE
BLGNOTAU

BLGNOTCE
BLGNOTCU

BLGESPUV–Put Variable Data

Gets data from the variable data area or literal data and stores it in the ESCB.

To move variable data:

Input TSCAIPTR containing a pointer to a valid ESCB.

Get variable data=YES in the Flag Field Specification panel of the USEREXIT control line.

TSCAVDAL containing the length of data to be moved; the TSCA variable data area containing the data to be moved.

The **New data** field in the Data Field Specification panel of the USEREXIT control line containing the 1- to 7-byte hex offset where the data is moved into the ESCB.

Note: Get variable data=YES and **Literal/Test data** are mutually exclusive. If both are specified, data is moved from the variable data area, and **Literal/Test data** is ignored.

Output
ESCB is updated with the contents of the variable data area.

Environment
TSP only.

To move literal data:

Input TSCAIPTR containing a pointer to a valid ESCB.

Literal/Test data field in the Data Field Specification panel of the USEREXIT control line calling this exit. This field contains the data to be moved into the ESCB.

New data field in the Data Field Specification panel of the USEREXIT control line containing the 1- to 7-byte hex offset where the data is moved into the ESCB.

Note: Get variable data= YES and **Literal/Test data** are mutually exclusive. If both are specified, data is moved from the variable data area, and **Literal/Test data** is ignored.

Output
ESCB is updated with the input value. Possible return and reason codes are listed in Table 65.

Environment
TSP only.

Table 65. BLGESPUV Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.

Table 65. BLGESPUV Return and Reason Codes (continued)

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
4	4	The length specified in TSCAVDAL or in the **Literal/Test data** field is not valid.
4	8	Offset specified in the **New data** field is not valid.
4	C	Neither variable nor literal data was specified for move (**Get variable data** = NO or blank, and the **Literal/Test data** field is blank).
8	4	ESCB pointer is not valid.

The BLGESC04 and BLGESC06 TSPs contain examples of this user exit.

BLGESSCT–Store Criteria

Scans the current RULES record for an occurrence of escalation criteria (s-word index 0121) and stores its associated prefix in the next available spot in the ESCB.

Input TSCAIPTR containing a pointer to a valid ESCB.

TSCAUFLD containing a 1- to 7-byte index of the next available slot.

TSCARPD containing the prefix of the escalation criteria (collected from running a FINDSDATA control line) to be moved.

Output

The next available slot in the ESCB is filled. Possible return and reason codes are listed in Table 66.

Environment

TSP only.

Table 66. BLGESSCT Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
4	4	The p-word contained in TSCARPD is not valid.
4	8	The p-word length contained in TSCARPDL is not valid.
4	C	Limit of criteria fields (30) has been reached.
8	4	ESCB pointer is not valid.

The BLGESC01 TSP contains an example of this user exit.

BLGESSEA–Get Escalation Criteria

Scans the current RULES record for an occurrence of data for each escalation criterion, and adds any criteria found to the variable data area as part of a search argument. The search argument locates problem records that meet the escalation criteria in the RULES record.

Input TSCAIPTR containing a pointer to a valid ESCB. The control block is valid if the return code is 0.

TSCAUFLD containing the number of criteria in the ESCB.

The TSCA variable data area containing `6,1,SE+` (the base argument for finding problem records).

Output

TSCAVDAL containing the length of data in the variable data area.

The TSCA variable data area containing the complete search argument for problem records that match the criteria in this RULES record.

Possible return and reason codes are listed in Table 67.

Environment

TSP only.

Table 67. BLGESSEA Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Successful completion.
4	4	TSCAUFLD did not contain the number of criteria in the ESCB.
4	8	Variable data area is full.
8	4	ESCB pointer is not valid.
8	8	Internal error was detected.

The BLGESC01 TSP contains an example of this user exit.

BLGNSYAL–Allocate Data Set to SYSOUT

Dynamically allocates a data set to SYSOUT with a user-specified destination. Only one data set can be allocated at a time and must be freed using user exit BLGNSYFR.

Input

```
Call BLGTSX 'USEREXIT','BLGNSYAL',ddname,lrecl,class,node,userid
```

- *ddname* is the optional *ddname* to which to allocate the data set. The *ddname* is optional. If it is not specified, the system assigns the *ddname* and it is returned in the REXX variable BLG_NOTIFYDD.
- *lrecl* is the logical record length to use to allocate the data set (1 to 3 digits). This is a required parameter.
- *class* is the SYSOUT class to which to allocate the data set (1 character). This is a required parameter.
- *node* and *userid* represent the SYSOUT destination. For immediate notification and escalation, this targets the TCP/IP SMTP server on the system where the server resides. The node is required and it must consist of 1 to 8 characters. The userid is optional and, if specified, must consist of 1 to 8 characters. If the userid is not specified, it defaults to SMTP.

Output

A SYSOUT data set is allocated using the specified inputs and a variable (V) record format. If a SYSOUT data set was previously allocated and not freed using BLGNSYFR, nothing occurs.

Environment

TSX only.

Possible return and reason codes are listed in Table 68.

Table 68. BLGNSYAL Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Processing successful, SYSOUT data set allocated.
8	4	SYSOUT data set is already allocated and not freed with BLGNSYFR. The data set remains allocated.
8	8	BLGNSYAL was called from a TSP. Only a TSX can call it.
8	12	Logic error. Internal control blocks could not be located. No addresses are returned.
12	4	No parameters were specified.
12	8	The DD name length is not valid.
12	12	The LRECL was either not specified, greater than 3 characters, or not numeric.
12	16	The SYSOUT class was not specified or it is more than 1 character.
12	20	The node was not specified or the length is not valid.
12	24	The user ID length is not valid.
16	4	An error occurred while allocating the SYSOUT data set. Additional messages may have been issued that identify the specific problem.

The BLGTXNOT TSX contains an example of this user exit.

BLGNSYFR–Free Data Set Allocated by BLGNSYAL

Frees the SYSOUT data set allocated by user exit BLGNSYAL.

Input

```
Call BLGTSX 'USEREXIT','BLGNSYFR'
```

Output

The SYSOUT data set is freed.

Environment

TSX only.

Possible return and reason codes are listed in Table 69.

Table 69. BLGNSYFR Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Processing successful, SYSOUT data set is freed.
0	4	SYSOUT data set was not previously allocated by user exit BLGNSYAL.

Table 69. BLGNSYFR Return and Reason Codes (continued)

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
8	8	BLGNSYFR was called from a TSP. Only a TSX can call it.
8	12	Logic error. Internal control blocks could not be located. No addresses are returned.
16	4	An error occurred while freeing the SYSOUT data set. Additional messages may have been issued that identify the specific problem.

The BLGTXNOT TSX contains an example of this user exit.

BLGUSERS–Extract Mail Address from USERS Record

Extracts the mail addresses from the USERS record that match an input name or mail alias.

Input Call BLGTSX ’USEREXIT’,’BLGUSERS’,*name,databaseid,*’SAVE’ or ’NOSAVE’

- *name* - The assignee name or mail alias that is defined in the USERS record. The name is required and must consist of 1 to 60 characters. If more than 60 characters are specified, the name is truncated to 60 characters.
- *databaseid* - An optional single number identifying the database from which to read the USERS record. If more than 1 character is specified only the first character is used. The default is the number 5.
- SAVE or NOSAVE - Saves the USERS record in storage, or does not save the record in storage. This parameter is required.

Output
Addresses are written to REXX stem BLG_ADDRESS. BLG_ADDRESS.0 contains the number of addresses returned. The maximum number of addresses that can be returned is 200.

Environment
TSX only.

Possible return and reason codes are listed in Table 70 on page 514.

Processing Notes:

You can specify that the USERS record be maintained in storage and not re-read on each call, unless another user updates it. If the user exit is called with the NOSAVE option the record is not maintained in storage and is re-read each time BLGUSERS is invoked. The NOSAVE option directs BLGUSERS to unconditionally read the USERS record and not maintain it in storage. By using BLGUSERS, your TSX does not have to explicitly display the USERS record in order to extract mail addresses from the list of names and addresses. BLGUSERS:

- Assumes the USERS record name/alias and address lists are sorted by name or alias. All entries with the same name must be together in the list.

- The name or alias length must be less than or equal to 60 characters.
- The address list in the USERS record must be defined as either string data or a maximum response of one word.
- Each name or alias in the USERS record must have a corresponding address, and vice versa.

If desired, TSXs can also access the USERS record data through normal TSP processing (by displaying the record and using FINDSDATA and/or GETLIST to search for and retrieve data).

Table 70. BLGUSERS Return and Reason Codes

Return Code (TSCAFRET)	Reason Code (TSCAFRES)	Description
0	0	Processing successful, at least one address is returned.
0	4	At least one name in the list did not have a corresponding address. At least one address is returned.
0	8	The number of addresses found exceeds the maximum allowed. The maximum number of addresses is returned.
4	4	The USERS record was not found. No addresses are returned.
4	8	The input name was not located in the USERS record or no address was defined for the first name occurrence. No addresses are returned.
8	4	Out of storage. No addresses are returned.
8	8	BLGUSERS was called from a TSP. Only a TSX can call it.
8	12	Logic error. The internal control blocks could not be located. No addresses are returned.
12	4	No parameters were specified or the assignee name or mail alias was not specified. Name and SAVE/NOSAVE are required. No addresses are returned.
12	8	The action parameter must be SAVE or NOSAVE. No addresses are returned.
12	12	First character of the name parameter was a blank. Remove leading blanks. No addresses are returned.
16	4	An error occurred while reading the USERS record. Other messages may be issued that indicate the specific error that occurred. No addresses are returned.
16	8	The specified or default database could not be accessed. No addresses are returned.

The BLGTXNOT TSX contains an example of this user exit.

E

Relating Publications to Specific Tasks

Your data processing organization can have many different users performing many different tasks. The books in the Tivoli Information Management for z/OS library contain task-oriented scenarios to teach users how to perform the duties specific to their jobs.

The following table describes the typical tasks in a data processing organization and identifies the Tivoli Information Management for z/OS publication that supports those tasks. See "The Tivoli Information Management for z/OS Library" on page 521 for more information about each book.

Typical Tasks

Table 71. Relating Publications to Specific Tasks

If You Are:	And You Do This:	Read This:
Planning to Use Tivoli Information Management for z/OS	Identify the hardware and software requirements of Tivoli Information Management for z/OS. Identify the prerequisite and corequisite products. Plan and implement a test system.	*Tivoli Information Management for z/OS Planning and Installation Guide and Reference*
Installing Tivoli Information Management for z/OS	Install Tivoli Information Management for z/OS. Define and initialize data sets. Create session-parameters members.	*Tivoli Information Management for z/OS Planning and Installation Guide and Reference* *Tivoli Information Management for z/OS Integration Facility Guide*
	Define and create multiple Tivoli Information Management for z/OS BLX-SPs.	*Tivoli Information Management for z/OS Planning and Installation Guide and Reference*
	Define and create APPC transaction programs for clients.	*Tivoli Information Management for z/OS Client Installation and User's Guide*
	Define coupling facility structures for sysplex data sharing.	*Tivoli Information Management for z/OS Planning and Installation Guide and Reference*
Diagnosing problems	Diagnose problems encountered while using Tivoli Information Management for z/OS	*Tivoli Information Management for z/OS Diagnosis Guide*

Table 71. Relating Publications to Specific Tasks (continued)

If You Are:	And You Do This:	Read This:
Administering Tivoli Information Management for z/OS	Manage user profiles and passwords. Define and maintain privilege class records. Define and maintain rules records.	*Tivoli Information Management for z/OS Program Administration Guide and Reference* *Tivoli Information Management for z/OS Integration Facility Guide*
	Define and maintain USERS record. Define and maintain ALIAS record. Implement GUI interface. Define and maintain command aliases and authorizations.	*Tivoli Information Management for z/OS Program Administration Guide and Reference*
	Implement and administer Notification Management. Create user-defined line commands. Define logical database partitioning.	*Tivoli Information Management for z/OS Program Administration Guide and Reference*
	Create or modify GUI workstation applications that can interact with Tivoli Information Management for z/OS. Install the Tivoli Information Management for z/OS Desktop on user workstations.	*Tivoli Information Management for z/OS Desktop User's Guide*
Maintaining Tivoli Information Management for z/OS	Set up access to the data sets. Maintain the databases. Define and maintain privilege class records.	*Tivoli Information Management for z/OS Planning and Installation Guide and Reference* *Tivoli Information Management for z/OS Program Administration Guide and Reference*
	Define and maintain the BLX-SP. Run the utility programs.	*Tivoli Information Management for z/OS Operation and Maintenance Reference*
Programming applications	Use the application program interfaces.	*Tivoli Information Management for z/OS Application Program Interface Guide*
	Use the application program interfaces for Tivoli Information Management for z/OS clients.	*Tivoli Information Management for z/OS Client Installation and User's Guide*
	Create Web applications using or accessing Tivoli Information Management for z/OS data.	*Tivoli Information Management for z/OS World Wide Web Interface Guide*

Table 71. Relating Publications to Specific Tasks (continued)

If You Are:	And You Do This:	Read This:
Customizing Tivoli Information Management for z/OS	Design and implement a Change Management system. Design and implement a Configuration Management system. Design and implement a Problem Management system.	*Tivoli Information Management for z/OS Problem, Change, and Configuration Management*
	Design, create, and test terminal simulator panels or terminal simulator EXECs. Customize panels and panel flow.	*Tivoli Information Management for z/OS Terminal Simulator Guide and Reference* *Tivoli Information Management for z/OS Panel Modification Facility Guide*
	Design, create, and test Tivoli Information Management for z/OS formatted reports.	*Tivoli Information Management for z/OS Data Reporting User's Guide*
	Create a bridge between NetView® and Tivoli Information Management for z/OS applications. Integrate Tivoli Information Management for z/OS with Tivoli distributed products.	*Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications*
Assisting Users	Create, search, update, and close change, configuration, or problem records. Browse or print Change, Configuration, or Problem Management reports.	*Tivoli Information Management for z/OS Problem, Change, and Configuration Management*
	Use the Tivoli Information Management for z/OS Integration Facility.	*Tivoli Information Management for z/OS Integration Facility Guide*
Using Tivoli Information Management for z/OS	Learn about the Tivoli Information Management for z/OS panel types, record types, and commands. Change a user profile.	*Tivoli Information Management for z/OS User's Guide*
	Learn about Problem, Change, and Configuration Management records.	*Tivoli Information Management for z/OS Problem, Change, and Configuration Management*
	Receive and respond to Tivoli Information Management for z/OS messages.	*Tivoli Information Management for z/OS Messages and Codes*
	Design and create reports.	*Tivoli Information Management for z/OS Data Reporting User's Guide*

F

Tivoli Information Management for z/OS Courses

Education Offerings

Tivoli Information Management for z/OS classes are available in the United States and in the United Kingdom. For information about classes outside the U.S. and U.K., contact your local IBM representative or visit **http://www.training.ibm.com** on the World Wide Web.

United States

IBM Education classes can help your users and administrators learn how to get the most out of Tivoli Information Management for z/OS. IBM Education classes are offered in many locations in the United States and at your own company location.

For a current schedule of available classes or to enroll, call 1-800-IBM TEACh (1-800-426-8322). On the World Wide Web, visit:

http://www.training.ibm.com

to see the latest course offerings.

United Kingdom

In Europe, the following public courses are held in IBM's central London education centre at the South Bank at regular intervals. On-site courses can also be arranged.

For course schedules and to enroll, call Enrollments Administration on 0345 581329, or send an e-mail note to:

contact_educ_uk@vnet.ibm.com

On the World Wide Web, visit:

http://www.europe.ibm.com/education-uk

to see the latest course offerings.

G

Where to Find More Information

The Tivoli Information Management for z/OS library is an integral part of Tivoli Information Management for z/OS. The books are written with particular audiences in mind. Each book covers specific tasks.

The Tivoli Information Management for z/OS Library

The publications shipped automatically with each Tivoli Information Management for z/OS Version 7.1 licensed program are:

- *Tivoli Information Management for z/OS Application Program Interface Guide*
- *Tivoli Information Management for z/OS Client Installation and User's Guide* *
- *Tivoli Information Management for z/OS Data Reporting User's Guide* *
- *Tivoli Information Management for z/OS Desktop User's Guide*
- *Tivoli Information Management for z/OS Diagnosis Guide* *
- *Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications* *
- *Tivoli Information Management for z/OS Integration Facility Guide* *
- *Tivoli Information Management for z/OS Licensed Program Specification*
- *Tivoli Information Management for z/OS Master Index, Glossary, and Bibliography*
- *Tivoli Information Management for z/OS Messages and Codes*
- *Tivoli Information Management for z/OS Operation and Maintenance Reference*
- *Tivoli Information Management for z/OS Panel Modification Facility Guide*
- *Tivoli Information Management for z/OS Planning and Installation Guide and Reference*
- *Tivoli Information Management for z/OS Program Administration Guide and Reference*
- *Tivoli Information Management for z/OS Problem, Change, and Configuration Management**
- *Tivoli Information Management for z/OS Reference Summary*
- *Tivoli Information Management for z/OS Terminal Simulator Guide and Reference*
- *Tivoli Information Management for z/OS User's Guide*
- *Tivoli Information Management for z/OS World Wide Web Interface Guide*

Note: Publications marked with an asterisk (*) are shipped in softcopy format only.

Also included is the Product Kit, which includes the complete online library on CD-ROM.

To order a set of publications, specify order number SBOF-7028-00.

Additional copies of these items are available for a fee.

Publications can be requested from your Tivoli or IBM representative or the branch office serving your location. Or, in the U.S., you can call the IBM Publications order line directly by dialing 1-800-879-2755.

The following descriptions summarize all the books in the Tivoli Information Management for z/OS library.

Tivoli Information Management for z/OS Application Program Interface Guide, SC31-8737-00, explains how to use the low-level API, the high-level API, and the REXX interface to the high-level API. This book is written for application and system programmers who write applications that use these program interfaces.

Tivoli Information Management for z/OS Client Installation and User's Guide, SC31-8738-00, describes and illustrates the setup and use of Tivoli Information Management for z/OS's remote clients. This book shows you how to use Tivoli Information Management for z/OS functions in the AIX, CICS®, HP-UX, OS/2®, Sun Solaris, Windows NT®, and OS/390 UNIX System Services environments. Also included in this book is complete information about using the Tivoli Information Management for z/OS servers.

Tivoli Information Management for z/OS Data Reporting User's Guide, SC31-8739-00, describes various methods available to produce reports using Tivoli Information Management for z/OS data. It describes Tivoli Decision Support for Information Management (a Discovery Guide for Tivoli Decision Support), the Open Database Connectivity (ODBC) Driver for Tivoli Information Management for z/OS, and the Report Format Facility. A description of how to use the Report Format Facility to modify the standard reports provided with Tivoli Information Management for z/OS is provided. The book also illustrates the syntax of report format tables (RFTs) used to define the output from the Tivoli Information Management for z/OS REPORT and PRINT commands. It also includes several examples of modified RFTs.

Tivoli Information Management for z/OS Desktop User's Guide, SC31-8740-00, describes how to install and use the sample application provided with the Tivoli Information Management for z/OS Desktop. The Tivoli Information Management for z/OS Desktop is a Java-based graphical user interface for Tivoli Information Management for z/OS. Information on how to set up data model records to support the interface and instructions on using the Desktop Toolkit to develop your own Desktop application are also provided.

Tivoli Information Management for z/OS Diagnosis Guide, GC31-8741-00, explains how to identify a problem, analyze its symptoms, and resolve it. This book includes tools and information that are helpful in solving problems you might encounter when you use Tivoli Information Management for z/OS.

Tivoli Information Management for z/OS Guide to Integrating with Tivoli Applications, SC31-8744-00, describes the steps to follow to make an automatic connection between NetView and Tivoli Information Management for z/OS applications. It also explains how to customize the application interface which serves as an application enabler for the NetView Bridge and discusses the Tivoli Information Management for z/OS NetView AutoBridge. Information on interfacing Tivoli Information Management for z/OS with other Tivoli management software products or components is provided for Tivoli Enterprise Console, Tivoli Global Enterprise Manager, Tivoli Inventory, Tivoli Problem Management, Tivoli Software Distribution, and Problem Service.

Tivoli Information Management for z/OS Integration Facility Guide, SC31-8745-00, explains the concepts and structure of the Integration Facility. The Integration Facility provides a task-oriented interface to Tivoli Information Management for z/OS that makes the

Tivoli Information Management for z/OS applications easier to use. This book also explains how to use the panels and panel flows in your change and problem management system.

Tivoli Information Management for z/OS Master Index, Glossary, and Bibliography, SC31-8747-00, combines the indexes from each hardcopy book in the Tivoli Information Management for z/OS library for Version 7.1. Also included is a complete glossary and bibliography for the product.

Tivoli Information Management for z/OS Messages and Codes, GC31-8748-00, contains the messages and completion codes issued by the various Tivoli Information Management for z/OS applications. Each entry includes an explanation of the message or code and recommends actions for users and system programmers.

Tivoli Information Management for z/OS Operation and Maintenance Reference, SC31-8749-00, describes and illustrates the BLX-SP commands for use by the operator. It describes the utilities for defining and maintaining data sets required for using the Tivoli Information Management for z/OS licensed program, Version 7.1.

Tivoli Information Management for z/OS Panel Modification Facility Guide, SC31-8750-00, gives detailed instructions for creating and modifying Tivoli Information Management for z/OS panels. It provides detailed checklists for the common panel modification tasks, and it provides reference information useful to those who design and modify panels.

Tivoli Information Management for z/OS Planning and Installation Guide and Reference, GC31-8751-00, describes the tasks required for installing Tivoli Information Management for z/OS. This book provides an overview of the functions and optional features of Tivoli Information Management for z/OS to help you plan for installation. It also describes the tasks necessary to install, migrate, tailor, and start Tivoli Information Management for z/OS.

Tivoli Information Management for z/OS Problem, Change, and Configuration Management, SC31-8752-00, helps you learn how to use Problem, Change, and Configuration Management through a series of training exercises. After you finish the exercises in this book, you should be ready to use other books in the library that apply more directly to the programs you use and the tasks you perform every day.

Tivoli Information Management for z/OS Program Administration Guide and Reference, SC31-8753-00, provides detailed information about Tivoli Information Management for z/OS program administration tasks, such as defining user profiles and privilege classes and enabling the GUI user interface.

Tivoli Information Management for z/OS Reference Summary, SC31-8754-00, is a reference booklet containing Tivoli Information Management for z/OS commands, a list of p-words and s-words, summary information for PMF, and other information you need when you use Tivoli Information Management for z/OS.

Tivoli Information Management for z/OS Terminal Simulator Guide and Reference, SC31-8755-00, explains how to use terminal simulator panels (TSPs) and EXECs (TSXs) that let you simulate an entire interactive session with a Tivoli Information Management for z/OS program. This book gives instructions for designing, building, and testing TSPs and TSXs, followed by information on the different ways you can use TSPs and TSXs.

Tivoli Information Management for z/OS User's Guide, SC31-8756-00, provides a general introduction to Tivoli Information Management for z/OS and databases. This book has a series of step-by-step exercises to show beginning users how to copy, update, print, create, and delete records, and how to search a database. It also contains Tivoli Information Management for z/OS command syntax and descriptions and other reference information.

Tivoli Information Management for z/OS World Wide Web Interface Guide, SC31-8757-00, explains how to install and operate the features available with Tivoli Information Management for z/OS that enable you to access a Tivoli Information Management for z/OS database using a Web browser as a client.

Other related publications include the following:

Tivoli Decision Support: Using the Information Management Guide is an online book (in portable document format) that can be viewed with the Adobe Acrobat Reader. This book is provided with Tivoli Decision Support for Information Management (5697-IMG), which is a product that enables you to use Tivoli Information Management for z/OS data with Tivoli Decision Support. This book describes the views and reports provided with the Information Management Guide.

IBM Redbooks™ published by IBM's International Technical Support Organization are also available. For a list of redbooks related to Tivoli Information Management for z/OS and access to online redbooks, visit Web site **http://www.redbooks.ibm.com** or **http://www.support.tivoli.com**

Index

File Number: S370/30xx/4300
Program Number: 5697-SD9

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC31-8753-00