

Publication Number
SH20-6602-02

**IMS Application Development
Facility II
Version 2 Release 2**

**Interactive ADF
Administration Guide**



IMS Application Development
Facility II
Version 2 Release 2

Interactive ADF
Administration Guide

Publication Number
SH20-6602-02

File Number
S/370/4300-32

Program Number
5665-348

Third Edition (December 1987)

This edition applies to Version 2, Release 2 of the program product IMS Application Development Facility II (5665-348), and to all subsequent releases and modifications unless otherwise indicated in new editions or Technical Newsletters.

This edition updates information in the following sections for enhanced migration functions and the implementation of multiple versions SYSIDS in IMSADF II:

- Chapter 5, "Installing an Application SYSID"
- Chapter 6, "Adding Developers to the SYSID"
- Chapter 9, "Migration Services"
- "Converting Tables to Versioning" on page A-6
- Appendix C, "Migration Phase II Sample JCL"
- Appendix D, "Sample Migration User Exit"

All revised material in this publication is shown with a vertical bar to the left of the page.

Information in this publication is subject to change. Changes will be published in new editions or technical newsletters. Before using this publication, consult either your *IBM System/370 and 4300 Processors Bibliography* (GC20-0001) or *IBM System/370 and 4300 Processors Bibliography of Industry Systems Systems and Application Programs* (GC20-0370) to learn which editions and technical newsletters are current and applicable.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available outside the United States. Any reference to an IBM program product in this document is not intended to state or imply that only IBM's program products may be used. Any functionally equivalent program products may be used instead.

Requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for readers' comments has been provided at the back of this publication. If this form has been removed, address comments to:

IBM Corporation
Information Processing
Department 6DD
220 Las Colinas Blvd.
Irving, Texas 75039-5513

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

Preface

This manual is a guide to administering the Interactive Application Development Facility (IADF) for IMSADF II Version 2 Release 2. It shows the IMSADF II administrator, TSO/ISPF systems programmers, and/or the IADF installer how to set up the environment for later application development. Those individuals are the intended audience for this guide. The reader should have experience in the area of TSO/ISPF. Knowledge of IMSADF II JCL procedures and terminology is necessary. Installation information is required, so the IMSADF II installer should be consulted.

This manual consists of eleven chapters and four appendixes.

- **Chapter 1, “IADF Administrative Overview”** introduces IADF and describes the organization of this guide.
- **Chapter 2, “Preparing TSO/ISPF to Use IADF”** details, for systems programmers, the interfaces between IADF and both TSO and ISPF.
- **Chapter 3, “Getting Started”** describes the panel formats for IADF and shows you how to obtain the IADF main menu and tutorial. It defines the levels of authority for IADF.
- **Chapter 4, “Add ADFID Administrators”** shows how you add administrative users to IADF.
- **Chapter 5, “Installing an Application SYSID”** describes the panels involved in defining a SYSID to &ia, and allocating the libraries to be used by developers.
- **Chapter 6, “Adding Developers to the SYSID”** shows you how to add developers to the SYSID and the access levels appropriate to each.
- **Chapter 7, “Establish SYSID Data Set Activities”** shows how to initialize or update the lists of data sets maintained by SYSID in ISPF tables for the Rules Generator, HLAL, or BTS execution.
- **Chapter 8, “Initialize SYSID Data Bases”** shows the panels used to allocate and initialize the data bases used by the developers of a SYSID.
- **Chapter 9, “Migration Services”** describes activities relating to movement of SYSID components from test mode to production libraries.
- **Chapter 10, “Additional Administration Services”** discusses services not required for development, but useful at any time. They include glossary and table maintenance, updating of your own IADF profile, and the ability to browse installation options.
- **Chapter 11, “IADF Structure and Flow”** describes functionally the structure of IADF. It lists primary clists and includes a description of them.
- The appendixes are:
 - **“Converting IADF Tables from Prior IMSADF II Releases”** which gives instructions for conversion of IADF ISPF tables developed under a prior release of IMSADF II.
 - **Appendix B, “IADF Naming Conventions”** which gives you the naming conventions of the ISPF tables built and maintained by IADF.

Appendix C, “Migration Phase II Sample JCL” shows a sample of the migration job stream.

Appendix D, “Sample Migration User Exit” lists a model migration exit supplied with IMSADF II.

Prerequisite Publications

- *IMS Application Development Facility II Version 2 Release 2 General Information*, GH20-6591, contains information about Version 2 Release 2 functions, and programming and hardware requirements.

Corequisite Publications

- *IMS Application Development Facility II Version 2 Release 2 Installation Guide*, SH20-6593 contains information necessary to install IMSADF II.
- *IMS Application Development Facility II Version 2 Release 2 Application Development Reference*, SH20-6594.

Related Publications

- *ISPF Version 2 Installation and Customization*, SC34-2143.
- *ISPF Version 2 Program Reference*, SC34-2139.
- *ISPF Version 2 Library Management*, SC34-2135.

IMSADF II Publications

- *IMS Application Development Facility II Version 2 Release 2 General Information*, GH20-6591.
- *IMS Application Development Facility II Version 2 Release 2 User Reference*, SH20-6592.
- *IMS Application Development Facility II Version 2 Release 2 Installation Guide*, SH20-6593.
- *IMS Application Development Facility II Version 2 Release 2 Application Development Reference*, SH20-6594.
- *IMS Application Development Facility II Version 2 Release 2 Application Development Guide*, SH20-6595.
- *IMS Application Development Facility II Version 2 Release 2 Rules Documentation User's Guide*, SH20-6596.
- *IMS Application Development Facility II Version 2 Release 2 Data Dictionary Extension User's Guide*, SH20-6597.
- *IMS Application Development Facility II Version 2 Release 2 Master Index*, SH20-6599.
- *IMS Application Development Facility II Version 2 Release 2 Introduction to Using the Interactive ADF*, SH20-6601.
- *IMS Application Development Facility II Version 2 Release 2 Interactive ADF Administration Guide*, SH20-6602.
- *IMS Application Development Facility II Version 2 Release 2 ~~DATABASE~~ 2 Application Specification Guide*, SH20-6603.
- *IMS Application Development Facility II Version 2 Release 2 Diagnosis Guide*, LY20-6401.

Contents

Chapter 1. IADF Administrative Overview	1-1
IMSADF II Product Description	1-2
ISPF/PDF Product Requirements	1-3
ISPF/PDF and IADF Relationship	1-3
Chapter 2. Preparing TSO/ISPF to Use IADF	2-1
TSO Clists	2-1
Preparing Your System for IADF	2-1
Modifying IADF/IAD2	2-3
Modifying the ISPF Primary Option Menu	2-3
TSO Data Set Allocation Levels	2-7
Models	2-7
Modifiable Skeletons and Clists	2-10
Migration Requirements	2-12
IEBCOPY	2-12
LMF Support	2-12
TSO Performance	2-12
Multiple ADFID Considerations	2-14
Chapter 3. Getting Started	3-1
Preparation for Application Development	3-1
Required Data	3-2
Levels in IADF	3-2
Bringing up Administrator Screens	3-3
The IADF Main Menu	3-7
Help Facility	3-8
Tutorial	3-9
IADF Main Menu (Advanced User)	3-11
IADF Administration Main Menu	3-12
Chapter 4. Add ADFID Administrators	4-1
IADF Administrators of this ADFID	4-1
IADF Panel Format	4-5
Multiple ADFID Environment	4-5
Chapter 5. Installing an Application SYSID	5-1
Multiple Version SYSIDS	5-1
Add the Application System	5-2
Name Data Sets for this SYSID	5-5
Allocate SYSID Libraries	5-11
Table Initialization	5-13
Deleting a SYSID	5-14
SYSID/Migration Path Maintenance	5-16
Chapter 6. Adding Developers to the SYSID	6-1
Create IADF Developers Within IMSADF II System Table	6-1
Migration Authority	6-2
Chapter 7. Establish SYSID Data Set Activities	7-1
Introduction	7-1
Tables Used in IMSADF II Program Invocation	7-1
Selection	7-3

Panel Formats for Data Set Activities	7-3
Command Usage	7-4
Establish Foreground Static data sets for the SYSID	7-7
Establish Foreground Dynamic Data sets for the SYSID	7-11
Establish BTS Data sets for the SYSID	7-14
Establish Optional Batch Tables	7-17
Static Data Sets for Batch	7-18
Dynamic Data Sets for Batch	7-20
Copy Function	7-22
Summary	7-24
A Technique for Administering Multiple SYSIDS	7-24
Example	7-25
 Chapter 8. Initialize SYSID Data Bases	8-1
Allocate Data Bases	8-1
Initialize Data Bases	8-5
 Chapter 9. Migration Services	9-1
Migration Concepts	9-1
Migration Function in IADF	9-2
Two-Phase Migration	9-2
Source/Load Migration	9-2
Centralized Log	9-3
User Exit	9-3
Checkpointing Capability	9-3
Migration Authority	9-4
Multiple Version SYSIDS	9-5
"ISPF Tables" Considerations	9-6
Library Management Facility	9-6
Format of the Control File	9-6
Migration User Exit	9-9
Communication with the User Exit	9-10
Migration Examples	9-12
Element Selection	9-13
Migration Menu Selection	9-14
Restart Panel	9-15
SYSID (No Versions)	9-17
Multiple Path Selection	9-18
Migration Function Menu	9-19
Checkpoint Panel	9-21
System Elements	9-22
Segment Element	9-23
Transaction Element	9-25
Transaction Drivers Element	9-27
Audit Group Element	9-29
Update Message Data Base	9-31
Help Panels	9-32
Messages	9-33
Alternate Routes	9-34
Secondary Destinations	9-35
Non-Standard Element	9-36
Phase I Completion	9-38
Migration Log	9-40
 Chapter 10. Additional Administration Services	10-1
Establish ISPF User Profile for IADF	10-1

Extract ISPF Tables from Rules Source	10-6
Dynamic Rules Extraction	10-9
Static Rules Extraction	10-16
Additional Tasks to Complete	10-19
Screen Image Conversion	10-21
MFS Generation	10-23
Transaction Control Tables	10-23
Glossary Services	10-28
Table Management	10-31
 Chapter 11. IADF Structure and Flow	 11-1
OPTION 1: Definition	11-2
Option 1.1: Data Layouts	11-2
Option 1.2: Panels	11-2
Option 1.3: Transactions	11-2
Option 1.4: Audits	11-3
Option 1.5: Messages	11-3
Option 1.6: IMSADF II Users	11-3
Option 1.7: IMSADF II Profiles	11-3
Option 1.8: Transaction Drivers	11-4
Option 1.9: Applications (SYSIDs)	11-4
Option 1.10: Routes	11-4
Option 1.11: Secondary Destinations	11-4
OPTION 2: Rules Generation	11-5
Option 2.1: Static Rules Generation	11-5
Option 2.2: Dynamic Rules Generation	11-5
Option 2.3: Audit Rules Generation	11-5
OPTION 3: Test	11-6
Option 3.1: PROTOTYPING	11-6
Option 3.1.1: Screen Image Creation	11-6
Option 3.1.2: Screen Image Translation	11-6
Option 3.1.3: Data Base	11-6
Option 3.1.4: Transaction	11-6
Option 3.1.5: Dialog Creation	11-6
Option 3.1.6: Dialog Test	11-6
Option 3.1.7: Prototype Control	11-7
Option 3.2: BTS	11-7
OPTION 4: IADF Utilities	11-8
Option 4.1: Extract	11-8
Option 4.2: SUBMIT Utility	11-8
Option 4.3: Glossary Function	11-8
Option 4.4: Rules Documentation	11-9
Option 4.5: Library Display	11-9
Option 4.6: IMSADF II Trace	11-9
Option 4.7: Global Find and Edit	11-9
Option 4.8: MFS Utility	11-9
Option 4.9: Data Dictionary Utilities	11-9
Option 4.10: DB2 RGLGEN process	11-10
OPTION 5: IADF User Profile	11-10
OPTION 6: Tutorial Function	11-10
IADF Administration Functions	11-11
OPTION 1: Systems	11-11
Option 1.1: Libraries	11-11
Option 1.2: Developers	11-12
Option 1.3: Static DSNS	11-12
Option 1.4: Dynamic DSNS	11-13

Option 1.5: BTS DSNS	11-13
Option 1.6: Initialization	11-13
OPTION 2: ADMINISTRATORS	11-13
OPTION 3: MIGRATION	11-13
OPTION 4: TABLES	11-14
OPTION 5: GLOSSARY	11-15

Appendix A. Converting IADF Tables	A-1
Converting IADF Tables from Prior IMSADF II Releases	A-1
Converting Tables to Versioning	A-6

Appendix B. IADF Naming Conventions	B-1
Distributed Tables in ADFTLIB	B-1
Table Naming Conventions	B-2
Rule Source Naming Conventions	B-4

Appendix C. Migration Phase II Sample JCL	C-1
--	-----

Appendix D. Sample Migration User Exit	D-1
---	-----

Index	X-1
--------------	-----

Figures

1-1.	Relationship Between TSO, ISPF, and IADF	1-3
1-2.	IADF Interface to Other IBM Program Products	1-4
2-1.	ISPF Primary Option Menu Edited to Include IADF	2-4
2-2.	ISPF Tutorial Edited to Access IADF Tutorial Panels	2-6
2-3.	Model Class Panel (ISREMCLS) Modified for IADF	2-8
3-1.	Installation Data Required for Administering IADF	3-2
3-2.	IADF Main Menu Panel	3-6
3-3.	IADF Main Menu (with Panel ID displayed)	3-7
3-4.	HELP Panel for IADF Main Menu	3-8
3-5.	IADF Main Menu (with Panel ID displayed)	3-9
3-6.	First Page of IADF Tutorial	3-10
3-7.	IADF Main Menu (Advanced User)	3-11
3-8.	IADF Administration Main Menu	3-12
4-1.	Selecting Administrators from IADF Administration Main Menu	4-2
4-2.	First Level 3 Administrator	4-2
4-3.	Inserting an Administrator Example	4-3
4-4.	Administrator Added	4-4
4-5.	IADF Administration Main Menu	4-4
5-1.	Select SYSTEMS from the IADF Administration Main Menu	5-3
5-2.	IADF System Definition Menu	5-3
5-3.	Display Existing SYSIDS for an ADFID	5-4
5-4.	SYSID Installation Example	5-5
5-5.	SYSID Version Warning Panel	5-7
5-6.	Migration Paths	5-8
5-7.	Display SYSID within ADFID	5-9
5-8.	Allocation Error Information Panel	5-10
5-9.	Library Allocation Panel	5-11
5-10.	Successful SYSID Libraries Allocation	5-12
5-11.	Redisplay IADF System Definition Menu with Completion Message	5-13
5-12.	SYSID Deletion Panel	5-14
5-13.	SYSID Deletion Panel	5-15
5-14.	Migration Paths Maintenance	5-16
5-15.	Version Mappings Maintenance Utility	5-17
6-1.	Select SYSTEMS from the IADF Administration Main Menu	6-3
6-2.	Selecting DEVELOPERS Option from IADF System Definition Menu	6-4
6-3.	Developers within SYSID Panel	6-5
6-4.	Adding a Developer of a SYSID	6-5
6-5.	Developers within SYSID Panel	6-6
7-1.	Installation Options and SYSID Tables	7-2
7-2.	Select SYSTEMS from the IADF Administration Main Menu	7-3
7-3.	Data Set Allocation	7-4
7-4.	Updating PARM Example	7-5
7-5.	Selecting STATIC DSNS Option from IADF System Definition Menu	7-7
7-6.	Foreground Static Rules Data Sets	7-8
7-7.	Successful Table Update	7-11
7-8.	Selecting DYNAMIC DSNS Option from IADF System Definition Menu	7-12
7-9.	Foreground Dynamic Data Set Panel	7-13
7-10.	Selecting BTS DSNS Option from IADF System Definition Menu	7-15
7-11.	BTS Data Sets	7-16
7-12.	Batch Static Rules Data Sets	7-18
7-13.	Batch Dynamic Data Sets	7-21

7-14.	Copying From "master" SYSID Table	7-25
8-1.	Select SYSTEMS to Initialize Data Bases	8-2
8-2.	Selecting INITIALIZE Option from IADF System Definition Menu	8-2
8-3.	Initialize Libraries/Data Bases Panel	8-3
8-4.	Allocate Data Bases	8-3
8-5.	Successful Data Base Allocation	8-4
8-6.	Data Base Initialization	8-5
8-7.	Foreground Data Base Initialization	8-6
8-8.	Foreground Data Base Initialization (Panel 2)	8-6
8-9.	Data Base Initialization	8-7
8-10.	Job Submission Panel	8-8
9-1.	IADF Main Menu (Advanced User)	9-14
9-2.	Checkpoint Information for Restarting	9-16
9-3.	SYSID without Versioning	9-17
9-4.	Multiple Targets (Paths) for A SYSID/VERSION	9-18
9-5.	IADF Main MIGRATION Panel	9-19
9-6.	Checkpoint Information	9-21
9-7.	System Copy Panel	9-22
9-8.	Segment Main Panel	9-23
9-9.	Segment Selection Panel	9-24
9-10.	Transaction Main Panel	9-25
9-11.	Individual Transaction Selection Panel	9-26
9-12.	Transaction Driver Main Panel	9-27
9-13.	Driver Selection Copy Panel	9-28
9-14.	Audit Group Selection Copy Panel	9-29
9-15.	Transaction Audit Selection Copy Panel	9-30
9-16.	Message Data Base Main Panel	9-31
9-17.	Help Panel Migration	9-32
9-18.	Messages Migration	9-33
9-19.	Alternate Routes Migration	9-34
9-20.	Secondary Destinations Migration	9-35
9-21.	Designate Output Library for Non-Standard Elements	9-36
9-22.	Input Non-Standard Element Library	9-37
9-23.	Selection List for Non-Standard Element COPY	9-37
9-24.	Job Submission Panel	9-38
9-25.	Sample Phase II Log	9-40
10-1.	Select DEVELOPMENT	10-2
10-2.	IADF Main Menu	10-2
10-3.	First Page of User Profile Panel	10-3
10-4.	Second Page of User Profile Panel	10-4
10-5.	Successful Product Overrides for SYSID	10-6
10-6.	Select DEVELOPMENT	10-7
10-7.	IADF Main Menu	10-7
10-8.	IADF Utilities Menu	10-8
10-9.	Extraction Utilities Panel	10-9
10-10.	Dynamic Rules Extraction Panel	10-10
10-11.	Dynamic Rules Extraction Summary Panel	10-11
10-12.	Using the ERR command to display the Extract Error Report	10-12
10-13.	Browsing the Dynamic Rules Extraction Error Report	10-12
10-14.	Selecting a Message to be Extracted	10-13
10-15.	Confirmation message for Dynamic Rules Extract	10-14
10-16.	Successful Dynamic Rules Extraction	10-15
10-17.	Successful Extraction	10-16
10-18.	Load ISPF Tables from Rules Generator Input	10-17
10-19.	Static Rules Generation in Foreground	10-18
10-20.	Successful Static Rules Extraction	10-19

10-21.	IADF Main Menu	10-20
10-22.	IADF Definition Menu	10-20
10-23.	Signon/Primary Option Menu definition	10-21
10-24.	Model Screen Image Signon Panel in Edit Mode	10-22
10-25.	IADF Main Menu	10-23
10-26.	IADF Main Menu	10-24
10-27.	IADF Definition Menu	10-24
10-28.	Transactions for a SYSID	10-25
10-29.	Transactions for a SYSID	10-25
10-30.	Control Information for Segments in Transaction	10-26
10-31.	Control Information for Fields in Transaction	10-27
10-32.	Transactions for a SYSID	10-28
10-33.	Select GLOSSARY	10-29
10-34.	Glossary Menu	10-29
10-35.	Adding a Term to the Glossary	10-30
10-36.	Adding a Term to the Glossary	10-30
10-37.	Select TABLES	10-31
10-38.	Table Management Menu	10-32
10-39.	Table Create	10-32
10-40.	Alter Columns in a Table	10-33
10-41.	Table Copy Function	10-34
A-1.	Invoke Table Conversion Process	A-2
A-2.	IADF Table Conversion Panel	A-3
A-3.	IADF SYSID Selection Panel	A-4
A-4.	New "ISPF Tables" Data Set panel	A-5
A-5.	IADF Table Conversion Report	A-6
A-6.	Invoke Versioning Conversion Clist	A-7
A-7.	Convert the SYSTEMS Table	A-8
A-8.	Convert Developers Table	A-9
B-1.	IMSADF II Master Table in ADFTLIB	B-1
B-2.	Naming Conventions for SYSID Tables	B-2
B-3.	Naming Conventions for Rule Source	B-4
C-1.	Migration Phase II Sample JCL	C-2
D-1.	Model for a Migration User Exit	D-2

Chapter 1. IADF Administrative Overview

This book tells you how to administer the IADF environment for the application developers who use IADF. In addition to showing you how to prepare TSO and ISPF for the IADF dialogs, this guide takes you step-by-step through the administration process.

This guide is intended for persons who are TSO/ISPF systems programmers, IMSADF II administrators, or installers. You should be an experienced ISPF user or systems programmer, and you should be familiar with IMSADF II terminology and JCL procedures. The bulk of technical information describing the IADF administrator panels is found in the HELP panels.

You must prepare your environment to invoke the IADF dialogs as described in Chapter 2, “Preparing TSO/ISPF to Use IADF.” This chapter is specifically for systems programmers.

Chapter 3, “Getting Started” tells you how to invoke the dialogs.

There is a chapter devoted to each task you perform as an administrator and each sequence of panels is presented for that activity. IADF provides a series of administrator panels to aid you in the following:

- Chapter 4, “Add ADFID Administrators”
- Chapter 5, “Installing an Application SYSID”
- Chapter 6, “Adding Developers to the SYSID”
- Chapter 7, “Establish SYSID Data Set Activities”
- Chapter 8, “Initialize SYSID Data Bases”

At the beginning of each chapter a list is presented with the step which is the chapter topic highlighted, so that you may see approximately where you are in the administration process. At the completion of them, you will have finished all the tasks required; and your developers may begin to use IADF.

Moving a completed SYSID from test to production mode is covered in Chapter 9, “Migration Services.” When you move the rules that you developed with IADF to your production environment, these services insure that everything required is included.

Chapter 10, “Additional Administration Services” is provided for auxiliary functions you may use at any time:

- “Establish ISPF User Profile for IADF”
Users may create their own ISPF profiles for IADF.
- “Extract ISPF Tables from Rules Source”
Users may convert existing Rules Generator source to IADF table format.
- “Table Management”
You may need to manipulate IADF tables.
- “Glossary Services”

You may add your own terms to the supplied glossary for benefit of the application developers at your location.

If you wish to know more about the organization of IADF and how processing is accomplished, Chapter 11, "IADF Structure and Flow" is provided.

If you have existing IADF development under a previous release of IMSADF II, the ISPF tables must be converted to the table format appropriate for Version 2 Release 2. "Converting IADF Tables from Prior IMSADF II Releases" gives step-by-step instructions you must follow.

You may find Appendix B, "IADF Naming Conventions," helpful, as it lists each table created and maintained by IADF.

It is a good idea to "install" the supplied IMSADF II application, SAMP, using this guide for your own experience. In addition, it must be available in order for your developers to execute the sample described in *IMS Application Development Facility II Version 2 Release 2 Introduction to Using the Interactive ADF*.

Note: You must authorize each of your application developers to this test system before they can use it.

As you progress through this manual, you will find that many of the examples use SAMP.

IMSADF II Product Description

The IMS Application Development Facility II (IMSADF II) is a general purpose application programming system. It is used to develop and maintain IMS/VS conversational, non-conversational, and batch applications, as well as CICS/OS/VS and DB2 applications.

IMSADF II runs as an application program under IMS/VS, CICS/OS/VS, or DB2. It provides a general purpose application execution program which encompasses functions common to a vast number of user applications. Examples of such functions are: end-user sign-on, data retrieval, data display, data update, and message transmission.

Data processing personnel of installations using IMSADF II customize the general purpose system by the specification of "rules," which define an application. Rules are stored in data bases or libraries, separate from the common function modules. Rules may be accessed by multiple user applications, reducing redundancy in programming efforts.

Generalized transaction drivers support all three IMS/VS processing modes: conversational, nonconversational, and batch, as well as the CICS/OS/VS and DB2 environments. At execution time, control is passed to a transaction driver. The driver references rules that direct how common function modules execute the user application system. The binding of all application elements -- input/output, data bases, transaction driver, common function modules, and rules -- occurs at execution time.

A significant programmer productivity tool, IMSADF II replaces conventional programming efforts in many instances.

ISPF/PDF Product Requirements

The Interactive System Productivity Facility (ISPF) Version 2, Product Number 5665-319, and the ISPF/Program Development Facility (ISPF/PDF or PDF) Version 2, Product Number 5665-317, are related IBM products. The Version 2 level of both products is required for IADF use.

- ISPF is a dialog manager for interactive applications. It provides control and services for processing of dialogs.
- PDF is a facility that aids in the development of various types of applications, including dialogs. It uses display terminals in an interactive environment to assist with many programming tasks.

ISPF is the base product for PDF and for other interactive applications, or dialogs. PDF is itself a dialog and runs with the control and services of ISPF.

The Library Management Facility (LMF) services of ISPF are used in execution of the IADF dialogs, but the LMF User SVC is required only if you use the PROMOTE function, selectable on an IADF Migration facility panel.

ISPF/PDF and IADF Relationship

The Interactive Application Development Facility (IADF) simplifies the development of IMSADF II applications. IADF consists of dialogs and programs that run under ISPF. IADF in turn invokes IMSADF II programs such as the Rules Generator, High Level Audit Language, and the Batch Driver.

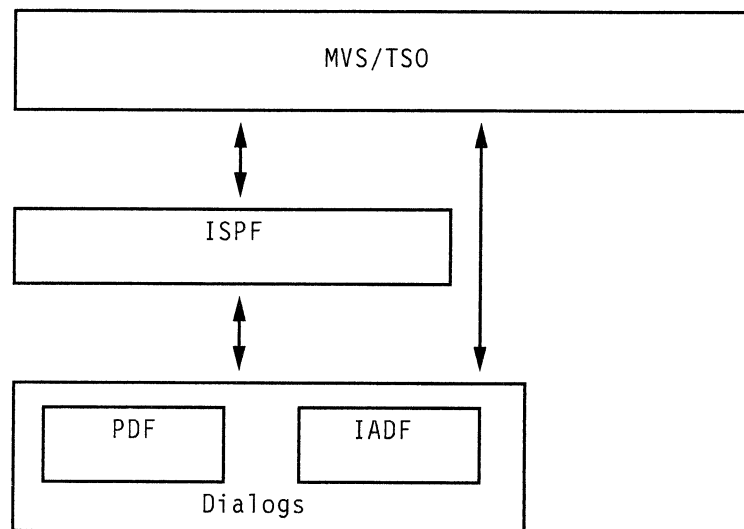


Figure 1-1. Relationship Between TSO, ISPF, and IADF

IADF provides the online user with all development tools that the basic IMSADF II product contains for the batch user. In addition, IADF has several features not available in batch. Examples of these features are:

- Extraction of data from DBDs to facilitate rules generation
- Screen prototyping
- Dialog simulation
- Online utilities
- Test to Production migration
- BTS execution of applications (if you are an IMS/VS user and you have this product)

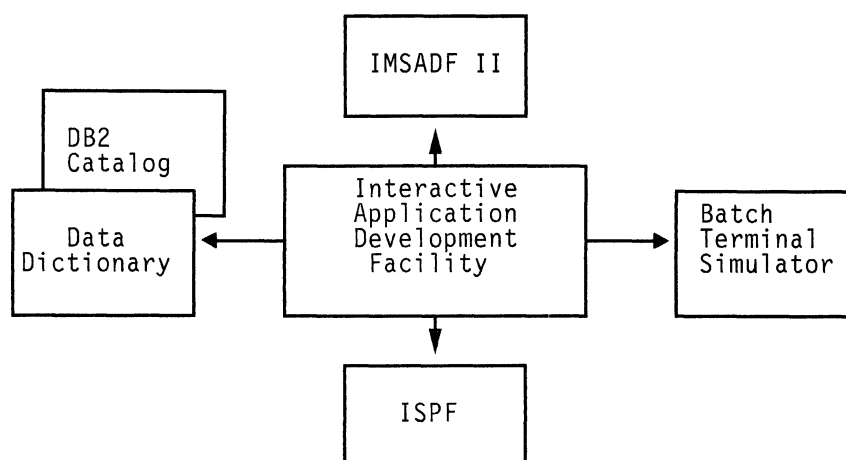


Figure 1-2. IADF Interface to Other IBM Program Products

IADF has interfaces to IMSADF II, BTS, ISPF, and the Data Dictionary and DB2 Catalog, making it the focal point for the development of IMSADF II applications.

Chapter 2. Preparing TSO/ISPF to Use IADF

This chapter is intended primarily for the ISPF and/or TSO systems programmer at your installation. It tells you how IADF gets invoked and how you can modify its TSO clists for your environment.

TSO Clists

One of two IMSADF II-supplied approaches may be used to start IADF.

- From outside ISPF:

1. Log on to TSO.

2. Enter Clist:

- IADF

Allocates libraries, panels, skeletons, system tables, messages, and programs.

- IAD2

Does no allocations. Assumes previous IADF entry, within TSO session.

Both invoke the primary IADF clist IMSADFM.

This method is described in “Modifying IADF/IAD2” on page 2-3.

- From an ISPF menu:

The ISPF menu is modified to contain an option that calls the IMSADFM clist. Select that option.

See “Modifying the ISPF Primary Option Menu” on page 2-3 for a description of this method.

If the installed version of IADF is other than the default of MFC1, all three clists require an additional parameter of ADFID(adfid), where ADFID is a keyword and adfid is replaced by the id of the installed version. For example, if ADF2 is the installed IADF version and IADF is the clist to be called, the following is entered:

```
IADF 'ADFID(ADF2)'
```

If you modify the ISPF menu, the ADFID value is modified on the panel itself.

Preparing Your System for IADF

You have properly installed the IMSADF II product using the *IMS Application Development Facility II Version 2 Release 2 Installation Guide*.

The ISPF libraries that must be allocated are shown in the following list, which is valid regardless of the method used to invoke IADF. The high level node is ADFNODE, since they may be shared across a release of IMSADF II. Any special remarks are noted in this list.

ISPPLIB	IMSADF.ADFPLIB - Panel input libraries Use of the PROTOTYPING facility of IADF development results in the creation of ISPF panels from IMSADF II Rules Generator source. If the panel library containing the new panels from this function are not concatenated to this DDNAME, users will not be able to view them. You should allocate a separate panel library which can be installation-wide for this purpose (use the same block size as ADFPLIB).
ISPSLIB	IMSADF.ADFSLIB - Skeleton input for file tailoring
ISPMLIB	IMSADF.ADFMLIB - Message input libraries
ISPTLIB	IMSADF.ADFTLIB - Table input libraries
ISPFIL	Allocated as temporary. Do not use Virtual I/O (VIO). Note: Related to ISPFIL are the DD statements ISPCTL0 and ISPCTL1, which may be allocated in a TSO logon procedure and are used as temporary sequential work files by ISPF. If these are allocated, ISPFIL, which is partitioned, is ignored. These DD statements must not be allocated by ISPF if IADF is to use ISPFIL correctly. You can find out if they are allocated by using ISPF Dialog Manager Services Option 7.3 to look at your variables. The variables ZTEMPF and ZTEMPN cannot be VIO.
ISPLLIB	Program load libraries IMSADF.ADFLOAD must be placed here. If you use multiple ADFIDs and do not share ADFLOAD, concatenate additional ADFLOAD libraries here.
ISPPROF	Profile data set This data set must have DISP=SHR in order for IADF Phase II migration to run. (see “Migration Concepts” on page 9-1 and Appendix D, “Sample Migration User Exit”).
SYSPROC	IMSADF.ADFCLIB - TSO Clist libraries

The IMSADF II DDNAME that must be allocated and the IMSADF II library name is shown below. This target library's high level node is ADFNODE, since tables may be shared across ADFIDs within a release.

ADFTLIB IMSADF.ADFTLIB - Table input library

Before you execute IADF, change the nodes of the data sets to the same ones you chose in installation job IMSADF.INSTALL(ADFAALOC).

All tables named in the following chapters reside in this data set unless otherwise specified.

Place the rest of the libraries you use in your installation in the concatenation of the **ISP- ddnames**.

Modifying IADF/IAD2

When you are ready to invoke IADF, you must execute the clist named IADF, supplied in IMSADF.ADFCLIB. This concatenates the IMSADF II ISPF libraries listed in “Preparing Your System for IADF” in front of your existing ones, so that IMSADF II superimposes its panels on top of the ISPF panels. The first panel displayed is not the ISPF Primary Option panel. If you are an Administrator in IADF, the first panel you see is the IADF Administration Main Menu panel (IADFADMN). If you are not an Administrator in IADF, the first panel you see is the IADF Main Menu panel (IADFMAIN).

Note: If you wish to invoke normal ISPF functions from the IMSADF II panels, enter SPF, PDF, or ISPF. When you do this, you see the normal ISPF panels appear.

You may invoke IADF or IAD2 in ISPF TEST mode by adding that parameter to the invocation as shown:

```
IADF 'ADFID(ADF2)' TEST
```

Modifying the ISPF Primary Option Menu

Add the libraries named in “Preparing Your System for IADF” to your ISPF invocation procedure.

You may choose to modify the supplied ISPF panels so that IADF is an option. The suggested panels for making the connection between ISPF and the IADF panels are:

- Primary Menu(ISR@PRIM)
- Tutorial panel(ISR00003)

Note: See your ISPF systems programmer for other possibilities.

If your installation has made changes to either of these panels, you should change your existing ones, rather than use the examples given here. The panels described in Figure 2-1 on page 2-4 and Figure 2-2 on page 2-6 show the supplied ISPF panels with the needed modifications highlighted.

Refer to Figure 2-1.

```

%----- ISPF/PDF PRIMARY OPTION MENU -----
%OPTION ==>_ZCMD
%
%                                +USERID  - &ZUSER
% 0 +ISPF PARMS  - Specify terminal and user parameters +TIME    - &ZTIME
% 1 +BROWSE      - Display source data or output listings +TERMINAL - &ZTERM
% 2 +EDIT        - Create or change source data          +PF KEYS  - &ZKEYS
% 3 +UTILITIES   - Perform utility functions
% 4 +FOREGROUND  - Invoke language processors in foreground
% 5 +BATCH       - Submit job for language processing
% 6 +COMMAND     - Enter TSO command or CLIST
% 7 +DIALOG TEST - Perform dialog testing
% 8 +LM UTILITIES- Perform library management utility functions
% 9 +IADF        - Interactive Application Development Facility
% C +CHANGES    - Display summary of changes for this release
% T +TUTORIAL    - Display information about ISPF/PDF
% X +EXIT        - Terminate ISPF using log and list defaults
%
+Enter%END+command to terminate ISPF.
%
)INIT
  .HELP = ISR00003
  &ZPRIM = YES      /* ALWAYS A PRIMARY OPTION MENU      */
  &ZHTOP = ISR00003 /* TUTORIAL TABLE OF CONTENTS                          */
  &ZHINDEX = ISR91000 /* TUTORIAL INDEX - 1ST PAGE                            */
  VPUT (ZHTOP,ZHINDEX) PROFILE
)PROC
  &ZSEL = TRANS( TRUNC (&ZCMD, '.')
    0, 'PANEL(ISPOPTA)'
    1, 'PGM(ISRBRO) PARM(ISRBRO01)'
    2, 'PGM(ISREDIT) PARM(P,ISREDM01)'
    3, 'PANEL(ISRUTIL)'
    4, 'PANEL(ISRFPA)'
    5, 'PGM(ISRJB1) PARM(ISRJPA) NOCHECK'
    6, 'PGM(ISRPTC)'
    7, 'PGM(ISRYXDR) NOCHECK'
    8, 'PANEL(ISRLPRIM)'
    9, 'CMD(%IMSADFM ADFID(MFC1)) NEWAPPL(ADF1)'
    C, 'PGM(ISPTUTOR) PARM(ISR00005)'
    T, 'PGM(ISPTUTOR) PARM(ISR00000)'
    ' ', ' '
    X, 'EXIT'
    *, '?' )

  &ZTRAIL = .TRAIL
)END

```

Figure 2-1. ISPF Primary Option Menu Edited to Include IADF

The previous figure shows the ISPF/PDF primary option menu (ISR@PRIM) supplied by ISPF. This panel is tailored by installation, and so may appear different at your location. However, major functions available would be the same. You see that all but two lines remain just as they are supplied. The two added lines include one displayed line:

```
%    9  +IADF          - Interactive Application Development Facility
```

and one non-displayed line:

```
          9,'CMD(%IMSADFM ADFID(MFC1)) NEWAPPL(ADF1)'
```

The displayed line shows that, if a user types 9 on the option line, IADF is invoked. If the user is an Administrator in IADF, the IADF Administration Main Menu is displayed. If the user is a Developer in IADF, the IADF Main Menu is shown next. The non-displayed line provides what is needed to invoke the IADF functions. IMSADFM is the main invoking clist. In the example, you are expected to replace "MFC1" with the four-character ADFID you use if it is different.

If you have multiple ADFIDs, you may add another option line identical to the current one, changing only ADFID. (Remember to add those libraries to your ISPF invocation procedure).

Figure 2-2 on page 2-6 is the ISPF/PDF primary option menu tutorial (ISR00003). This panel is tailored by installation, and so may appear different at your location. However, major functions available would be the same.

```

%TUTORIAL ----- TABLE OF CONTENTS ----- TUTORIAL
%OPTION ==>_ZCMD
+
%
      -----
      | ISPF PROGRAM DEVELOPMENT FACILITY TUTORIAL |
      -----
+
  The following topics are presented in sequence, or may be selected by
  entering a one-character selection code in the option field:
    %G+ GENERAL      - General information about ISPF
    %0+ ISPF PARMS   - Specify terminal and user parameters
    %1+ BROWSE       - Display source data or output listings
    %2+ EDIT         - Create or change source data
    %3+ UTILITIES    - Perform utility functions
    %4+ FOREGROUND   - Invoke language processors in foreground
    %5+ BATCH        - Submit job for language processing
    %6+ COMMAND      - Enter TSO command or CLIST
    %7+ DIALOG TEST  - Perform dialog testing
    %8+ LM UTILITY   - Perform library management functions
    %9+ IADF         - Interactive Application Development Facility
    %X+ EXIT         - Terminate ISPF using log and list defaults
  The following topics will be presented only if explicitly selected:
    %A+ APPENDIX A   - Dynamic allocation interface routine (DAIR) errors
    %B+ APPENDIX B   - ISPF listing formats
    %I+ INDEX        - Alphabetical index of tutorial topics
)PROC
  &ZSEL = TRANS(&ZCMD
              G,ISR01000
              0,ISP05000
              1,ISR10000
              2,ISR20000
              3,ISR30000
              4,ISR40000
              5,ISR50000
              6,ISR60010
              7,ISR70000
              8,ISR80000
              9,HAGD8010
              X,ISP90100
              A,*ISP93030
              B,*ISR95000
              I,*ISR91000
              )
)END

```

Figure 2-2. ISPF Tutorial Edited to Access IADF Tutorial Panels

Figure 2-2 shows the modifications that you must make in highlighted mode. Note that all but two lines remain just as they are supplied. The two added lines include one displayed line:

```

    %9+ IADF      - Interactive Application Development Facility

```

and one non-displayed line:

```

        9,HAGD8010

```

The displayed line shows that, if a user types 9 on the option line, the IADF Tutorial panel is shown next. The non-displayed line provides what is needed to invoke the IADF tutorial. HAGD8010 is the main panel.

TSO Data Set Allocation Levels

You must make sure enough data sets can be allocated to a TSO userid if you wish developers to execute IMSADF II services in foreground. The maximum number of **DD statements** which may be concurrently allocated is found in each TSO logon procedure. If you concatenate other data sets to the DDNAMES used in these services, you do not increase the number of DDNAMES, but you will increase the data set allocation time. The number increases, however, if you add DDNAMES.

From a TSO performance standpoint, you may wish to eliminate unnecessary data sets when you set up the tables required for the services. This is not so important for batch tables (see “Establish Optional Batch Tables” on page 7-17), since the services are not performed in foreground. Affected tables are found in “Establish Foreground Static data sets for the SYSID” on page 7-7 and “Establish Foreground Dynamic Data sets for the SYSID” on page 7-11.

The approximate minimum number of DDNAMES required for each IADF function is listed below.

RULEGEN	30 depending on DB2, Data Dictionary services
MFSUTL	15 in addition to RULEGEN for MFS screen generation.
BMS	10 in addition to RULEGEN for BMS screen generation.
Dynamic	40
BTS	30

There are three IADF commands which may be used to give assistance:

- FREEST** Frees static Rules Generator data sets
- FREEDY** Frees dynamic data sets
- FREEBTS** Frees BTS related data sets

For some functions of IADF (especially those invoking the IMS/VS batch region controller) you may need a TSO region size of 4096K. Otherwise 2048K should be adequate.

Models

IADF MODELS must be added to those supplied with ISPF. They are normally invoked by IADF during the development process without the programmer's being aware of it. They are useful if you are developing your own Rules Generator or audit input outside of IADF, but while you are in ISPF edit mode.

The panel supplied in ISP.ISPPLIB(ISREMCLS) must have the four IADF model classes added to it; however, the panel as supplied by ISPF does not have enough space to allow additions. IMSADF II has included in IMSADF.ADFPLIB a member, ADFEMCLS, that contains the ISPF model classes and also the IADF ones. You must take this panel, shown in Figure 2-3, copy it to your ISPF panel library, renaming it ISREMCLS (replacing the supplied panel). The panel shown has the IADF modifications highlighted.

```

)ATTR
  ¢ TYPE(OUTPUT) INTENS(HIGH)
)BODY CMD(ZCMD) MSG(ADFMSG) LMSG(ADFMSG) WIDTH(&ZSCREENW) EXPAND(##)
%#-# MODEL CLASSES #-#
%OPTION ==>_ZCMD
%
%
%
% 1 +CLIST      - ISPF services in CLIST commands
% 2 +COBOL      - ISPF services in COBOL programs
% 3 +EXEC       - ISPF services in EXEC commands
% 4 +FORTRAN    - ISPF services in FORTRAN programs
% 5 +MSGs       - Message format
% 6 +PANELS     - Panel formats and statements
% 7 +PLI        - ISPF services in PLI programs
% 8 +SKELS      - File tailoring control statements
% 9 +PASCAL     - ISPF services in PASCAL programs
% 10+SIMAGE     - IMSADF II Panel Formats and Field definitions
% 11+AUDITOR    - IMSADF II Auditor Services
% 12+RULES      - IMSADF II Static Rules
% 13+SPECIAL    - IMSADF II Special Process Services
%
%
+
+Enter%END+command to cancel MODEL command.+
¢ADFMSG
)INIT
  .CURSOR = ZCMD
  .HELP = ISR2P100
  .MSG = ' ' /* BLANK MSG INITIALLY */
  &ISRMCLAS = ' ' /* AND THE CLASS HOLDER VARIABLE */
)PROC
  IF (&ISRMONCL = 'Y') /* ARE WE ONLY OBTAINING A CLASS ? */
    &ISRMCLAS = TRANS(&ZCMD 1,CLIST 2,COB 3,EXEC 4,FORT 5,MSGs
                      6,PANELS 7,PLI 8,SKELS *,*)
  IF (&ISRMONCL = 'Y') /* ARE WE ONLY OBTAINING A CLASS ? */
    &ISRMDSPL = ' '
  &ZSEL = TRANS(TRUNC (&ZCMD, '.')) /* FOLLOW SELECTION TO NEXT PANEL */
    1, 'PANEL(ISREMCMD)'
    CLIST, 'PANEL(ISREMCMD)' /* CLIST UNDER COMMAND SERVICES */
    2, 'PANEL(ISREMCOB)'
    COB, 'PANEL(ISREMCOB)' /* COBOL PROGRAM SERVICE MODELS */
    COBOL, 'PANEL(ISREMCOB)'
    3, 'PANEL(ISREMEXC)'
    EXEC, 'PANEL(ISREMEXC)' /* EXEC FUNCTION SERVICES */
    EXEC2, 'PANEL(ISREMEXC)'

```

Figure 2-3 (Part 1 of 2). Model Class Panel (ISREMCLS) Modified for IADF

```

        4, 'PANEL(ISREMFTN)'
        FTN, 'PANEL(ISREMFTN)'          /* FORTRAN PROGRAM SERVICE MODELS */
        FORT, 'PANEL(ISREMFTN)'
        FORTRAN, 'PANEL(ISREMFTN)'
        5, 'PGM(ISRECMR) PARM(ISREMSG)' /* GO RIGHT OUT TO GET */
        MSGS, 'PGM(ISRECMR) PARM(ISREMSG)' /* THE MSG MODEL */
        6, 'PANEL(ISREMPNL)'
        MENUS, 'PANEL(ISREMPNL)'        /* OLD TYPE PANELS */
        PANELS, 'PANEL(ISREMPNL)'
        7, 'PANEL(ISREMPLI)'
        PLI, 'PANEL(ISREMPLI)'          /* PLI FOR TSO AND MVS (DEFAULT) */
        PL1, 'PANEL(ISREMPLI)'          /* PLI FOR TSO AND FOR DOS */
        PLIOPT, 'PANEL(ISREMPLI)'       /* PLIOPT DOES HAVE SOME USAGE YET */
        PLS, 'PANEL(ISREMPLI)'         /* PLS OPTION IS TEMPORARY FOR NOW */
        8, 'PANEL(ISREMSKL)'
        PROCS, 'PANEL(ISREMSKL)'
        SKELS, 'PANEL(ISREMSKL)'        /* FOR BOTH PROCS AND SKELS */
        9, 'PANEL(ISREMPAS)'
        PASCAL, 'PANEL(ISREMPAS)'       /* PASCAL PROGRAM SERVICE MODELS */
        SIMAGE, 'PANEL(ADFPN01)'
        10, 'PANEL(ADFPN01)'
        AUDITOR, 'PANEL(ADFAUDI)'
        11, 'PANEL(ADFAUDI)'
        RULES, 'PANEL(ADFRL01)'
        12, 'PANEL(ADFRL01)'
        SPECIAL, 'PANEL(ADFSP01)'
        13, 'PANEL(ADFSP01)'
        ' ', ' '
        *, '?' )
IF (&ZSEL = '?')
    IF (&ISRMDSPL ^= 'RETURN' )
        .MSG = ISRYM012
        .RESP = 'ENTER'                /* WANT TO RESET RESP FOR RE-ENT */
    IF (&ISRMDSPL = 'RETURN' )
        .RESP = 'END'
        &ISRMCLAS = '?'
    &ISRMEND = 'N'                    /* SET THE END INDICATOR TO NO */
    IF (.RESP = END)                  /* DID WE GET AN END FROM WHERE */
        IF (&ISRMDSPL ^= 'RETURN ' ) /* WAS IT BECAUSE WE HAVE MEMBER */
            &ISRMEND = 'Y'           /* ITS BECAUSE USER HIT END KEY */
        IF (&ISRMONCL = 'Y')         /* ARE WE ONLY OBTAINING A CLASS */
            IF (.RESP ^= 'RETURN ' ) /* HAS USER HIT EXTENDED RETURN */
                IF (&ISRMCLAS ^= ' ' ) /* MAKE SURE WE HAVE SOME CLASS */
                    .RESP = END      /* ONLY WANT CLASS, NO OTHER DISP. */
)END

```

Figure 2-3 (Part 2 of 2). Model Class Panel (ISREMCLS) Modified for IADF

Modifiable Skeletons and Clists

Skeletons: are provided in IMSADF.ADFSLIB for use in IADF. Many are provided for IMSADF II installation purposes and should not be modified.

Those in the following list may be modified in your own library. This should be done with great care and only if necessary. Several of these skeletons contain hard-coded DL/I parameter lists. You should review them to ensure that these parameter lists are appropriate for your installation. Some modifications may be necessary depending on which release of IMS/VS is used at your installation. Any modifications to these skeletons should be done prior to attempting to use any of the IMS/VS-related functions of IADF.

ADFBTSM	Contains a BTS control statement which places '999999' in the IOPCB userid field. It enables users to perform BTS testing of the supplied applications (SYSID=???? and SAMP) whether or not the RACF DEFADF option has been implemented. Data supplied for loading test data bases contains sign-on profile information for userid '999999'. Please refer to <i>IMS Application Development Facility II Version 2 Release 2 Installation Guide</i> for more information on this option.
ADFDATAS	Performs a data base load for the IMSADF II data bases.
ADFDBBPR	IMS/VS 1.2 batch procedure which uses ACBLIB.
ADFDLIPR	IMS/VS 1.2 batch procedure.
ADFINIT2	Allocates IMSADF II Dynamic rules data bases. This member contains buffer specifications for the Work Data Base. If your data base requires larger buffers because of the size of the segments, you should change the DFSVSAMP DD contents.
ADFMIJCL	Creates Phase II migration job stream. This member allocates the ISP- DDNAMEs used by migration. The data set names are customized with both ISPNODE and ADFNODE (values from the PRODUCT table or from your profile). You may modify this if you have to do extensive editing of the job stream to accommodate extra data sets. See Appendix D, "Sample Migration User Exit" for more information.
ADFWORKI	Scratches, allocates, and initializes the IMSADF II WORK data base.
PROCALS	Executes the High Level Audit Compiler.
PROCBS	Executes the IMSADF II Batch Driver to update the IMSADF II data bases.
PROCRDCS	Executes the Rules Document facility under the IMSADF II Batch Driver.

If the IADF Data Dictionary interface is to be used at your installation, some modification to the IADF skeletons may be necessary if the Dictionary load modules, DBDs, and PSBs are not found in the IMS/VS data sets. If your installation has separate Dictionary data sets, please modify the ADSKDD, ADSKDDRP, ADSKDD1 and ADSKDD2 skeletons in the following manner:

- The Data Dictionary DDPGMLIB must be in the STEPLIB concatenation.
- The Data Dictionary PSBLIB must be in the DDPSBLIB concatenation.

- Clists:** are provided in IMSADF.ADFCLIB for use in IADF.

ADFBTSA This clist contains a hard-coded DL/I parameter list. It is recommended that you review this parameter list to ensure that it is appropriate for your installation. If you wish to use DBRC on test data bases, you must change this clist.

This clist contains a hard-coded DL/I parameter list. It is recommended that you review this parameter list to ensure that it is appropriate for your installation. If you wish to use DBRC on test data bases, you must change this clist.

DIRDF(20) change 20 to your value

```
SET SESP=&EVAL(&PRIM&I)/2)
SET SESP=&SESP+(&PRIM&I)/3)
```

If you have TSO/E and you wish to preserve lower case in the prototyping activities, you may change the following clists. Place the word "ASIS" in each CONTROL statement that is **not** preceded by the word "ISPEXEC".

Chapter 2. Preparing TSO/ISPF to Use IADF 2-11

Migration Requirements

IEBCOPY

The MIGRATION function of IADF invokes IEBCOPY to move/copy rules and modules from test to production libraries. The clist IMSADF.ADFCLIB(ADFCOPY) invokes IEBCOPY in this way:

```
ISPEXEC SELECT PGM(IEBCOPY)
```

If you are running in a TSO/E Release 2 environment, this invocation is satisfactory. If you do not use this version of TSO/E and you copy libraries using "SPFCOPY", you must change that line in ADFCOPY to select the program SPFCOPY.

Please refer to *ISPF Installation and Customization* SC34-2143, under the topic "Foreground Compress Procedures" found in the PDF installation section for a full discussion of techniques you may take to invoke IEBCOPY in an unauthorized ISPF environment.

LMF Support

LMF services are invoked on behalf of MIGRATION (the move function), which does not require the installation of the LMF user SVC. The LMF "PROMOTE" function is not required for IADF migration.

However, IADF supports PROMOTE if you indicate that you use LMF on a migration panel. In that case you must have the SVC installed, and the library architecture in place.

TSO Performance

The following list presents some of the ways you can influence the TSO environment for IADF. Some of them affect specific releases of TSO and ISPF. Some are relatively easy to implement, while others require more evaluation. In certain cases the performance gains should be traded off against loss of flexibility and the difficulty of modification and maintenance.

1. Place ISPF modules in the Link Pack Area (LPA) and, if you are in MVS/XA, place them above the 16 megabyte line.
2. Migrate to TSO/E if possible, since it contains several improvements for clist performance. One of these is a reduction in path length for SET and IF statements, which are widely used in clist processing.
3. For foreground invocation of IADF services, eliminate unnecessary data sets to reduce allocation time. Eliminating DDNAMES which are not used in your environment reduces the number of DD statements allocated. For more information, please see "TSO Data Set Allocation Levels" on page 2-7.
4. If resources for supporting foreground execution are limited, perform the following in batch:
 - Rules Generator
 - Extraction of static rules
This is important if you are extracting hundreds of segments.
 - Batch driver invocation.

5. Preallocate foreground data sets. You may allocate DDNAMES and their data sets used in IADF foreground services at TSO logon. This reduces allocation time during the invocation of the service, but does limit your flexibility. You may wish to allocate all permanent data sets found in these tables:

- “Establish Foreground Static data sets for the SYSID” on page 7-7
- “Establish Foreground Dynamic Data sets for the SYSID” on page 7-11
- “Establish BTS Data sets for the SYSID” on page 7-14

All data sets in a selected DDNAME concatenation must be moved to the logon procedure. You must then use the “delete” command available with each table to delete all DDNAMES (and concatenated data sets) from the foreground tables which will now be preallocated.

6. Your developers can remove inactive fields from the transaction control table. These tables can contain many fields which may never be referenced by a user or IADF command. By removing inactive fields, you can eliminate several ISPF table operations when you perform transaction generation, effectively reducing elapsed as well as CPU time. “Transaction Control Tables” on page 10-23 describes these operations.
7. The table library for each SYSID, described as “ISPF tables” in Chapter 5, “Installing an Application SYSID,” should contain no more than two hundred or so members to avoid long waits for IADF menu selections. In addition if the developer knows the segment or transaction which is to be changed or created, he may supply the key and avoid the menu construction process.
8. A technique you may wish to employ is to preprocess the panel library if you use ISPF Version 2 Release 2. Part of the display processing time is eliminated by using the internal panel format. In addition space requirements may be reduced by as much as 20%. Once the panel is processed by an ISPF utility, ISPPREP, it can no longer be edited.

To preprocess panels, you must copy the supplied library (IMSADF.ADFPLIB) and make changes to it.

Most of the IADF dialog panels contain the following instruction:

```
)BODY CMD(ZCMD) WIDTH(&ZSCREENW) EXPAND(##)
```

One restriction of ISPPREP is that a variable may not be used to define the panel width, so you must change it to read:

```
)BODY CMD(ZCMD) WIDTH(80) EXPAND(##)
```

You may use the IADF utility GFIND to make the change.

You may then invoke ISPPREP. Panel ADFPROB2 will not be converted, but you may copy it to the same library as the converted panels.

Multiple ADFID Considerations

If you have more than one ADFID in the same release of IMSADF II, you have unique libraries for the libraries whose high level node is NEWADF, as described in the *IMS Application Development Facility II Version 2 Release 2 Installation Guide*. You have only one set of ISPF IMSADF II libraries. You may process multiple ADFIDs with one IADF clist if you add the load library (IMSADF.ADFLOAD) from each ADFID to ISPLLIB (if this library is not shared among them already).

Establish the proper ADFID as shown:

```
IADF 'ADFID(adfid1)'  
or  
IADF 'ADFID(adfid2)'
```

If you have modified the ISPF Primary Option Menu, you will have to do so again to add the same option for the next ADFID.

Chapter 3. Getting Started

Preparation for Application Development

The steps listed below are those you must perform before applications can be developed interactively. The tasks are divided into two groups: those that pertain to the IMSADF II product, known as the ADFID; and those that concern a specific application system, known as a SYSID.

Note: You may have more than one ADFID installed in your location.

The activity relating to ADFID is:

Add Administrators to this ADFID

The activities pertaining to an application SYSID are:

1. Add the Application System (SYSID).
2. Add Developers to the Application System (SYSID).
3. Create Static Rule (RULEGEN) data sets.
4. Create Dynamic data sets.
5. Create BTS data sets.
6. Allocate the Application System (SYSID) Data Bases.

At the beginning of each chapter this list is presented with the step which is the chapter topic highlighted, so that you may see approximately where you are in the administration process.

The sequence of steps may vary somewhat, but the following chapters discuss them in this order.

Required Data

Installation data you may require for the steps described in this guide is listed in Figure 3-1. This information is used to create JCL and dynamically allocate data sets to your TSO userid. Fill in what is valid for your installation. If you do not know this information, talk with your IMSADF II installer.

Note: In order to use IADF, your installer must have chosen the DEFADF parameter RGLIB=L. Object text form of exits is not supported by IADF.

Environment = ____ (IMS/VS or CICS)

DB2 for IMSADF II D/B = ____ (N = use DL/I, Y = DB2)

ADFID = _____

SYSID = _____ (4-character application system identifier)

SYSID = SAMP for this example.

PGROUP = _____ (2-character project/group code)

PGROUP = PG for this example.

Node for shared data sets (ADFNODE) = _____

Node for Unique data sets (NEWADF) = _____

IMS/VS RESLIB node (IMSNODE) = _____

IMS/VS DBD/PSB node (IMSTEST) = _____

CICS node (CICNODE) = _____

DBDGEN procedure name = _____

PSBGEN procedure name = _____

ACBGEN procedure name = _____

MFSUTL procedure name = _____

Figure 3-1. Installation Data Required for Administering IADF

Levels in IADF

The administrative services of IADF supports two categories of users which contain three levels each.

- Administrative users are registered in the SPECUSER table. In this publication they are formally titled "Administrators", but in previous versions they were designated as "Special Users." The levels are described below in terms of decreasing authority.

Level 3

Global IADF administrators and have access to all systems (SYSIDS) defined and IADF versions (ADFIDs). Typically the IMSADF II administrator, an ISPF systems programmer, and/or backup personnel would have this designation.

Level 2 Extends throughout the SYSID boundaries. They may create or modify application systems (SYSIDs) defined in a particular IADF version (ADFID). Personnel carrying this designation may be project managers in charge of specific SYSIDS and/or lead developers.

Non-administrative user Modify only their own IADF/ISPF profile. Any user who is not designated as an Administrator to IADF falls into this category.

- Development user levels are ordered in terms of access within an application system (SYSID). They may be given different levels in different SYSIDs. The levels are described below in terms of decreasing authority.

Level 3 Read, Write, and Delete capability for all activities within a SYSID. Personnel carrying this designation may be project managers in charge of specific SYSIDS and/or lead developers. If you are an Administrator, you must register yourself as a Developer of a given SYSID, normally at this level, if you wish to manage the development activities for this SYSID (otherwise you are a Level 1 Developer).

Level 2 Read, Write, but no Delete access for SYSID activities.

Level 1 Modify only their own IADF/ISPF profile. Otherwise, they may browse a SYSID and invoke the Rules Generator.

This manual does not pertain to Developers, with the exception of “Establish ISPF User Profile for IADF” on page 10-1 and “Extract ISPF Tables from Rules Source” on page 10-6.

Some parts of the remainder of this guide apply only to Level 3 Administrators.

Bringing up Administrator Screens

PANELID is an ISPF/PDF command which displays the panel name in the upper left hand corner of the screen. Although under normal conditions of IADF use, you may prefer not to utilize the PANELID command, the screens presented in this guide contain the panel name as though the command had been issued. You may verify them against the name of the your panel.

The Primary Function Key (PFK) values listed below are used in the IADF administrative panels. There is an IADF command, PFK, which invokes ISPF services to set or to display those values.

When you use IADF the commands of greatest interest to you are:

HELP Displays a HELP panel.

END Returns to the previous panel.

RETURN Returns to the IADF Administration Main Menu or to a panel from which a HELP or KEYS command was entered, without displaying the intervening panels.

Note: If you have chosen 6 for DEVELOPMENT, the RETURN PF key brings you back to the IADF Main Menu, not the Administration Main Menu.

UP	Causes a scroll toward the top of data.
DOWN	Causes a scroll toward the bottom of data.
CURSOR	Moves the cursor to the first input field on line 2, normally the command or option field.

IADF panel format is consistent with that of ISPF. The short message appears in the upper right-hand corner of the panel, while the long message occupies the third line of the panel. Instructions for a menu panel are found at the bottom of the panel. Panels which are not menus have instructions placed within the body of the panel, where necessary.

IADF uses ISPF services to invoke certain functions such as data set allocation. You may see normal ISPF screens displayed during the course of your administrative duties if you require those services.

Invoke IADF as described in “TSO Clists” on page 2-1.

If the user is not an Administrator, the first IADF panel to appear, under normal logon conditions, is the IADF Main Menu shown in the Figure 3-2. If the user is an Administrator, the IADF Administration Main Menu is displayed initially. Option 6 on the Administration Main Menu brings the administrative user to the IADF Main Menu shown in the Figure 3-2. The IADF Main Menu groups the functions available into 6 major categories. Each category is identified by a number preceding the category name. You enter the number for the category you wish to process on the option line, which is near the top of the panel.

- | | |
|---------------------|---|
| 1 DEFINITION | <p>Supports the definition of all components of an IMSADF II application, including:</p> <ul style="list-style-type: none"> • Data Layouts (Segments and Fields, DB2 Tables and Columns, etc.) • Panels (Transaction Display panels, Signon, Primary Option Menu, HELP panels for the end-user) • Transactions • Audits • Messages • IMSADF II Userids and Signon Profiles for end-users • Transaction Drivers • SYSIDs |
| 2 GENERATION | <p>Supports the generation of all components defined to IADF. The generation step is analogous to a compilation step; it is necessary prior to executing an IMSADF II application.</p> |
| 3 TEST | <p>Supports the developer in testing the IMSADF II application. This includes both prototyping and BTS (Batch Terminal Simulator) testing (IMS/VS environment only).</p> <ul style="list-style-type: none"> • The IADF Prototype feature provides IMSADF II screen definition, conversion of IMSADF II screens to ISPF format, and dialog simulation that does not require previous execution of the Rules Generator. |

- The BTS interface allows the developer to test an IMSADF II application using actual IMS/VS data bases. The generation step must be completed prior to running BTS.

4 UTILITIES

Allows the developer to:

- Extract information to build IADF tables from static and dynamic rule source
- Submit jobs
- Access the IADF online Glossary
- Invoke Rules Documentation
- Browse a list of the IADF libraries
- Set IADF and IMSADF II trace options
- Invoke GFIND to perform a global search of a partitioned data set
- Call the MFS Utility
- Invoke the IMSADF II interface to the IBM Data Dictionary
- Use the RGLGEN utility, which extracts information from the DB2 Catalog

5 IADF PROFILE

Tailors the IADF functions to user needs. This is an ISPF profile, not the signon profile used in the execution of IMSADF II.

6 TUTORIAL

Presents a complete tutorial, ranging from general topics to details on IADF and IMSADF II.

MFC1 ----- IADF MAIN MENU -----		
OPTION ==>		
KEY ==>	SYSID ==>	LEVEL:
1	DEFINITION	- Create, modify or browse application information (Data Layouts, Panels, Transactions, Audits, Messages, Userids and Profiles, Transaction Drivers, SYSIDs)
2	GENERATION	- Perform application generation
3	TEST	- Prepare for and perform test (PROTOTYPE, BTS)
4	UTILITIES	- Perform IADF Utilities (Extract, Submit, Glossary, Documentation, SYSID Info, Trace, GFIND, MFS Utility, Data Dictionary, DB2 RGLGEN)
5	IADF PROFILE	- Create or modify IADF User Profile
6	TUTORIAL	- Display information about IADF
X	EXIT	- Terminate IADF DEVELOPMENT

ENTER **OPTION NUMBER** OR PRESS **END KEY** TO EXIT

Figure 3-2. IADF Main Menu Panel

The IADF Main Menu

The IADF version (ADFID) is shown in the far left part of the panel heading. SYSID is the four-character identifier of the IMSADF II system to be processed. The first two characters of SYSID must be unique to the IADF version. Initially no SYSID is displayed because none has been installed as yet. Subsequently, a value for this keyword is displayed.

LEVEL is the authorization level of the developer of an application system (SYSID), not an administrator. No value is displayed until you have attempted to utilize a function that requires level verification prior to execution.

KEY may be filled in with an appropriate value, such as Segment identifier, during application development, but it is not used for administrative functions.

```
IADFMAIN ----- IADF MAIN MENU -----
OPTION ==>

KEY ==>                SYSID ==>        LEVEL:

  1  DEFINITION        - Create, modify or browse application information
                        (Data Layouts, Panels, Transactions, Audits, Messages,
                        Userids and Profiles, Transaction Drivers, SYSIDs)

  2  GENERATION        - Perform application generation

  3  TEST              - Prepare for and perform test (PROTOTYPE, BTS)

  4  UTILITIES         - Perform IADF Utilities
                        (Extract, Submit, Glossary, Documentation, SYSID Info,
                        Trace, GFIND, MFS Utility, Data Dictionary, DB2 RGLGEN)

  5  IADF PROFILE      - Create or modify IADF User Profile

  6  TUTORIAL          - Display information about IADF

  X  EXIT              - Terminate IADF DEVELOPMENT

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT
```

Figure 3-3. IADF Main Menu (with Panel ID displayed)

Notice that the name of the IADF Main Menu is IADFMAIN. This panel identifier can be displayed by entering the PANELID command on the COMMAND or OPTION line of any IADF panel.

Help Facility

A full HELP facility is provided for detailed information on IADF. You should expect to gain most of your technical information from it.

Press the HELP PF key (usually PF1) or ENTER HELP on the option line to see an example of a HELP panel.

```
HAGMD0 ----- HELP: IADF HELP MAIN MENU -----
COMMAND ==>

The following topics are presented in sequence, or may be selected by
number:

0 GENERAL      - General Information about IADF

1 DEFINITION   - Create, modify or browse application information
                 (Data Layouts, Panels, Transactions, Audits, Messages,
                 Userids and Profiles, Transaction Drivers, SYSIDs)

2 GENERATION   - Perform application generation

3 TEST        - Prepare for and perform test (PROTOTYPE, BTS)

4 UTILITIES    - Perform IADF Utilities
                 (Extract, Submit, Glossary, Documentation, SYSID Info,
                 Trace, GFIND, MFS Utility, Data Dictionary, DB2 RGLGEN)

5 IADF PROFILE - Create or modify IADF User Profile

X EXIT        - Terminate IADF
```

Figure 3-4. HELP Panel for IADF Main Menu

Figure 3-4 is the HELP panel for the IADF Main Menu. Some of the HELP panels, like this one, are menus in themselves. Entering the option number on the command line displays a panel containing HELP information on the option chosen.

HELP may be accessed from any of the IADF panels by pressing the HELP PF key. Return to the original menu by pressing the END PF key (usually PF3).

Tutorial

Tutorial services are provided when you select that option on the IADF Main Menu as shown in Figure 3-5.

```
IADFMAIN ----- IADF MAIN MENU -----
OPTION ==> 6

KEY ==>          SYSID ==>          LEVEL:

  1  DEFINITION    - Create, modify or browse application information
                        (Data Layouts, Panels, Transactions, Audits, Messages,
                        Userids and Profiles, Transaction Drivers, SYSIDs)

  2  GENERATION    - Perform application generation

  3  TEST          - Prepare for and perform test (PROTOTYPE, BTS)

  4  UTILITIES     - Perform IADF Utilities
                        (Extract, Submit, Glossary, Documentation, SYSID Info,
                        Trace, GFIND, MFS Utility, Data Dictionary, DB2 RGLGEN)

  5  IADF PROFILE  - Create or modify IADF User Profile

  6  TUTORIAL      - Display information about IADF

  X  EXIT          - Terminate IADF DEVELOPMENT

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT
```

Figure 3-5. IADF Main Menu (with Panel ID displayed)

To view the first panel of the tutorial, select option 6.

HAGD8010 ----- IADF TUTORIAL ----- TUTORIAL
COMMAND ===>

This tutorial provides information about the features and operation of the Interactive Application Development Facility.

You can use the information in two different ways:

View all of the information in sequence. To do this
press the ENTER key each time you are ready for the next panel.

Select only parts of the information to read. To do this
enter the TOP command. This will take you directly to
the table of contents. Make your selection on that panel.

The next two panels contain more details about how to use this tutorial.

Press the **ENTER** key to go to the next panel, or
type the **TOP** command to go directly to the table of contents, or
type **SPF/PDF/ISPF** commands to go to the ISPF primary option menu, or
press the **END** key to return to the IADF Main Menu.

Figure 3-6. First Page of IADF Tutorial

This is the first panel of the tutorial. The first part of the tutorial teaches you how to use the rest of the tutorial.

Press the **END** PF key to return to the IADF Main Menu.

IADF Main Menu (Advanced User)

There is another way to invoke IADF services instead of using the IADF Main Menu. It is shown in Figure 3-7.

```

IADFADVM ----- IADF MAIN MENU (Advanced User) -----
OPTION ==>

KEY ==>          SYSID ==>          VERSION ==>          USER: TS0ID1
                                                           LEVEL:

----- DEFINITION -----
 1 ADFPROF      19 RREK
 2 AUDSEG      20 SDT
 3 AUDSUB      21 SEGMENTS
 4 AUDTBL      22 SIGNON
 5 AUDTRX      23 SYSTEM
 6 DBEK        24 TBL
 7 DBS         25 TRX
 8 DRIVER      26 USER
 9 ESDS
10 HELPPANL
11 KSDS
12 MAP
13 MESSAGE
14 OUT
15 POMENU
16 PS
17 ROUTES
18 RRDS

- GENERATION -
27 AUDITB
28 AUDITF
29 DYNAMICB
30 DYNAMICF
31 FREEDYNA
32 FREESTAT
33 STATICB
34 STATICF

----- TEST -----
35 BTS
36 FREEBTS
37 PROTOTYPE

----- UTILITIES -----
38 ADFIN      51 MFSGEN
39 ADFOUT     52 OUTLIST
40 ADFTRACE   53 PFKS
41 BROWSEF    54 PROFILE
42 DBDGEN     55 PSBGEN
43 DDREPORT   56 REPORTS
44 EDITOR     57 RGLGEN
45 EXTRACT    58 SPF
46 GFIND      59 SUBMIT
47 GLOSSARY   60 SYSID
48 IADFTRACE  61 TERMINAL
49 LOG        62 TUTORIAL
50 MIGRATION

ENTER NAME OR OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 3-7. IADF Main Menu (Advanced User)

The IADF Main Menu (Advanced User) eliminates menu selections. You may now execute all IADF functions with a maximum of two panel levels. In most cases you execute the function directly. To activate the IADF Main Menu (Advanced User), you enter a "Y" for IADF Advanced Menu on your IADF User Profile panel (Figure 10-4 on page 10-4).

You should use this panel when you perform migration (see "Migration Menu Selection" on page 9-14).

IADF Administration Main Menu

The primary focus of this guide is on the options available on the IADF Administration Main Menu, displayed in Figure 3-8.

```
IADFADMN ----- IADF ADMINISTRATION MAIN MENU -----
OPTION ==>

ADMINISTRATIVE LEVEL:3
KEY ==>                SYSID ==>                LEVEL:

  1  SYSTEMS          - Create or modify IMSADF II  systems (SYSID)

  2  MIGRATION        - Perform migration    TYPE ==> M ( M Move  C Copy )

  3  ADMINISTRATORS   - Specify administrative users

  4  TABLES          - Create, modify, and/or copy ISPF tables

  5  GLOSSARY         - Update glossary terms

  6  DEVELOPMENT      - Perform IMSADF II  development tasks

  X  EXIT             - Terminate IADF

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT
```

Figure 3-8. IADF Administration Main Menu

The rest of this manual is spent in describing its functions:

ADMINISTRATORS	Chapter 4, "Add ADFID Administrators"
SYSTEMS	Chapter 5, "Installing an Application SYSID"
MIGRATION	Chapter 9, "Migration Services"
TABLES	"Table Management" on page 10-31
GLOSSARY	"Glossary Services" on page 10-28

Chapter 4. Add ADFID Administrators

You are at the point in your administrative duties that is highlighted in the following list:

1. **Add Administrators to this ADFID.**
2. Add the Application System (SYSID).
3. Add Developers to the Application System (SYSID).
4. Create Static Rule (RULEGEN) data sets.
5. Create Dynamic data sets.
6. Create BTS data sets.
7. Add the Application System (SYSID) Data Bases.

IADF Administrators of this ADFID

This chapter shows you how to add (or delete) Administrators for an ADFID.

This activity updates the ISPF table 'SPECUSER', where each row denotes those Administrators for an ADFID. It contains entries for two types of Administrators: Level 3 and Level 2.

Level 3 Administrators may access all IADF versions and all application systems (SYSID) defined. This highest category has all ADFID global functions and may also add additional Administrators. The number of Level 3 Administrators should be restricted. Persons who typically have this function are the IMSADF II installer, IMSADF II administrator, and/or backup personnel.

The scope of Level 2 Administrators is that of an application system (SYSID). They may create or modify any SYSID defined under an ADFID. They may administer all functions within that SYSID. They may not access or create other IADF versions and they may not access or modify the SPECUSER table. An application project manager would be normally a Level 2.

Note: The level you possess as an Administrator does not carry over to the development (SYSID) user.

```

IADFADMN ----- IADF ADMINISTRATION MAIN MENU -----
OPTION ==> 3

ADMINISTRATIVE LEVEL:3
KEY ==>          SYSID ==>          LEVEL:

  1 SYSTEMS      - Create or modify IMSADF II  systems (SYSID)
  2 MIGRATION    - Perform migration    TYPE ==> M ( M Move  C Copy )
  3 ADMINISTRATORS - Specify administrative users
  4 TABLES      - Create, modify, and/or copy ISPF tables
  5 GLOSSARY     - Update glossary terms
  6 DEVELOPMENT  - Perform IMSADF II  development tasks
  X EXIT         - Terminate IADF

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 4-1. Selecting Administrators from IADF Administration Main Menu

Administrators may be created to distribute the IADF administration tasks. Only Administrators of Level 3 can add or delete other Administrators.

Enter 3 on the option line to add an Administrator.

```

ADFPR08 ----- ADMINISTRATORS -----
COMMAND ==>          SCROLL ==> PAGE

Line Commands: D Delete  I Insert
CMD ADFID USERID    Level Description (optional)

'  MFC1  TSOID1      3    IADF ADMINISTRATOR
***** BOTTOM OF DATA *****

```

Figure 4-2. First Level 3 Administrator

The TSO userid of the individual who installed the first ADFID for IMSADF II is automatically inserted into the special user table with a level of 3.

```
ADFP08 ----- ADMINISTRATORS -----  
COMMAND ==> i mfc1 tsoid2                                SCROLL ==> PAGE  
  
Line Commands: D Delete I Insert  
CMD ADFID USERID   Level Description (optional)  
  
'  MFC1  TSOID1    3    IADF ADMINISTRATOR  
***** BOTTOM OF DATA *****
```

Figure 4-3. Inserting an Administrator Example

To insert a second Administrator, enter "I" followed by the ADFID and the TSO userid on the command line.

Note: An alternate method of inserting is to key I in the line command column of the current entry, key the new ADFID, userid, level, and description over the original ones, then press ENTER. This adds another entry but does not destroy the original entry.

```

ADFPR08 ----- ADMINISTRATORS ----- Special User Added
COMMAND ===>                                SCROLL ===> PAGE

Line Commands: D Delete I Insert
CMD ADFID USERID   Level Description (optional)

'  MFC1  TS0ID2    3    USER 1
'  MFC1  TS0ID1    3    IADF ADMINISTRATOR
***** BOTTOM OF DATA *****

```

Figure 4-4. Administrator Added

The modified special user table is displayed. A confirmation message stating that the profile has been updated appears on this panel. When all modifications have been made, press the END PF key.

```

IADFADMN ----- IADF ADMINISTRATION MAIN MENU -----
OPTION ===>

ADMINISTRATIVE LEVEL:3
KEY ===>                                SYSID ===>                                LEVEL:

  1 SYSTEMS      - Create or modify IMSADF II  systems (SYSID)

  2 MIGRATION    - Perform migration    TYPE ===> M ( M Move C Copy )

  3 ADMINISTRATORS - Specify administrative users

  4 TABLES      - Create, modify, and/or copy ISPF tables

  5 GLOSSARY     - Update glossary terms

  6 DEVELOPMENT  - Perform IMSADF II  development tasks

  X EXIT        - Terminate IADF

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 4-5. IADF Administration Main Menu

The IADF Administration Main Menu is displayed.

IADF Panel Format

This chapter contains the first example of a screen format which is common to all the IADF panels, as shown in Figure 4-2. A table approach is used with descriptive headings listed above the appropriate columns. Above that are shown the valid commands that you may enter in the CMD column.

Keyed tables are distinguished from non-keyed tables by the length of the command column (CMD). Keyed tables have a one-character CMD column. Non-keyed tables have a three-character CMD column.

The table described in Figure 4-2 is a keyed table, denoted by the single quotation in the CMD column. Use of the Insert on the command line instead of in the CMD column is recommended on all IADF keyed tables.

Multiple ADFID Environment

The installer of the *first* ADFID automatically becomes a Level 3 Administrator. This action does not occur for subsequent ADFIDs. The Level 3 Administrator for the first ADFID may add any other Administrators desired, including those for additional ADFIDs.

Chapter 5. Installing an Application SYSID

You are at the point in your administrative duties that is highlighted in the following list:

1. Add Administrators to this ADFID.
2. **Add the Application System (SYSID).**
3. Add Developers to the Application System (SYSID).
4. Create Static Rule (RULEGEN) data sets.
5. Create Dynamic data sets (DYNAMIC).
6. Create BTS data sets.
7. Allocate the Application System (SYSID) Data Bases.

If you developed applications under IADF in a previous release of IMSADF II, you must convert your existing tables to the format required in Version 2 Release 2 instead of continuing further in this chapter.

Follow the instructions in “Converting IADF Tables from Prior IMSADF II Releases” on page A-1 of this book. Any new SYSIDS developed under Version 2 Release 2 may then be added following the dialog in this chapter. **Convert existing tables immediately after you perform the tasks indicated in Chapter 4, “Add ADFID Administrators” on page 4-1 .**

Multiple Version SYSIDS

Many installations require multiple levels of testing (with the ability to perform developmental activities at each level) of an application system (SYSID) before it is migrated to production libraries. You may establish multiple versions of **one** SYSID using IADF. Because an installation may have a pre-defined change control flow external to IMSADF II, IADF imposes no limits on the number of versions of a SYSID which may be established. This migration concept is called **versioning**, and its use is optional.

Implementation of multiple version SYSIDS is spread across these tasks:

1. Initialize SYSID (“Add the Application System” on page 5-2)
2. Add Developers (Chapter 6, “Adding Developers to the SYSID”)
3. Migration (Chapter 9, “Migration Services”)

It is recommended that you employ versioning. In addition to your having more than one version of a SYSID on which you can perform development activities or maintenance, migration services can provide more structured change control and authorization verification. Since versioning is closely related to migration, please refer to “Migration Concepts” on page 9-1 for more information.

To create multiple versions of a SYSID, a four-character version identifier is added at SYSID initialization. The default is **BASE**. It is identified in the SYSTEMS table with a key of

ADFID SYSID VERSION

The Administrator defines any migration paths desired. He is presented with a panel at initialization which shows all existing versions, and must indicate if the new

SYSID version can migrate "TO" or "FROM" them. Paths are stored in the ADFVER table (in IMSADF.ADFTLIB).

It is possible to "wrap around" versions and overlay them. For example, if

```
SYSID Version BASE ==> SYSID Version TEST and  
SYSID Version TEST ==> SYSID Version PROD and  
SYSID Version PROD ==> SYSID Version BASE
```

then PROD could overlay TEST

Though the version paths allow for maximum flexibility, in practice each installation usually follows a constant migration pattern known to the Administrator who must establish migration paths. Migration paths for a SYSID may be changed after initialization by a Level 2 Administrator using special utilities (see "SYSID/Migration Path Maintenance" on page 5-16).

A SYSID version may be migrated to any other version of the **same** SYSID in the **same** ADFID.

At migration time, if a version has more than one path, the user is presented with all the possible migration paths to decide which one of them is the target.

Key versioning concepts are:

- Each version is initialized as a separate SYSID with separate libraries and tables.
- You **can** overlay versions when paths are chosen improperly.
- Migration occurs within multiple versions of a **single** SYSID within **one** ADFID.

Multiple version SYSIDS and their migration have been implemented through PTFs distributed to IMSADF II Version 2 Release 2.

Before you continue further in this chapter, refer to "Converting Tables to Versioning" on page A-6 and perform the tasks described.

Add the Application System

This chapter shows you how to install an application system (SYSID) for later use by developers, which is the first step of SYSID-related administrative activities.

This series of panels is available for use **ONLY** by these groups:

- Level 3 Administrators (ADFID scope)
- Level 2 Administrators (SYSID scope)

Each time you install a SYSID, you add a row to a table called SYSTEMS. The table is also keyed by ADFID, so you can have multiple SYSIDS in the table.

```

IADFADMN ----- IADF ADMINISTRATION MAIN MENU -----
OPTION ==> 1

ADMINISTRATIVE LEVEL:3
KEY ==>          SYSID ==>          LEVEL:

  1 SYSTEMS      - Create or modify IMSADF II  systems (SYSID)

  2 MIGRATION    - Perform migration    TYPE ==> M ( M Move  C Copy )

  3 ADMINISTRATORS - Specify administrative users

  4 TABLES      - Create, modify, and/or copy ISPF tables

  5 GLOSSARY     - Update glossary terms

  6 DEVELOPMENT  - Perform IMSADF II  development tasks

  X EXIT        - Terminate IADF

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 5-1. Select SYSTEMS from the IADF Administration Main Menu

```

IADFSYS ----- IADF SYSTEM DEFINITION MENU -----
OPTION ==> 1

ADMINISTRATIVE LEVEL:3
KEY ==>          SYSID ==> samp          LEVEL:

  1 LIBRARIES    - Create and initialize the libraries which
                  comprise an IMSADF II  system

  2 DEVELOPERS   - Specify TSO userids of developers who have
                  access to an IMSADF II  system

  3 STATIC DSNS  - Specify data sets for generation of data layouts,panels,
                  transactions, drivers    TYPE==> F (F=Foreground B=Batch)

  4 DYNAMIC DSNS - Specify data sets for generation of audits, messages,
                  help panels, routes, secondary destinations, userids,
                  and profiles            TYPE==> F (F=Foreground B=Batch)

  5 BTS DSNS     - Specify data sets for BTS execution

  6 INITIALIZE   - Initialize IMSADF II  libraries and data bases

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 5-2. IADF System Definition Menu

A "system", known as a SYSID in IMSADF II terms, has to be registered in order to do application development tasks in IADF. To define a system, it is necessary to create and initialize the libraries which comprise it. The system contains all the static and dynamic rules for a SYSID in ISPF tables. Using these tables, you are able to generate a complete IMSADF II application.

Select option 1 to create the libraries for a new SYSID. Enter your SYSID in the SYSID field.

Note: Naming conventions dictate that the first two characters should be different from any other SYSID or ADFID in your installation.

If this is the first SYSID you are installing, you will see Figure 5-4.

If you have existing SYSIDS, you are presented with Figure 5-3.

```
ADFP04 ----- SYSTEMS WITHIN MFC1 -----
COMMAND ==> i samp                                SCROLL ==> PAGE

Line Commands: D delete I initialize R reinitialize U update
CMD SYSID VERS DDNAME  Data Set Name              Description
-----
'  BANK  TEST BANK    TS0ID1.TEST2.ISPFTBLS      BANK SYSTEM
'  ABCD  BASE ABCE    TS0ID1.TEST3.ISPFTBLS      ANOTHER ONE
***** BOTTOM OF DATA *****
```

Figure 5-3. Display Existing SYSIDS for an ADFID

To initialize the new SYSID enter "i samp" on the command line.

Press ENTER.

Name Data Sets for this SYSID

```
ADFP05 ----- INSTALLATION OF A NEW SYSTEM ID -----
COMMAND ==> I

Available commands: C Cancel  D Delete  I Initialize  R Reinitialize  U Update
SYSTEM ID  ==> SAMP
VERSION ID  ==> BASE
Description ==> IMSADF II Sample

Libraries and Data Bases to be used as targets for this system are:

  TYPE          DDNAME  DATA SET NAME
Rules source (1)  RULSRC  TS0ID1.TEST.RULSRC
Rules load  (2)  RLOAD   TS0ID1.TEST.RULLIB
Dynamic source(3) DYNSRC  TS0ID1.TEST.DYNSRC
Image source (4)  IMAGE   TS0ID1.TEST.ADFIMG
Program Load (5)  PROG    TS0ID1.TEST.RULLIB
Composite load(6)
ISPF tables  (8)  SAMP    TS0ID1.TEST.ISPFTBLS

Audit Data Base(21)----- TS0ID1.TEST.AUDIT
Message          (22)----- TS0ID1.TEST.MSG
Signon           (23)----- TS0ID1.TEST.SIGNON
Work             (24)----- TS0ID1.TEST.WORK
NOTE:
      Complete the above information and press END to
      Insert the request for a System Initialization
```

Figure 5-4. SYSID Installation Example

Refer to Figure 5-4.

The top third of the panel is as follows:

- Type "I" on the Command line to indicate that the system is to be initialized.
- The four-character SYSID is displayed beside the "SYSTEM ID" heading.
- Enter a four-character (non-blank) version identifier. The default is "BASE." Refer to "Multiple Version SYSIDS" on page 5-1 for details.
- Enter a short description of the application beside "Description".

The data sets shown are used in application development; they give you added flexibility by allowing you to specify your own libraries in addition to those required by the Rules Generator. Most are later referenced by the IMSADF II programs in a concatenation. They are displayed on later selection panels for IMSADF II functions, such as Figure 7-9 and Figure 7-6. For example, "Dynamic Source" is the library which contains input to the High Level Audit Language (HLAL).

The number beside the data set in the "Type" column is the same number you see later in Figure 7-6 of the form "&DSNn".

The DDNAME is used to dynamically allocate the library for TSO during IADF processing. Accordingly, it should not be a specific DDNAME for an IMSADF II program nor ISPF nor any DDNAME that could be allocated during a TSO session, so that you do not receive errors.

Enter DDNAME and data set names describing the libraries you wish developers to use for this SYSID. Defaults that appear may remain from a previous SYSID. The HELP panels give you assistance on the purpose of the libraries.

Note: If you do not use Composite Rule Load modules (installation parameter COMPMOD=NO), blank the DDNAME to bypass it.

These data sets are used in migration and must exist or be allocated in the current activity:

- Rules Source (&DSN1)
- Rules Load (&DSN2)
- Dynamic Source (&DSN3)
- ISPF tables (&DSN8)
The "ISPF Tables" is a data set which contains the initialization information about this SYSID. You **must** specify a unique ISPF Table data set for each SYSID (to avoid enqueues during development).

For multiple version SYSIDS, each DDNAME and data set name **must be unique in each version** as migration dynamically allocates them. Different SYSIDS within the same version may share the same set of libraries.

All data set names must be fully qualified, without enclosing quotation marks. The data sets do not have to exist at this time, though some may if you are referencing data sets for an existing SYSID. In fact it is recommended that you allow IADF to allocate them in conjunction with this activity because it is much easier to do so.

The dynamic rules data bases, shown in the bottom third of the panel are named at this time, but are not allocated in this activity.

When you have finished your selections, press the END PF key.

A search is now made in the SYSTEMS table for other versions of the same SYSID in order to establish the migration paths for the current SYSID/VERSION.

- If no versions are found, the next screen you see is Figure 5-7.
- If multiple versions have already been established, you see Figure 5-6.
- If this is the second version of the same SYSID, Figure 5-5 is presented as a warning, in case you did not intend to establish multiple versions of the current SYSID.

```

ADFVRV3----- SYSID ALREADY DEFINED -----
COMMAND ==>
USERID   - TS0ID1                                TIME - 12:36
*****
*
*
*  SYSID:  SAMP                                *****
*  VERSION: TEST                                *****
*
*
*  WARNING: This SYSID has already been defined. *****
*
*          You are establishing a Multiple-Version SYSID. *****
*          Press ENTER if you wish to continue. *****
*          You must enter next the migration path(s) between this *****
*          version and other versions of the same SYSID. *****
*
*          If you do not want a new version for the same SYSID, *****
*          press the END key to reject this definition. *****
*          Change the SYSID you are creating. *****
*
*
*****
COMPLETE AND PRESS ENTER TO CONTINUE OR END, CANCEL TO EXIT

```

Figure 5-5. SYSID Version Warning Panel

When you press ENTER on Figure 5-5, a search is made for duplicate data set names for libraries involved in migration (entered in Figure 5-4).

- Rules Source (&DSN1)
- Rules Load (&DSN2)
- Dynamic Source (&DSN3)
- ISPF tables (&DSN8)

In addition the DDNAME for the "ISPF Tables" data set (&DSN8) must be different from DDNAMES of other versions of the same SYSID.

If the new SYSID/VERSION is accepted, you see Figure 5-6.

```

ADFVRV4 ----- MIGRATION PATHS FOR SAMP  BASE -----
COMMAND ==>                                SCROLL ==> PAGE

Available command: Cancel
Migrate SYSID: SAMP

Instructions:
Enter Y to define a path between these two versions
Enter N to prevent a path between these two versions

From TES1 to TEST ==> n (Y|N) From TEST to TES1 ==> y (Y|N)
From TES1 to BASE ==> n (Y|N) From BASE to TES1 ==> y (Y|N)

***** BOTTOM OF DATA *****

```

Figure 5-6. Migration Paths

Figure 5-6 displays all the versions available and allows you to enter "Y" or "N" for each of the rows, establishing the path(s) to be available when migrating from this SYSID/VERSION. Be careful in your path selections, partly to avoid overlaying versions. In addition, this panel is automatically presented to you **only on initialization**.

If you enter paths incorrectly, an Administrator of Level 3 can change them by issuing the PATHC IADF command (described in "SYSID/Migration Path Maintenance" on page 5-16). PATHC must also be used when the other SYSID commands are invoked: Delete, Reinitialize, and Update.

The paths entered are stored in the table ADFVER, which is used for migration.


```

ADFPR04 ----- SYSTEMS WITHIN MFC1 -----request queued
COMMAND ==>                                SCROLL ==> PAGE

Line Commands: D delete I initialize R reinitialize U update
CMD SYSID VERS DDNAME Data Set Name Description
-----
' SAMP BASE SAMP TS0ID1.TEST.ISPFTBLS IMSADF SAMPLE
' BANK TEST BANK TS0ID1.TEST2.ISPFTBLS BANK SYSTEM
' ABCD BASE ABCD TS0ID1.TEST3.ISPFTBLS ANOTHER ONE

***** BOTTOM OF DATA *****

```

Figure 5-7. Display SYSID within ADFID

If you are Level 3 and you have multiple SYSIDS installed, your SYSID is displayed in the list shown in Figure 5-7 and "request queued" in the upper right hand side of the panel.

Press the END PF key.

The initialization process begins with one of several panels:

- Figure 5-8 is shown if you **have not** preallocated your "ISPF tables" data set.
- Figure 5-9 is displayed if you **have** preallocated your "ISPF tables" data set, but not your other libraries.
- Figure 5-4 is displayed with a message indicating that initialization is taking place if all your named libraries are preallocated.

Press ENTER.

```

ADFT001 ----- ERROR DETECTED ----- Allocation error
COMMAND ==>

USERID   - TS0ID1                                TIME    - 12:00
*****
*
* Unexpected Return Code from TSO COMMAND          ***
* Control Information:ALLOCATE LIBRARY: TS0ID1.TEST.ISPFTBLS ***
*
* The process invoked previously WILL NOT be completed, but ***
* a recovery will be made if possible.             ***
*
* Message Returned:                               ***
*   ADFC010                                         ***
*
* Return code:                                     ***
*   12                                              ***
*
* Function:                                         ***
*   ADFCREAT                                        ***
*****
PRESS ENTER OR END TO CONTINUE

```

Figure 5-8. Allocation Error Information Panel

This panel appears if you do not have your "ISPF Tables" library preallocated. This condition is normally not an error, for when you press END or ENTER, you see Figure 5-9.

Allocate SYSID Libraries

Figure 5-9 displays the libraries you just entered. The data bases are not allocated on this panel.

```

ADFIN0L ----- CREATE IMSADF II  LIBRARIES -----
COMMAND ==> a

  A Allocate library
  B Browse library                      (execution type must be online)
  D Delete library
  X Exclude library from global command
  If the command is entered at the top, applies to ALL data sets (GLOBAL)

Execution type ==> 0                      (0=online, B=batch)
Unit           ==>                      (3330/3350/3380. For Batch allocation)

  CMD  DATA SET NAME                TYPE          Space in  Block  Volume
                   Blocks            size
  '    TS0ID1.TEST.RULSRC            RULES source  30        6080   ADFVOL
  '    TS0ID1.TEST.RULLIB            RULES load   30        19069  ADFVOL
  '    TS0ID1.TEST.DYNSRC            Dynamic source 30        6080   ADFVOL
  '    TS0ID1.TEST.ADFIMG            IMAGE source  30        6080   ADFVOL
  x    TS0ID1.TEST.RULLIB            Program Load
  x                                     COMPOSITE
  '    TS0ID1.TEST.ISPFTBLS          ISPF tables  30        3120   ADFVOL

PRESS ENTER IF MORE FUNCTIONS ARE REQUIRED OR END TO PROCESS AND EXIT

```

Figure 5-9. Library Allocation Panel

Space allocations are blank if you have never added a SYSID or may contain residual values for another SYSID (to reduce data entry). Allocations shown in this figure are examples only, not recommendations. Space is designated for primary allocation only.

Directory blocks are fixed at 20; if that is not adequate for your location, you may change the IMSADF.ADFCLIB(ADFINIT) as described in “Clists” on page 2-11.

Secondary allocation is approximately 3/4 of the primary space; If that is not adequate for your location, you may change the IMSADF.ADFCLIB(ADFINIT) clist as shown in “Clists” on page 2-11. Enter the block size appropriate to the “TYPE” of library:

- Source--should be a multiple of 80.
- Load--should be a maximum block size for a load module library for your location.

You should make these compatible with the IMSADF II installation libraries, since your libraries are concatenated first in some instances and should have a large enough block size to avoid errors.

- Composite--is a load module library if you use Composite Rules load modules. If you do not, place an “x” in its CMD column.

Enter the volumes on which you want the libraries allocated or leave blank to take the TSO default. In this example ADFVOL was used for all of them.

Enter "A" on the command line to allocate the libraries.

Exclude from the allocation list those libraries which are previously allocated. You may share data sets among SYSIDS, so any or all of them might exist in this case. Another type of library which must be excluded is a data set which has already been named in this list. Exclude them by entering an "x" in the command column (CMD) of the list.

Press ENTER.

ADFIN0L ----- CREATE IMSADF II LIBRARIES -----Function Complete
 COMMAND ==>

A Allocate library
 B Browse library (execution type must be online)
 D Delete library
 X Exclude library from global command
 If the command is entered at the top, applies to ALL data sets (GLOBAL)

Execution type ==> 0 (0=online, B=batch)
 Unit ==> (3330/3350/3380. For Batch allocation)

CMD	DATA SET NAME	TYPE	Space in Blocks	Block size	Volume
*	TSOID1.TEST.RULSRC	RULES source	30	6080	ADFVOL
*	TSOID1.TEST.RULLIB	RULES load	30	19069	ADFVOL
*	TSOID1.TEST.DYNSRC	Dynamic source	30	6080	ADFVOL
*	TSOID1.TEST.ADFIMG	IMAGE source	30	6080	ADFVOL
X	TSOID1.TEST.RULLIB	Program Load			
X		COMPOSITE			
*	TSOID1.TEST.ISPFTBLS	ISPF tables	30	3120	ADFVOL

PRESS ENTER IF MORE FUNCTIONS ARE REQUIRED OR END TO PROCESS AND EXIT

Figure 5-10. Successful SYSID Libraries Allocation

When this panel is shown again, allocation has completed. For those libraries allocated, an "*" appears in the command column. If the allocation failed, an error code appears, and more information may be obtained by pressing the HELP PF key. The libraries that were excluded by keying "x" in the CMD still have an "X" in that column. The completion message appears on the the panel.

Press the END PF key.

You see Figure 5-11.

```

IADFSYS ----- IADF SYSTEM DEFINITION MENU ---- System SAMP created
OPTION ==> 1

ADMINISTRATIVE LEVEL:3
KEY ==>          SYSID ==> SAMP  LEVEL: 3

  1 LIBRARIES      - Create and initialize the libraries which
                    comprise an IMSADF II  system

  2 DEVELOPERS     - Specify TSO userids of developers who have
                    access to an IMSADF II  system

  3 STATIC DSNS    - Specify data sets for generation of data layouts,panels,
                    transactions, drivers  TYPE==> F (F=Foreground B=Batch)

  4 DYNAMIC DSNS   - Specify data sets for generation of audits, messages,
                    help panels, routes, secondary destinations, userids,
                    and profiles          TYPE==> F (F=Foreground B=Batch)

  5 BTS DSNS       - Specify data sets for BTS execution

  6 INITIALIZE     - Initialize IMSADF II  libraries and data bases

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 5-11. Redisplay IADF System Definition Menu with Completion Message

When initialization is complete, the IADF System Definition Menu (IADFSYS) is displayed with the message "SYSTEM SAMP CREATED". The reason for "Level: 3" is that, having added the SYSID, you were automatically added as a Level 3 Developer.

Press the RETURN PF key to return to the IADF Administration Main Menu.

Table Initialization

Every time a SYSID is "Initialized", a set of tables is created by ADFCREAT clist. These tables are required because they contain structural information for the system. The tables and their contents are:

ALL	Segment ID and type for all the segments under IADF
CROS	Cross references between segments and transactions
HLP	IMSADF II help panels for online transactions.
MSG	Messages for the Message Data Base
PFL	Profiles for the Signon Profile data base
PR	List of the IADF Development users of the system, one of them being the TSOID of the creator of the SYSID. The level is also stored.
ROU	Alternate Routes
SDT	Secondary Destinations
SYS	Signon and Primary Option Menu data
USR	Users for the Profile Data Base

Refer to Appendix B, "IADF Naming Conventions" for the format of the full table name.

During the initialization process, the tables are created in the "ISPF Tables" library, in this case, TSOID1.TEST.ISPFTBLS.

Note: This data set does not have to be in the TSO LOGON procedure or in the invoking clists for IADF/ISPF for the designated developers. It is allocated by IADF during the development process. During the initialization process, the message at the bottom of the panel changes to say that the system is being initialized, and the keyboard is locked.

Deleting a SYSID

When you delete a SYSID, there is a definite order in which it should be performed. Assume you created a system and allocated data sets for use by a development group, but then realized that there were errors severe enough that you did not want to reinitialize or update it, but just wanted to delete it.

The preferred method is discussed first:

1. From the System Definition Menu, select LIBRARIES. Enter the SYSID in the "KEY" field.
2. When you see the panel describing the SYSID (as in Figure 5-4), enter "D", then press the END key to delete the system.

```
ADFDL01 ----- C O N F I R M   D E L E T I O N -----  
COMMAND ===>  
  
A System is going to be deleted, its id is:  
  
SAMP  
  
Date Created:  10/08/85  
Time Created:  12:00:00  
By:            TSOID1  
DESCRIPTION:  
  
Press ENTER key to CONFIRM DELETE  
Press END key to CANCEL DELETE.
```

Figure 5-12. SYSID Deletion Panel

You must confirm the deletion by pressing ENTER for the SYSID to be deleted. Subsequently you are shown the Systems Definition Menu with the SYSTEM SAMP DELETED completion message.

If you do not share libraries and data bases with another SYSID, you delete them as follows:

1. From the System Definition Menu, select INITIALIZE.
2. On the INITIALIZATION menu, enter the SYSID and select LIBRARIES.

3. On the library panel (Figure 5-9), enter "D" on the command line or beside each data set which you wish to delete.
4. The requested data sets are deleted.
5. Do the same for data bases, if required.

It is not recommended that you delete the data sets before the SYSID.

First, all data sets would be deleted except the "ISPF Tables", since it is open for an existing SYSID when you are on the Libraries/Data Bases panels.

Next, when you tried to delete the SYSID, you would have an allocation error, because the "Rules Source" and "Dynamic Source" data sets are allocated during that time; and they were deleted during the first step. Nevertheless, you are shown Figure 5-13.

ADFDL01 ----- C O N F I R M D E L E T I O N -----
 COMMAND ===>

The system to be deleted was not Initialized properly:

SAMP

Date Created: 10/08/85

Time Created: 12:00:00

By: TS0ID1

DESCRIPTION:

Press **ENTER** key to **CONFIRM** DELETE

Press **END** key to **CANCEL** DELETE.

Figure 5-13. SYSID Deletion Panel

When you see this panel and ENTER, the SYSID will be deleted. You may then use ISPF to delete the "ISPF Tables" data set if you wish. If a SYSID has multiple versions, its paths do not have to be deleted (see "SYSID/Migration Path Maintenance" on page 5-16).

SYSID/Migration Path Maintenance

This utility is called as a command, PATHC, from any IADF panel. It is available to Level 3 Administrators. The purpose of this utility is to enable you to change migration paths if it is necessary. You must use it if you wish to change paths as a result of SYSID reinitialization or update. If you delete the SYSID, you are not required to delete the paths, but you may if you wish.

Migration paths are maintained in table ADFVER (in IMSADF.ADFTLIB). The key is

ADFID SYSID fromversion toversion

After authorization verification occurs, Figure 5-14 is displayed to request the SYSID whose paths are to be modified.

ADFVRV1----- MIGRATION PATHS MAINTENANCE UTILITY -----

COMMAND ==>

ADFID:

MFC1

Administration Level: 3

SYSTEM ID ==>

SAMP

Instructions:

If a SYSID is entered, its migration paths are scanned.
If any are found, they are presented to you.
You may **Add**, **Change**, or **Delete** paths.

PRESS **END** TO EXIT OR **ENTER** TO PROCEED

Figure 5-14. Migration Paths Maintenance

Type the SYSID and press ENTER.

If the SYSID contains path information, Figure 5-15 is displayed.

```
ADFVRV2----- VERSION MAPPINGS FOR A SYSID -----
COMMAND ==>                                     SCROLL ==> PAGE

Available command: C Cancel

System ID:    SAMP

Line Commands: D Delete, I Insert

C      FROM      TO      CREATED      BY
-      - - - - - - - - - -
'      BASE      TEST     09/22/86 19:50:36 TS0ID1
'      TEST      PROF     09/18/86 20:06:45 TS0ID1
'      TEST      TES1     09/23/86 13:32:45 TS0ID1
'      BASE      TES1     09/23/86 13:32:48 TS0ID1
***** BOTTOM OF DATA *****
```

Figure 5-15. Version Mappings Maintenance Utility

The panel allows you to delete, insert, or change the defined paths. (A change is actually a delete followed by an insert to the table).

Note: All maintenance is performed on a storage copy of the table. When you exit from this activity, ADFVER is updated.

Basic path checking is provided to avoid accidental migrations to another level. If the SYSID does not contain any versions or no path information exists, a message is displayed.

Migration to other SYSIDS is not supported.

Chapter 6. Adding Developers to the SYSID

You are at the point in your administrative duties that is highlighted in the following list:

1. Add Administrators to this ADFID.
2. Add the Application System (SYSID).
3. **Add Developers to the Application System (SYSID).**
4. Create Static Rule (RULEGEN) data sets.
5. Create Dynamic data sets.
6. Create BTS data sets.
7. Allocate the Application System (SYSID) Data Bases.

Create IADF Developers Within IMSADF II System Table

This chapter describes how you may add users with definite access levels to the SYSID you created in Chapter 5, “Installing an Application SYSID.” The ISPF/IADF table that is updated is “ssssPRad” where “ad” is the first two characters of ADFID, and “ssss” is SYSID; each Developer is a row in the table. The table is stored in the “ISPF Tables” data set for each SYSID.

This series of panels is available for use **ONLY** by these groups:

- Level 3 Administrators (ADFID users)
- Level 2 Administrators (SYSID users)

The term “users” needs to be redefined at this point. Prior to this chapter, all users were administrators, who functioned at a global level.

The users to be added by the Administrators are developers. Each individual authorized in any way for SYSID activities, **including migration**, must be named either specifically or generically. They must be named with the specific authorities they are to receive, or they will be able only to browse and perform rules generations on existing SYSIDS. The levels and their authorities are listed in order of decreasing function:

Level 3	Able to Read, Modify, and Delete all activities pertaining to his SYSID. This level must be specifically named as a Developer. The one who installs the SYSID is automatically added as a level 3 Developer.
Level 2	Read/Write access to all SYSID activities. This level must be named as a Developer.
Level 1	Read access only for all SYSID activities. This level is normally not named and is comprised of all remaining Developers. All Level 1 Developers can invoke the Rules Generator on behalf of existing IADF components. A Level 1 Developer may also perform migration, but must be registered.

Note: If you establish a "Master Rules" SYSID, all developers (who would be Level 1) may then access elements and perform generations, while integrity is assured.

If you are an administrator and **you** wish some level of authority over this SYSID, be sure you are named, since your administrative level of authority terminates at the SYSID level.

Migration Authority

The Developers table contains two columns which apply to migration, one for migration **FROM** this SYSID, the other for migration **TO** this SYSID. There are four possible values:

Blank No migration is allowed for this developer (default).

M Move form of migration allowed.

C Copy form of migration allowed.

B Both forms allowed.

"Generic" users are supported in the Developers table, so that each individual who may perform migration need not be named.

#(ADM) Grants access for migration to all Administrators with the ability (not a requirement) to specify a level number, such that only Administrators with that level are allowed access.

#(DEV) Is used to grant access to any developer whose level matches the one specified at the "Level" column. Access must be specified for this generic profile (for example, only Level 3 Developers might be permitted migration privileges).

Generic Administrative users are automatically established by IADF when the Developers table is initialized, but may be deleted to restrict access if so desired. A generic profile may also specify the FROM or TO columns to reduce the type of function allowed.

The columns allow you the maximum amount of flexibility in change control. You may desire to "move" applications to the production system, but to "copy" them to a maintenance system (movement would delete them from production, resulting in interrupted service). You may limit migration to Administrators only, in which case the generic Administrator is all that is required; but for a sensitive application, you may desire to have only one or two individuals responsible for migration and not to use the generic identifier. You may wish **any** developer to copy applications **from** production, but only your change control team to perform migration **to** production.

During migration, the source SYSID must pass verification that both the requester and the migration form match in the Developers table under the "FROM" column; otherwise the function is not performed. If the user is not found in the table, migration is not allowed.

Once the "FROM" SYSID validation is completed, verification of the same table in the target system is performed, where the "TO" column for this user must pass the same criteria for both user and form of migration. If **both** are not true, migration is denied.

Note: If only one version of a SYSID exists, there is **no authorization verification performed on the "TO" system.**

The migration authority scheme described in this section was developed for multiple version SYSIDS. Implementation of multiple version SYSIDS is spread across these tasks:

1. Initialize SYSID (“Add the Application System” on page 5-2)
2. Add Developers (Chapter 6, “Adding Developers to the SYSID”)
3. Migration (Chapter 9, “Migration Services”)

For current IADF users: If you did not convert any existing Developers tables to multiple version format at the same time you did the SYSTEMS table (described in Chapter 5, “Installing an Application SYSID”), follow instructions found in “Converting Tables to Versioning” on page A-6 **before** continuing further in this task.

```
IADFADMN ----- IADF ADMINISTRATION MAIN MENU -----
OPTION ==> 1

ADMINISTRATIVE LEVEL:3
KEY ==>          SYSID ==>          LEVEL:

  1 SYSTEMS      - Create or modify IMSADF II  systems (SYSID)
  2 MIGRATION    - Perform migration  TYPE ==> M ( M Move  C Copy )
  3 ADMINISTRATORS - Specify administrative users
  4 TABLES      - Create, modify, and/or copy ISPF tables
  5 GLOSSARY     - Update glossary terms
  6 DEVELOPMENT  - Perform IMSADF II  development tasks
  X EXIT         - Terminate IADF

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT
```

Figure 6-1. Select SYSTEMS from the IADF Administration Main Menu

From the IADF Administration Main Menu enter 1 on the option line to select the System Definition Menu.

```

IADFSYS ----- IADF SYSTEM DEFINITION MENU -----
OPTION ==> 2

ADMINISTRATIVE LEVEL:3
KEY ==>          SYSID ==> SAMP LEVEL: 3

  1 LIBRARIES      - Create and initialize the libraries which
                    comprise an IMSADF II system

  2 DEVELOPERS     - Specify TSO userids of developers who have
                    access to an IMSADF II system

  3 STATIC DSNS    - Specify data sets for generation of data layouts,panels,
                    transactions, drivers TYPE==> F (F=Foreground B=Batch)

  4 DYNAMIC DSNS   - Specify data sets for generation of audits, messages,
                    help panels, routes, secondary destinations, userids,
                    and profiles TYPE==> F (F=Foreground B=Batch)

  5 BTS DSNS       - Specify data sets for BTS execution

  6 INITIALIZE     - Initialize IMSADF II libraries and data bases

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 6-2. Selecting DEVELOPERS Option from IADF System Definition Menu

Enter 2 on the option line of the System Definition Menu to select the USERS panel.

Note: After this function has finished executing, your SYSID level is displayed on all IADF panels containing "LEVEL:". The first action during an IADF session that is directly related to a given system triggers the display of Developer level, since Developer level is defined on a SYSID basis. Options 3-5 on the System Definition menu also trigger the display of Developer level.

```

ADFPR07----- DEVELOPERS IN SAMP BASE -----
COMMAND ==> i tsoid2 3                                SCROLL ==> PAGE

Line Commands: D delete I insert
CMD USERID   Level  Exper  FROM  TO    Description (optional)
'   TSOID1      3      A      B      B      IADF Administrator
'   #(ADM)              A      B      B      ALLOW ANY ADMINISTRATOR
***** BOTTOM OF DATA *****

```

Figure 6-3. Developers within SYSID Panel

The Administrator who initialized the SYSID is automatically added as a Level 3 Developer and is permitted both forms of migration in the "FROM" and "TO" SYSID/VERSIONs. In addition the generic Administrator was added. The blank in the "Level" column indicates that any Administrator may perform migration for the SYSID/VERSION. Migration is the **only** authority granted by the generic profile.

To add another Developer for SAMP, enter "I" in the command line followed by the TSO userid and the access level desired. (You can also type over one of the existing rows using the "I" line command).

```

ADFPR07----- DEVELOPERS IN SAMP BASE --- User TSOID2 added
COMMAND ==> i #(DEV) 3                                SCROLL ==> PAGE

Line Commands: D delete I insert
CMD USERID   Level  Exper  FROM  TO    Description (optional)
'   TSOID2      2      A      C      C      PROJECT MANAGER
'   TSOID1      3      A      B      B      IADF Administrator
'   #(ADM)              A      B      B      ALLOW ANY ADMINISTRATOR
***** BOTTOM OF DATA *****

```

Figure 6-4. Adding a Developer of a SYSID

Figure 6-4 shows the Developer added with defaults and a completion message. You may key over the values you wish and add a description. "Exper" denotes experience level of the Developer. Enter N(ovice) or A(dvanced). Enter the permitted migration forms for the "FROM" and "TO" versions:

Blank No migration is allowed for this developer (default).
M Move form of migration allowed.
C Copy form of migration allowed.
B Both forms allowed.

You may add the optional generic Developer userid as is shown on the command line.

```

ADFPR07----- DEVELOPERS IN SAMP BASE --- User #(DEV) added
COMMAND ===>                                SCROLL ===> PAGE

Line Commands: D delete I insert
CMD USERID   Level  Exper  FROM  TO    Description (optional)
'  #(DEV)      3      A      C      Allow Level 3 Developer
'  TS0ID2      2      A      C      C    PROJECT MANAGER
'  TS0ID1      3      A      B      B    IADF Administrator
'  #(ADM)      A      B      B    ALLOW ANY ADMINISTRATOR
***** BOTTOM OF DATA *****

```

Figure 6-5. Developers within SYSID Panel

Figure 6-5 shows that the generic Developer userid has been added. (Values entered for the generic Developer are for illustrative purposes only). In this instance it allows any Level 3 Developer to invoke the COPY form of migration "FROM" this SYSID/VERSION. Notice that migration "TO" this SYSID/VERSION is not permitted for the generic userid.

Press the RETURN PF key to see the IADF Administration Main Menu.

Chapter 7. Establish SYSID Data Set Activities

You are at the point in your administrative duties that is highlighted in the following list:

1. Add Administrators to this ADFID.
2. Add the Application System (SYSID).
3. Add Developers to the Application System (SYSID).
- 4. Establish Static Rules data sets.**
- 5. Establish Dynamic Rules data sets** (Not applicable to DB2 users).
- 6. Establish BTS data sets** (Not applicable to CICS users).
7. Allocate Application System (SYSID) Data Bases (applicable only to DL/I access for the Dynamic rules data bases).

Introduction

This chapter describes the tables you must establish for each SYSID and some optional tables you can create. These tables are used by developers when they invoke IMSADF II programs and BTS. The services may be interactively performed by TSO, or a batch job stream may be submitted to perform the same tasks.

Your tasks may be more easily understood if a preview of this chapter is made:

- You must establish the foreground tables appropriate to your environment (see Figure 7-1 for this information).
- You may wish to establish batch tables. They are described in “Establish Optional Batch Tables” on page 7-17.
- No matter which tables you establish for a SYSID, all of them have the same format and are modified the same way, if you find it necessary. Modifications are described in “Panel Formats for Data Set Activities” on page 7-3.
- To help you consolidate this material, “Summary” on page 7-24 is included at the end of the chapter.
- You may be able to minimize your data set activities if you can use procedures described in “A Technique for Administering Multiple SYSIDS” on page 7-24.

This series of panels is available for use ONLY by these groups:

- Level 3 Administrators (ADFID users)
- Level 2 Administrators (SYSID users)

Tables Used in IMSADF II Program Invocation

The activities in this chapter create (or update) ISPF tables describing data sets required by IMSADF II programs and BTS. The data sets themselves are not used until development time, but the information about them is stored now. Each of the highlighted topics is a major topic in this chapter. The purpose of this activity is twofold:

- Establish ISPF tables (where “ssss” is the SYSID) of data sets required by IADF:

- ssssST** For Rules Generator, MFS Utility and BMS Utility, described in “Establish Foreground Static data sets for the SYSID” on page 7-7.
- ssssDY** For High Level Audit Compiler and batch driver, described in “Establish Foreground Dynamic Data sets for the SYSID” on page 7-11. (Not applicable to DB2 users)
- ssssBTS** For foreground BTS execution, described in “Establish BTS Data sets for the SYSID” on page 7-14. (Not applicable to CICS/OS/VS users)
- ssssSTB** Optional for Rules Generator, MFS Utility and BMS Utility, described in “Static Data Sets for Batch” on page 7-18.
- ssssDYB** Optional for High Level Audit Compiler and batch driver, describe in “Dynamic Data Sets for Batch” on page 7-20. (Not applicable to DB2 users)

- to allow you to modify the data sets or table if that requirement arises.

To modify the panels you see during this activity, you should be very familiar with the cataloged procedures at your location for the following:

- ????G** Rules Generator
- MFSUTL** MFS Utility
- DFHMAPS** BMS Utility
- ????A** High Level Audit Compiler
- ????B** Batch driver
- BTS** Batch Terminal Simulator (if you are an IMS/VS user and you have this product).

Also reference to the data sets you described in Figure 5-4 may be necessary.

Figure 7-1 summarizes the required and optional tables you create for each SYSID, based on the environment chosen by the installer of IMSADF II. Columns are marked Y/N/O for Yes/No/Optional. “F/G” identifies the table as foreground. The lower line in the column heading gives the table identifier, which is prefixed by the SYSID.

INSTALLATION ENVIRONMENT	F/G ST	F/G DY	F/G BTS	BATCH STB	BATCH DYB
IMS/DL/I	Y	Y	O	O	O
IMS/DB2	Y	N	N	O	N
CICS/DL/I	Y	Y	N	O	O
CICS/DB2	Y	N	N	O	N

Figure 7-1. Installation Options and SYSID Tables

Selection

All activities in this chapter begin with selection of the System Definition menu from the IADF Administration Main Menu.

```
IADFADMN ----- IADF ADMINISTRATION MAIN MENU -----
OPTION ==> 1

ADMINISTRATIVE LEVEL:3
KEY ==>                      SYSID ==> SAMP    LEVEL: 3

  1 SYSTEMS      - Create or modify IMSADF II  systems (SYSID)
  2 MIGRATION    - Perform migration    TYPE ==> M ( M Move  C Copy )
  3 ADMINISTRATORS - Specify administrative users
  4 TABLES      - Create, modify, and/or copy ISPF tables
  5 GLOSSARY     - Update glossary terms
  6 DEVELOPMENT  - Perform IMSADF II  development tasks
  X EXIT        - Terminate IADF

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT
```

Figure 7-2. Select SYSTEMS from the IADF Administration Main Menu

From the IADF Administration Main Menu, enter 1 on the option line to select the System Definition menu.

If the SYSID SAMP was used before, its identifier is retained. If you want to change SYSIDs, enter the new value. Notice that your Developer level for this SYSID (not your administrative level) is displayed after "LEVEL:", near the top of the panel to the right of the SYSID.

Panel Formats for Data Set Activities

The panels presenting detailed lists of data sets (see Figure 7-6, Figure 7-9, and Figure 7-11) are identical for the highlighted activities and are explained together since they apply to all. Symbolic names (preceded by an ampersand) are displayed for data sets which will be resolved at execution time. Fixed names may be substituted. The data sets of the form &DSNn, where "n" is a number, may be the same ones you entered in Figure 5-4. Others are temporarily allocated as needed by IADF. The number corresponds to the number in parentheses on the original panel. The node values for ADFNODE/NEWADF/IMSNODE/IMSTEST are found in your user profile. If data set names are not enclosed in quotes, the TSO userid is prefixed.

All data sets used are listed. You normally have no need to modify any of the "work" or SYSOUT data sets, but they are presented if the occasion should arise.

Review the defaults to ensure that they are appropriate. Those values that you wish to change, may be modified by keying the new values over the original ones.

Scroll **DOWN** to view all the data sets (depending on the terminal type, they may split in different places).

Command Usage

With the commands for insert, update, and delete entered under the "C" column of the appropriate data set, it is possible to alter the ISPF SYSID table containing this information.

Update To change "work" data set allocation parameters, enter "U" on the "C" column beside the desired DDNAME.

```
GENUPDS ----- UPDATE DATA SETS -----
COMMAND ==>

DDNAME:  SYSUT1

DSNAME      ==>
Disposition ==> SHR                      (SHR,OLD,MOD)
Member       ==>
Description  ==> SYSUT1
STATUS       ==> P                      (P = Protected)
MBR          ==> N                      (N = NO member name allowed)
Allocate     ==> A                      (A = allocate new)
ATTR         ==>                      (Attribute name to use)
Type         ==>                      (CYL for cylinder, TRACKS or Blank)
Size         ==> 1700                  (Space in above Units)
DIR          ==>                      (Number of Directory blocks)
Allow Copy   ==>                      (N = Row cannot be replaced by Copy)
SYSOUT       ==>                      (Use for Batch SYSOUT data sets)
PARMS        ==>

COMPLETE AND PRESS ENTER OR END TO UPDATE OR ENTER CANCEL TO EXIT
```

Figure 7-3. Data Set Allocation

You are shown Figure 7-3 with the data set information on it. In this example, it is SYSUT1. A data set is assumed to be permanent, unless it is a "work" data set as shown in the panel, in which case the current space allocation is given along with an "A", meaning that the program requiring it allocates it "NEW" at the appropriate time. You can use the allocation panel to increase the space allocation if you run out of space on a work data set.

Notes:

1. Do not key an "A" in the "Allocate" column of a permanent data set. The data set will be deleted at execution time and an attempt made to reallocate it. Your data will be lost, and the allocation will fail.
2. To eliminate the allocation for temporary data sets which are not used in your environment, change the "A" beside "Allocate" to a blank. This both reduces the DDNAMES which are allocated during your session and also saves time during data set allocation. The procedure should only be used when temporary data sets are not required in your environment. The description of each table tells you which data sets may be candidates.

The "Allow Copy" parameter pertains to the copying of foreground tables to batch. This optional function may not be desired in your installation, so it is described in "Copy Function" on page 7-22. "N" means that this data set is not eligible for copying, while a blank means that it may be copied. Batch tables are described in "Establish Optional Batch Tables" on page 7-17.

The SYSOUT line appears only if this panel refers to a data set in a batch table. You may place a SYSOUT class in this parameter. The distributed value is an asterisk(*).

You may also update the PARMS column if the DDNAME is one that invokes a program. You can recognize appropriate DDNAME in the data set panels by the presence of a name under the MEMBER column. Another reason to update PARMS occurs when the DDNAME has special JCL requirements that you want to add for batch tables.

GENUPDS ----- UPDATE DATA SETS -----
COMMAND ==>

DDNAME: RULESMOD

DSNAME	====>	'&ADFNODE..ADFLOAD'	
Disposition	====>	SHR	(SHR,OLD,MOD)
Member	====>	MFC1DFR2	
Description	====>	RULES GENERATOR	
STATUS	====>	P	(P = Protected)
MBR	====>		(N = NO member name allowed)
Allocate	====>		(A = allocate new)
ATTR	====>		(Attribute name to use)
Type	====>		(CYL for cylinder, TRACKS or Blank)
Size	====>	1700	(Space in above Units)
DIR	====>		(Number of Directory blocks)
Allow Copy	====>		(N = Row cannot be replaced by Copy)
SYSOUT	====>		(Use for Batch SYSOUT data sets)
PARMS	====>	&ADFID.G01,500,15	

COMPLETE AND PRESS **ENTER** OR **END** TO UPDATE OR ENTER **CANCEL** TO EXIT

Figure 7-4. Updating PARM Example

When you first see Figure 7-4, the initial parameters are displayed. Where the desired program is one supplied with IMSADF II (such as the Rules Generator in the figure), the program name itself is customized, containing the ADFID as the first four characters, and is the first parameter in the PARMS column. The parameters passed to an invoking program are positional and begin with the second value in the PARMS column. In that situation, you **must not** update the first parameter, but may key over values beginning with the second parameter value beside the PARMS column. In the example, you could change parameters passed to the Rules Generator, beginning with value "500" if you wished.

Notes:

1. Because the searching of a STEPLIB within a clist is not possible under TSO, a program is called to attach the task invoking the requested function (i.e., the Rules Generator). The task itself may have a named DD statement to serve as a STEPLIB. Parameters must be passed through the attaching task to the actual program to be invoked.
2. IADF PARMS are supplied for programs which invoke the IMS/VS batch region controller, DFSRRC00, at the IMS/VS 1.2 release level. You may have to change the supplied values if your release is 1.3 or greater or if you have parameters pertaining to associated products, such as DBRC or the IRLM.

This consideration applies to the tables for Dynamic rules and BTS.

Insert "I" may be used to add another data set to your TASKLIB or other library concatenation.

Note: TASKLIB is used as STEPLIB when you invoke IMSADF II programs interactively.

A DDNAME may also be inserted, though it is not recommended. Adding a DDNAME and data set(s) increases the number of data sets which must be allocated at execution time. In addition, compatibility with future releases of IMSADF II may be affected, since a DDNAME added by the user is not supported by the IADF table conversion procedures.

Delete Use of "D" should be reserved for those data sets which you have previously inserted. The rest of the data sets are required for proper execution of the program.

Establish Foreground Static data sets for the SYSID

This option is used to define the required data sets for interactive static rule generation.

```
IADFSYS ----- IADF SYSTEM DEFINITION MENU -----
OPTION ==> 3

ADMINISTRATIVE LEVEL:3
KEY ==>                SYSID ==> SAMP  LEVEL: 3

  1 LIBRARIES      - Create and initialize the libraries which
                    comprise an IMSADF II  system

  2 DEVELOPERS     - Specify TSO userids of developers who have
                    access to an IMSADF II  system

  3 STATIC DSNS    - Specify data sets for generation of data layouts,panels,
                    transactions, drivers  TYPE==> F (F=Foreground B=Batch)

  4 DYNAMIC DSNS   - Specify data sets for generation of audits, messages,
                    help panels, routes, secondary destinations, userids,
                    and profiles          TYPE==> F (F=Foreground B=Batch)

  5 BTS DSNS       - Specify data sets for BTS execution

  6 INITIALIZE     - Initialize IMSADF II  libraries and data bases

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT
```

Figure 7-5. Selecting STATIC DSNS Option from IADF System Definition Menu

Enter 3 on the option line to select STATIC DSNS, making sure "f" is the value beside "TYPE". Under normal conditions you do not need to modify the list of data sets which you are presented.

To modify the panels you see during this activity, you should be very familiar with the cataloged procedures for the Rules Generator and screen generation utilities at your location.

RULPNLS ----- RULEGEN SYSTEM DATA SETS -----
 COMMAND ==> SCROLL ==> PAGE

Line Commands: D Delete I Insert U Update

C	DDNAME	DSNAME	MEMBER	Description
'	TASKLIB	'&ADFNODE..ADFLOAD'		STEPLIB FOR SYSTEM
'		'&DSN2'		USER RULES LOAD
'		'&NEWADF..RULLIB'		ADF RULES
'	MACLIB	'SYS1.MACLIB'		MACLIB LIBRARIES
'		'&ADFNODE..ADFMAC'		ADF MACLIB
'	PGMLOAD	'&DSN5'		PROGRAM LOAD LIBRARY
'	SYSLIB	'&DSN2'		USER RULES LOAD
'		'&DSN30'		USER EXITS
'		'&NEWADF..RULLIB'		ADF RULES
'		'&ADFNODE..ADFLOAD'		EXITS & ADFID CSECT
'	IMAGELIB	'&DSN4'		IMAGE LIBRARY
'	ADFLIB	'&DSN1'		RULES SOURCE
'	GENLIST	ADF.GENLIST		GENLIST DATA SET
'	SCREENS	ADF.SCREENS		MFS SOURCE
'	CICSMAPS	ADF.CICSMAPS		BMS SOURCE
'	CICSPPT	ADF.CICSPPT		CICS PPT
'	RULEASM			RULEASM
'	RULELEL	ADF.RULELEL		RULELEL
'	SYSUT1			SYSUT1
'	SYSUT2			SYSUT2
'	SYSUT3			SYSUT3
'	SYSLIN	ADF.SYSLIN		SYSLIN
'	SYSUDUMP	'NULLFILE'		SYSUDUMP
'	ASMLIST	ASMLIST		ASMLIST
'	GENDECK	'NULLFILE'		
'	SYSPRINT	ADF.LIST		LIST DATA SET
'	ISPF			TABLE BUILD
'	DDADFX			ADFIN
'	ADFSQLHO			SOURCE FOR TABLE HANDLER
'	ADFSQLHW			TABLE HANDLER WORK FILE
'	DBRMLIB		SSSSXX	STORE DB2 DBRM
'	DB2PRINT			SYSPRINT FOR DB2
'	DB2TERM			SYSTEM FILE
'	SYSLMOD	'&DSN2'		SYSLMOD

Figure 7-6 (Part 1 of 2). Foreground Static Rules Data Sets

```

' SYSLINK  ADF.SYSLIN                                SYSLINK
'          ADF.RULELEL                                RULELEL
' RULESMOD '&ADFNODE..ADFLOAD'                        MFC1DFR2  RULES GENERATOR
' REFIN    '&IMSNODE..REFERAL'                        REFERRAL LIBRARY'
' REFOUT   REFOUT                                     REFOUT
' REFRD    REFOUT
'          '&IMSNODE..REFERAL'
' SYSTEXT
' SYSPRINT SYSPRIN1
' SYSUDUMP 'NULLFILE'
' SEQBLKS  SEQBLKS
' SYSIN    ADF.SCREENS                                FROM RULEGEN
' DFSUPA   '&IMSNODE..RESLIB'                        DFSUPAA0  MFS STEP 1
' UTPRINT  UTPRINT
' SEQBLKS  SEQBLKS
' FORMAT   '&DSN32'                                    FORMAT LIBRARY
' DFSUNU   '&IMSNODE..RESLIB'                        DFSUNUB0  MFS STEP 2
' BMS1LIST ADF.BMS1LIST                                ASSEMBLER LIST FOR BMS
' BMSMAC   '&CICNODE..MACLIB'
'          'SYS1.MACLIB'
' MAPASM   ADF.MAPASM
' BMS2LIST ADF.BMS2LIST                                LINK-EDIT LIST FOR BMS
' MAPLIB   '&CICNODE..LOADLIB'                        MAP LIBRARY
***** BOTTOM OF DATA *****

```

Figure 7-6 (Part 2 of 2). Foreground Static Rules Data Sets

The panels for the Rules Generator data sets are displayed. If you use the Data Dictionary, DB2, or wish to extract ISPF data, you may wish to insert data set names in those DDNAMES (refer to the *IMS Application Development Facility II Version 2 Release 2 Application Development Reference* for a description of the Rules Generator DDNAMES).

Review the defaults to ensure that they are appropriate.

In order to create the table initially, you need only scroll through the panels, pressing the END PF key to process the changes. Some values you should be aware of are:

1. The table for the Rules Generator itself is not environment sensitive, but contains all DDNAMES/data sets possible. If you wish to reduce the number of DDNAMES allocated (for TSO performance reasons as well), you may eliminate those not used in your environment:
 - For IMS/VS users, press ERASE EOF beside the DSNAMES for CICSMAPI and CICSPT and **in addition** enter "u" beside them to update the parameters. Change the "A" beside the "Allocate" row to a blank, which will prevent the allocation.
 - If you do **not** use DB2, enter "u" beside the DDNAMES ADFSQ LHO, ADFSQ LHW, DBRMLIB, DB2PRINT, and DB2TERM to update the parameters. Change the "A" beside the "Allocate" row to a blank, which will prevent the allocation.

2. There is table information for both the IMS/VS MFS Utility and CICS Map generation. It is not necessary for you to modify it, for IADF selects the proper environment based on your IMSADF II installation options or the SYSTEM statement in the Rules Generator input.
3. &DSN30 represents an optional library where the developer may place exits. The name is entered on a panel by the developer (the only condition is that it be preallocated). You need do nothing about this data set; if it exists at development time, it is placed in the SYSLIB concatenation; if it does not, it is ignored.
4. For IMS/VS users &DSN32 is the symbolic name of a dynamically allocated format library which receives output from the MFS Utility. Prior to invoking programs which create formats, you enter a data set name on the appropriate panels. The value of &DSN32 is changed based on the data set name you enter.

If you always want formats to go to the same library, you may replace &DSN32 at this time with a permanent data set name, and your developers will not have to enter a format data set each time they execute the utility. Some flexibility is removed, because they will not be able to direct formats to private libraries.

5. For IMS/VS users the MFS Utility programs DFSUPAA0 and DFSUNUB0 have an IMS/VS parameter, DEVCHAR, passed to them upon invocation. It is the suffix of the IMS/VS device characteristics table to be used for the format generation. If the value at your installation is not 0, you must update the parameter lists passed to both programs for your value. To do this, enter "U" beside the DDNAME DFSUPA (refer to "Update" on page 7-4 for an discussion of the update process) and key in the parameters that DFSUPAA0 expects to receive at invocation. Refer to the MFSUTL cataloged procedure at your installation for the correct values.

Do the same for DDNAME DFSUNU.

6. For CICS users the DDNAMES from BMS1LIST to the end are used to create CICS maps. The data sets BMS1LIST and BMS2LIST are used for listing output of the assembler and linkage editor and are partitioned data sets, since multiple mapsets may be produced in a single invocation of the Rules Generator. If you wish these data sets to be permanent, they must be partitioned.
7. For CICS users the DDNAME CICSPT is a sequential data set in which the PPT entries created by the Rules Generator are placed. Since this is a temporary data set, allocated and used during the TSO session, your developers must copy the entries to a permanent data set whenever it is important to preserve them for later input to CICS.

As developers regenerate a transaction, the same information will be overlaid in CICSPT, so they should copy the PPT entries permanently after the last Rules Generator invocation for the transaction.

```

IADFSYS ----- IADF SYSTEM DEFINITION MENU    RULEGEN table updated
OPTION ==>

ADMINISTRATIVE LEVEL:3
KEY ==>                                SYSID ==> SAMP  LEVEL: 3

  1  LIBRARIES      - Create and initialize the libraries which
                     comprise an IMSADF II  system

  2  DEVELOPERS     - Specify TSO userids of developers who have
                     access to an IMSADF II  system

  3  STATIC DSNS    - Specify data sets for generation of data layouts,panels,
                     transactions, drivers    TYPE==> F (F=Foreground B=Batch)

  4  DYNAMIC DSNS   - Specify data sets for generation of audits, messages,
                     help panels, routes, secondary destinations, userids,
                     and profiles            TYPE==> F (F=Foreground B=Batch)

  5  BTS DSNS       - Specify data sets for BTS execution

  6  INITIALIZE     - Initialize IMSADF II  libraries and data bases

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 7-7. Successful Table Update

You are shown again Figure 7-7 with the completion message.

If you wish to use a batch table for background execution of the Rules Generator, please refer to “Establish Optional Batch Tables” on page 7-17. **If DB2 is your access method for the Dynamic rules data bases, your administrative tasks are now complete. Your developers may now begin to use IADF.** Do not complete activities described in the remainder of this chapter nor in Chapter 8, “Initialize SYSID Data Bases.”

Establish Foreground Dynamic Data sets for the SYSID

This section discusses the data sets required for dynamic rules processing for this SYSID in foreground.

Note: Omit this step if DB2 is the access method for the Dynamic rules data bases. DB2 attachment facilities for either CICS or IMS/VS are required and the Dynamic rules data bases tables are “owned” by the DB2 address space. For this reason Dynamic rules data bases activities are not provided by IADF.

Under normal conditions you do not need to modify the list of data sets you are presented, except for the first time.

The term “dynamic data set” requires clarification to avoid confusion with MVS Dynamic Allocation services. Dynamic used here refers to data sets used for what is termed, in IMSADF II, dynamic rules: those that affect the audit, message, and signon profile data bases. The data sets listed in these panels are those required by the High Level Audit Language (HLAL) and the IMSADF II batch driver that updates the data bases. They are dynamically allocated (through MVS) later in the development process as their services are invoked.

```

IADFSYS ----- IADF SYSTEM DEFINITION MENU -----
OPTION ==> 4

ADMINISTRATIVE LEVEL:3
KEY ==>          SYSID ==> SAMP  LEVEL: 3

  1 LIBRARIES      - Create and initialize the libraries which
                    comprise an IMSADF II  system

  2 DEVELOPERS     - Specify TSO userids of developers who have
                    access to an IMSADF II  system

  3 STATIC DSNS    - Specify data sets for generation of data layouts,panels,
                    transactions, drivers  TYPE==> F (F=Foreground B=Batch)

  4 DYNAMIC DSNS   - Specify data sets for generation of audits, messages,
                    help panels, routes, secondary destinations, userids,
                    and profiles          TYPE==> F (F=Foreground B=Batch)

  5 BTS DSNS       - Specify data sets for BTS execution

  6 INITIALIZE     - Initialize IMSADF II  libraries and data bases

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 7-8. Selecting DYNAMIC DSNS Option from IADF System Definition Menu

Enter 4 on the option line to select DYNAMIC DSNS from the IADF System Definition Menu. Make sure "f" is the value beside "TYPE".

```

RULPNLD ----- DYNAMIC SYSTEM DATA SETS -----
COMMAND ==> SCROLL ==> PAGE

Line Commands: D Delete I Insert U Update
C DDNAME      DSNAME      MEMBER      Description
-----
' TASKLIB      '&ADFNODE..ADFLOAD'      MFC1DFR2    STEPLIB FOR HLAL
' LSTFILO      LSTFILO                      LIST FILE
' OUTDATA      OUTDATA                      BATCH TRANSACTIONS
' LSTFILE      LSTFILE                      LST FILE
' WK0          WK0                          WK0
' WK1          WK1                          WK0
' WK2          WK2                          WK2
' KY0          WK4                          WK4
' KY1          WK5                          WK5
' MRGHOLD      WK6                          WK6
' MRGHLD1      WK7                          WK7
' MRGHLD2      WK8                          WK8
' TBL          WK9                          WK9
' STADOUT      STADOUT                     STATIC AUDIT LIST
' MSGOUT      MSGOUT                     STATIC AUDIT MESSAGE
' KEYAUDT      WK10
' AUDMOD      '&ADFNODE..ADFLOAD'      MFC1DFR2    HLAL MODULE
' DFSRESLB     '&IMSNODE..RESLIB'          IMS RESLIB
' TASKLIB      '&IMSNODE..RESLIB'          STEPLIB
'              '&ADFNODE..ADFLOAD'      ADFLOAD
'              '&NEWADF..RULLIB'        ADF RULES
' IMS          '&IMSTEST..DBDLIB'      DBD LIBRARY
'              '&IMSTEST..PSBLIB'      PSB LIBRARY
' IEFORDER     'NULLFILE'                  LOG
' RSTRIN       'NULLFILE'                  RESTART DATASET
' TRANSOUT     TRANSOUT
' PRINTER      PRINTER
' CARDOUT      CARDOUT
' MSGOUT      MSGOUT
' SYSUDUMP     'NULLFILE'
' DLIMOD       '&ADFNODE..ADFLOAD'      MFC1DFR1    IMS REGION CTL
' &RULESDD     RULES DD
' &COMPMDD     COMPOSITE DD
' &ADFID.AUDT  '&DSN21'                  AUDIT IF ADFID = MFC1
' &ADFID.MSGS  '&DSN22'                  MESSAGE DB
' &ADFID.SIGN  '&DSN23'                  SIGNON DB
' DFSVSAMP     VSAM BUFFER POOL
' ERRMSG       'NULLFILE'
' ERRTRX       'NULLFILE'
' SECTR        'NULLFILE'
' MFDPAR01     '&DSN21'                  XACT SEQ DATA SET
' MFDPM01      '&DSN22'                  AUDIT IF ADFID = MFC1
' MFDPSP01     '&DSN23'                  MESSAGE DB
'              SIGNON DB
***** BOTTOM OF DATA *****

```

Figure 7-9. Foreground Dynamic Data Set Panel

Referring to Figure 7-9, observe that two sets of IMSADF II data bases are listed. The names of the data bases if the ADFID is MFC1 is MFDPAR01, MFDPSP01, and MFDPMS01. If the ADFID is anything else, they are AUDT, SIGN, and MSGS all prefixed by the ADFID.

Because you are working on MFC1, use the ERASE EOF key under the DSNNAME column to delete the data set names for DDNAMES &ADFID.AUDT, &ADFID.SIGN, and &ADFID.MSGS. This avoids allocation of those data sets. If a data set name is not present in the DSNNAME column of the table (for a permanent data set), the entire DDNAME is ignored by IADF.

RULESDD and COMPMDD, installation parameter options, are shown as &RULESDD/&COMPMDD in the table. The DDNAMES are symbolic parameters which at execution time have the values (if any) created by your installation substituted.

If your location does not use them, ignore &RULESDD/&COMPMDD, and the TASKLIB (STEPLIB) will be used to load rules. For most of your development cycle, you may prefer this mode. If you wish to load rules at execution time using a RULESDD value, enter either a permanent rules load data set name or &DSN2 (which is known to the SYSID). If you wish to access composite rules using the COMPMDD value, enter the name of your library in the DSNNAME column.

Review the defaults to ensure that they are appropriate.

Use the "Update" line command to review the information for the DDNAME DLIMOD. It contains the parameter list required by the IMS/VS batch region controller, DFSRRC00, at the IMS/VS 1.2 release level. The parameter list may need to be modified, depending on the release of IMS/VS and/or ancillary products is used at your installation.

Press the END PF key to store the table. Figure 7-7 is displayed with the message DYNAMIC TABLE UPDATED on the short message line.

If you wish to use a batch table for background execution please refer to "Establish Optional Batch Tables" on page 7-17. If you use BTS, you establish its table in the next topic.

Establish BTS Data sets for the SYSID

This section discusses the data sets required in BTS execution for this SYSID (IADF supports foreground BTS only). Omit this task if you do not use BTS at your installation.

Notes:

1. CICS/OS/VS users should omit this step. The BTS product does not support CICS.
2. DB2 users also may not invoke BTS in foreground.

Under normal conditions you do not need to modify the list of data sets you are presented, except for the first time.

To modify the panels you see during this activity, you should be very familiar with the cataloged procedures for BTS at your location.

```

IADFSYS ----- IADF SYSTEM DEFINITION MENU    DYNAMIC table updated
OPTION ==>

ADMINISTRATIVE LEVEL:3
KEY ==>                                SYSID ==> SAMP  LEVEL: 3

  1  LIBRARIES      - Create and initialize the libraries which
                     comprise an IMSADF II  system

  2  DEVELOPERS     - Specify TSO userids of developers who have
                     access to an IMSADF II  system

  3  STATIC DSNS    - Specify data sets for generation of data layouts,panels,
                     transactions, drivers    TYPE==> F (F=Foreground B=Batch)

  4  DYNAMIC DSNS   - Specify data sets for generation of audits, messages,
                     help panels, routes, secondary destinations, userids,
                     and profiles           TYPE==> F (F=Foreground B=Batch)

  5  BTS DSNS       - Specify data sets for BTS execution

  6  INITIALIZE     - Initialize IMSADF II  libraries and data bases

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 7-10. Selecting BTS DSNS Option from IADF System Definition Menu

Enter 5 on the option line to select BTS from the IADF System Definition Menu.

BTSPNLS ----- BTS SYSTEM DATA SETS -----			SCROLL ==>	PAGE
COMMAND ==>				
Line Commands: D Delete I Insert U Update				
C	DDNAME	DSNAME	MEMBER	Description

'	TASKLIB	'BTS.LOAD'		BTS LOAD LIBRARY
'		'&IMSNODE..RESLIB'		IMS RESLIB
'		'&ADFNODE..ADFLOAD'		IMSADF MODULES
'		'&DSN2'		USER RULES LOAD
'	DFSRESLB	'&IMSNODE..RESLIB'		IMS RESLIB
'	FORMAT	'&DSN32'		FORMAT LIBRARY
'		'&IMSNODE..FORMAT'		FORMAT LIBRARY
'	IEFRDER	'NULLFILE'		IMS LOG
'	IMS	'&IMSTEST..PSBLIB'		IMSADF PSBLIB
'		'&IMSTEST..DBDLIB'		IMSADF DBDLIB
'	IMSACB	'&IMSTEST..ACBLIB'		IMSADF ACBLIB
'	DFSESL	'&DB2NODE..SSPGM'		FOR DB2 EXECUTION
'		'&IMSNODE..RESLIB'		IMS RESLIB
'	BTSOUT	BTSOUT.LIST		BTS OUT
'	BTSPUNCH	PUNCH.DATA		BTS PUNCH
'	BTSSNAP	BTSSNAP.LIST		BTS SNAP
'	BTSDEBUG	BTSTRACE.LIST		BTS TRACE
'	QALTPCB	QALTPCB.DATA		BTS QALTPCB
'	QALTRAN	QALTRAN.DATA		BTS QALTRAN
'	QIOPCB	QIOPCB.DATA		BTS QIOPCB
'	SYSUDUMP	BTSDUMP.LIST		BTS DUMP
'	BTSLOAD	'BTS.LOAD'	BTSTSOST	BTS MODULE
'	&RULESDD			RULES DD
'	&COMPMDD			COMPOSITE DD
'	&ADFID.AUDT	'&DSN21'		AUDIT IF ADFID=MFC1
'	&ADFID.SIGN	'&DSN23'		SIGNON DB
'	&ADFID.MSGS	'&DSN22'		MESSAGE DB
'	&ADFID.WORK	'&DSN24'		WORK DB
'	ADFIDUMP	ADFIDUMP		ADF DUMP
'	DFSVSAMP			VSAM BUFFER POOL
'	MFDPAR01	'&DSN21'		AUDIT IF ADFID=MFC1
'	MFDPMS01	'&DSN22'		MESSAGE DB
'	MFDPSP01	'&DSN23'		SIGNON DB
***** BOTTOM OF DATA *****				

Figure 7-11. BTS Data Sets

The panel for BTS data sets is displayed in Figure 7-11.

The FORMAT concatenation places the library that you used as output of the MFS Utility in front of another format library which contains all the rest of the IMSADF II formats. You may wish to place more libraries in that concatenation.

Referring to Figure 7-11, observe that two sets of IMSADF II data bases are listed. The names of the data bases if the ADFID is MFC1 is MFDPAR01, MFDPSP01, and MFDPMS01. If the ADFID is anything else, they are AUDT, SIGN, and MSGS all prefixed by the ADFID.

Because you are working on MFC1, use the ERASE EOF key under the DSNNAME column to delete the data set names for DDNAMES &ADFID.AUDT, &ADFID.SIGN, and &ADFID.MSGS. This avoids allocation of those data sets. If a data set name is not present in the DSNNAME column of the table, the entire DDNAME is ignored by IADF (for a permanent data set).

RULESDD and COMPMDD, installation parameter options, are shown as &RULESDD/&COMPMDD in the table. The DDNAMES are symbolic parameters which at execution time have the values (if any) created by your installation substituted.

If your location does not use them, ignore &RULESDD/&COMPMDD, and the TASKLIB (STEPLIB) will be used to load rules. For most of your development cycle, you may prefer this mode. If you wish to load rules at execution time using a RULESDD value, enter either a permanent rules load data set name or &DSN2 (which is known to the SYSID). If you wish to access composite rules using the COMPMDD value, enter the name of your library in the DSNNAME column.

Review the defaults to ensure that they are appropriate.

Change the 'BTS.LOAD' data set name if this is not where BTS exists.

Press the END PF key to store the table. Figure 7-7 is displayed with the message BTS TABLE UPDATED.

Establish Optional Batch Tables

Batch tables are not required for either foreground or batch invocation of IADF services the developer selects.

If the developer selects foreground, the tables established in "Establish Foreground Dynamic Data sets for the SYSID" on page 7-11 and "Establish Foreground Static data sets for the SYSID" on page 7-7 are available for use. If the developer chooses batch submission (and the batch tables do not exist), IADF builds JCL from the appropriate skeletons: PROCGS, PROCALS, or PROCBS. The skeletons may not be satisfactory as distributed for your environment and you may not wish to modify them. For this purpose the batch table option is supplied.

Note: While establishment of batch tables is optional, such is **not true** for the foreground tables. The developer is the one who chooses to invoke IMSADF II services in foreground or batch. If the developer selects batch invocation and tables have not been set up, the skeleton still results in the building of a job stream. If the developer selects "foreground" and you have not set up those tables, there will be nothing to allocate.

You may establish batch tables in several ways:

1. By scrolling through the table presented to you, modifying data set names, as you did when you set up the foreground table, then saving it.
2. By scrolling through the foreground table presented and saving it to establish an "initial" table. Then invoke the same function again and use the "copy" command to copy appropriate rows from the foreground table to the batch table. (This command is described in "Copy Function" on page 7-22).

If your TSO region size is limited, you may prefer to have a batch table for the dynamic data sets (since they invoke the IMS/VS region controller, they require more storage) but use foreground for the Rules Generator and screen utilities.

Deleting unnecessary data sets and DDNAMES is recommended for foreground tables (described in “TSO Performance” on page 2-12), but is not necessary for batch tables, since IADF services then do not affect foreground allocation.

Static Data Sets for Batch

On the SYSTEM DEFINITION MENU shown in Figure 7-5 you select:
TYPE ==> B beside option 3 (Static DSNS). The Rules Generator batch tables are displayed in Figure 7-12.

RULPNLS ----- RULEGEN SYSTEM DATA SETS BATCH -----				
COMMAND ==>			SCROLL ==> PAGE	
Line Commands: D Delete I Insert U Update				
C	DDNAME	DSNAME	MEMBER	Description
'	RULESMOD	'&ADFNODE..ADFLOAD'	&ADFID.G01	RULES GENERATOR
'	IMAGELIB	'&DSN4'		IMAGE LIBRARY
'	GENLIST			GENLIST DATA SET
'	GENDECK			
'	SCREENS	&&&MFS		MFS SOURCE
'	RULEASM			RULEASM
'	RULELEL			RULELEL
'	ASMLIST			ASMLIST
'	MACLIB	'SYS1.MACLIB'		MACLIB LIBRARIES
'		'&ADFNODE..ADFMAC'		ADF MACLIB
'	SYSUT1			SYSUT1
'	SYSUT2			SYSUT2
'	SYSUT3			SYSUT3
'	SYSLIN			SYSLIN
'	PGMLOAD	'&DSN5'		PROGRAM LOAD LIBRARY
'	SYSLIB	'&DSN2'		USER RULES LOAD
'		'&DSN30'		USER EXITS
'		'&NEWADF..RULLIB'		ADF RULES
'		'&ADFNODE..ADFLOAD'		EXITS & ADFID CSECT
'	SYSLMOD	'&DSN2'		SYSLMOD/USER RULES
'	SYSPRINT			
'	SYSUDUMP			
'	TASKLIB	'&ADFNODE..ADFLOAD'		STEPLIB FOR SYSTEM
'		'&DSN2'		USER RULES LOAD
'		'&NEWADF..RULLIB'		ADF RULES
'	ADFSQLHO			SOURCE FOR TABLE HANDLER
'	ADFSQLHW			TABLE HANDLER WORK FILE
'	DBRMLIB		SSSSXX	STORE DB2 DBRM

Figure 7-12 (Part 1 of 2). Batch Static Rules Data Sets

RULPNLS ----- RULEGEN SYSTEM DATA SETS BATCH -----
 COMMAND ==> SCROLL ==> PAGE

Line Commands: D Delete I Insert U Update

C	DDNAME	DSNAME	MEMBER	Description
'	DB2PRINT			SYSPRINT FOR DB2
'	DB2TERM			SYSTEM FILE
'	CICSMAPS	&&&&BMS		BMS SOURCE
'	CICSPPT	&&&&PPT		CICS PPT
'	ISPF	&&&&ISPF		TABLE BUILD
'	DDADFX	&&&&ADFX		ADFIN
'	ADFLIB	'&DSN1'		RULES SOURCE
'	DFSUPA	'&IMSNODE..RESLIB'	DFSUPAA0	MFS STEP 1
'	REFIN	'&IMSNODE..REFERAL'		REFERAL LIBRARY
'	REFOUT	&&&&TEMPPDS		REFOUT
'	REFRD	&&&&TEMPPDS		
'		'&IMSNODE..REFERAL'		
'	SYSTEXT	&&&&TXTPASS		
'	SYSPRINT			
'	SYSUDUMP			
'	SEQBLKS	&&&&BLKS		
'	SYSIN	&&&&MFS		FROM RULEGEN
'	DFSUNU	'&IMSNODE..RESLIB'	DFSUNUB0	MFS STEP 2
'	UTPRINT			
'	SEQBLKS	&&&&BLKS		
'	FORMAT	'&DSN32'		FORMAT LIBRARY
'	ASMMAP		IFOX00	ASSEMBLER FOR BMS
'	SYSPRINT			
'	SYSLIB	'&CICNODE..MACLIB'		CICS MACRO LIBRARY
'		'SYS1.MACLIB'		
'	SYSUT1			
'	SYSUT2			
'	SYSUT3			
'	SYSPUNCH	&&&&MAP		
'	SYSIN	&&&&BMS	&&MAPNAME	
'	LINKMAP		IEWL	LINK-EDIT FOR BMS
'	SYSPRINT			
'	SYSMOD	'&CICNODE..LOADLIB'	&&MAPNAME	BMS MAP LIBRARY
'	SYSUT1			
'	SYSLIN	&&&&MAP		
'	SYSPRINT			

***** BOTTOM OF DATA *****

Figure 7-12 (Part 2 of 2). Batch Static Rules Data Sets

The panels displayed are similar in format to those for foreground. You scroll through the table, making changes if desired, then save them by pressing END.

Several differences between the batch and foreground tables should be noted:

1. The statement actually invoking the program (containing an entry in the **MEMBER** column) appears as the first entry in the section of the table under which its **DDNAMES** are listed. The reason is that in job creation the `// EXEC` statement must appear first in each step.
2. Temporary data sets appear in the **DSNAME** column prefixed by four ampersands (&&&&). This format is required for correct batch job generation, while the foreground form contains a **DSNAME** to which the TSO userid is prefixed at allocation.
3. The Extract data set name (**DDNAME ISPF**) must be permanent in order to run Extract in batch. This function is described in "Extract ISPF Tables from Rules Source" on page 10-6.
4. If you enter "U" under the Line Command ("C") column, to display pertinent information about the allocation of a given data set, an extra line appears for batch tables:

```
SYSOUT      ==> *          (Use for Batch SYSOUT data sets)
```

This line may be completed with another **SYSOUT** class, if required, for those data sets directed to **SYSOUT**.

5. If you use a permanent data set for MFS screens (entered on panel **ADFRGB**) and you wish to retain it for batch table use, change the &&&&MFS to &DSN31 beside the **SCREENS DDNAME**. In addition, you must update its parms, removing the 'A' beside "Allocate" to prevent deletion of the data set.

Even though both the **MFS** and **BMS** generation utilities are provided in the table, only one is selected for generation. The selection is based on either installation options or on the Rules Generator **SYSTEM** statement. Do not eliminate one set.

If you are a **CICS** user and you use the H-level assembler, you may wish to change the **MEMBER** of the **DDNAME ASMMAP** from **IFOX00** to **IEV90** for performance reasons.

Dynamic Data Sets for Batch

On the **SYSTEM DEFINITION MENU** shown in Figure 7-8 you select **TYPE ==> B** beside option 4 (Dynamic DSNS). The dynamic batch tables are displayed in Figure 7-13.

The panels displayed are similar in format to those for foreground. You scroll through the table, making changes if desired, then save them by pressing **END**.

```

RULPNLD ----- DYNAMIC SYSTEM DATA SETS BATCH -----
COMMAND ==>                                SCROLL ==> PAGE

Line Commands: D Delete I Insert U Update
C DDNAME      DSNAME      MEMBER      Description
-----
' AUDMOD      ' &ADFNODE..ADFLOAD'      &ADFID.A02  HLAL MODULE
' TASKLIB      ' &ADFNODE..ADFLOAD'      &ADFID.A02  STEPLIB FOR HLAL
' LSTFILO      &&&&WKX                      &ADFID.A02  LIST FILE
' OUTDATA      &&&&WK0                      &ADFID.A02  BATCH TRANSACTIONS
' LSTFILE      &&&&WK1                      &ADFID.A02  LST FILE
' WK0          &&&&WK2                      &ADFID.A02  WK0
' WK1          &&&&WK3                      &ADFID.A02  WK0
' WK2          &&&&WK4                      &ADFID.A02  WK2
' KY0          &&&&WK5                      &ADFID.A02  WK4
' KY1          &&&&WK6                      &ADFID.A02  WK5
' MRGHOLD      &&&&WK7                      &ADFID.A02  WK6
' MRGHLD1      &&&&WK8                      &ADFID.A02  WK7
' MRGHLD2      &&&&WK9                      &ADFID.A02  WK8
' TBL          &&&&WKA                      &ADFID.A02  WK9
' STADOUT      &&&&WKA                      &ADFID.A02  STATIC AUDIT LIST
' MSGOUT      &&&&WKA                      &ADFID.A02  STATIC AUDIT MESSAGE
' KEYAUDT      &&&&WK10                     &ADFID.A02  WK10
' DLIMOD      ' &ADFNODE..ADFLOAD'      DFSRRC00    IMS REGION CTL
' DFSRESLB     ' &IMSNODE..RESLIB'      DFSRRC00    IMS RESLIB
' TASKLIB      ' &IMSNODE..RESLIB'      DFSRRC00    STEPLIB
'              ' &ADFNODE..ADFLOAD'      DFSRRC00    ADFLOAD
'              ' &NEWADF..RULLIB'      DFSRRC00    ADF RULES
' IMS          ' &IMSTEST..DBDLIB'      DFSRRC00    DBD LIBRARY
'              ' &IMSTEST..PSBLIB'      DFSRRC00    PSB LIBRARY
' IEFORDER     ' NULLFILE'              DFSRRC00    LOG
' RSTRIN       ' NULLFILE'              DFSRRC00    RESTART DATASET
' TRANSOUT
' PRINTER
' CARDOUT      &&&&CARDS
' MSGOUT
' SYSUDUMP
' &RULESDD
' &COMPMDD
' &ADFID.AUDT  ' &DSN21'
' &ADFID.MSGS  ' &DSN22'
' &ADFID.SIGN  ' &DSN23'
' DFSVSAMP
' ERRMSG       ' NULLFILE'
' ERRTRX       ' NULLFILE'
' SECTR        ' NULLFILE'
' MFDPAR01     ' &DSN21'
' MFDPM01      ' &DSN22'
' MFDPS01      ' &DSN23'
***** BOTTOM OF DATA *****

```

Figure 7-13. Batch Dynamic Data Sets

Note: Possible changes described in “Establish Foreground Dynamic Data sets for the SYSID” on page 7-11 also apply to batch.

Several differences between the batch and foreground tables should be noted:

1. The statement actually invoking the program (containing an entry in the MEMBER column) appears as the first entry in the section of the table under which its DDNAMES are listed. The reason is that in job creation the // EXEC statement must appear first in each step.
2. Temporary data sets appear in the DSNNAME column prefixed by four ampersands (&&&&). This format is required for correct batch job generation, while the foreground form contains a DSNNAME to which the TSO userid is prefixed at allocation.
3. If you enter “U” under the Line Command (“C”) column, to display pertinent information about the allocation of a given data set, an extra line appears for batch tables:

SYSOUT ====> * (Use for Batch SYSOUT data sets)

This line may be completed with another SYSOUT class, if required, for those data sets directed to SYSOUT.

Copy Function

If you prefer not to enter batch table data set information, particularly if it is quite similar to that established for the foreground tables, you may use the “copy” command to copy the table information from the foreground table to an **existing** batch table. This “copy” command is not a data set copy, but copies **rows** from one ISPF table to another. Since the tables are not arranged in one-to-one correspondence, its “unit of copy” is DDNAME. It copies a DDNAME and all its concatenated data set information from the foreground table to the batch table. The **purpose** of COPY is to allow you to enter data set names once, for the foreground table and with one command, to update the batch table.

You may use this function:

- After a batch table has been established.
- If you have many user modifications to make to a batch table.
- If modifications to the table are similar between the foreground and batch invocations of the IADF service.

If your table requires little modification from the distributed table originally presented, you do not gain productivity by using “copy”.

The “Allow Copy” parameter assigned to a DDNAME determines what data is copied from the foreground table to batch. This parameter is displayed when you enter “U” under the “Line Command” column beside any DDNAME (see Figure 7-3). The distributed value is set to determine applicability of the DDNAME for the copy command.

The purpose of the batch table is to generate a job stream which is then submitted by the developer. Naturally, the job stream should not contain errors. Certain foreground data sets are incompatible with batch job stream creation and those have “Allow Copy” set to “N” (for no). Others may participate in the copy and have the correct format for either foreground or batch invocation. In those the indicator is blank, which allows the copy to take place.

The following is a list of data sets or categories which do not allow copy as distributed:

1. Batch temporary data sets (those beginning with &&&&).
2. Foreground temporary data sets whose DSNNAME is shown without single quotations. At TSO allocation, the developer's TSO userid is prefixed.
Note: When they are copied, no changes are made to the DSNNAME, but at execution time, permanent data sets are created in batch with the developer's TSO userid prefixed.
3. Data sets involved in BMS map generation have different DDNAMES in foreground and batch tables. Since the DDNAMES do not match, the copy function would fail if it were allowed.

These data sets allow copying:

1. Permanent data sets--IMSADF II libraries, &DSNn data sets.

Normally you should not change the parameters for a data set. A situation in which you might wish to change the "Allow Copy" setting can occur when you change a data set distributed as temporary (copy not allowed) to permanent. In that situation it is permissible to copy the data set, since it would generate correct batch JCL. An example is the SCREENS data set in the Rules Generator invocation, in which MFS source is placed. If you wish to keep that source, you can make it a permanent data set.

If you choose to use the copy function, be aware of the following:

1. You should test the job stream creation process before turning it over to the developers to insure that no errors exist.
2. The "Allow Copy" parameter setting for the first data set in a concatenation determines the ability to copy the entire concatenation.
3. As long as "Allow Copy" is permitted for a DDNAME, its concatenation is copied, without regard for proper JCL coding convention in batch. You can create job streams which cause JCL errors for your developers when you change the setting of "Allow Copy".
4. If you delete a DDNAME in foreground, it must be deleted in batch, i.e. DDNAMES must match or the copy will not complete.
5. If you find data sets which did not copy, examine the "Allow Copy" parameter.
6. A DDNAME may exist twice in a table (TASKLIB, for example). A match is made based on the **description field** which must be the same in both tables, otherwise the copy terminates with a "tables are different" message.
7. If the results of the copy are not correct, use the CANCEL command to discard the table.

Though this discussion has been concerned with copying foreground tables to batch, it applies equally when the reverse is true. **You may copy batch tables to foreground,** but you can cause allocation errors in TSO if they are not correct.

Summary

In this chapter we have covered the tables you must establish for each SYSID and some optional tables you can create. It seems appropriate now to summarize exactly what activities are required, which ones you can omit, and how you go about it.

- No matter which tables you establish for a SYSID, all of them have the same format and may or may not require that you modify data set names and DDNAMES for your environment. All modifications are performed the same.
- You must establish the foreground tables appropriate to your environment (see Figure 7-1 for this information).
- You do not have to use batch tables if the skeletons supplied in IMSADF.ADFSLIB generate job streams appropriate for your installation. If you wish to invoke IADF services in foreground, do not create these tables.
- If you wish to use batch tables, you can establish them in the following ways:
 - Scroll through the batch panel, making changes, and save the results.
 - Establish the table by pressing END.
 - Use the copy command to copy from the corresponding foreground table to an existing batch table.

For the first SYSID you add, you must perform the table functions described in this chapter. For all subsequent SYSIDS, you may employ the technique described in the following section to establish the remainder. Suggestions on ways to minimize administration tasks for multiple SYSIDS are found in “A Technique for Administering Multiple SYSIDS.”

A Technique for Administering Multiple SYSIDS

The procedures described in this section can help to reduce the administrative tasks you must perform when you set up multiple SYSIDS. They are predicated on two things:

- You set up the tables for the first SYSID as described in this chapter.
- The rest of the SYSIDS have more in common with the tables for the first SYSID than they do with the distributed table model.

This technique is not productive if each SYSID has dissimilar characteristics or if the SYSID information requires few changes to the distributed table models.

The first SYSID you set up contains the “master” set of SYSID tables. Accordingly, do not plan to perform development activities using this SYSID. Restrict developer access to this SYSID to a few Administrators.

Note: You may also place all your “master rules” in this SYSID for use by your developers.

Add the remainder of your SYSIDS as described in Chapter 5, “Installing an Application SYSID.”

Use IADF Table Management services (described in “Table Management” on page 10-31) to copy the appropriate table from the “master” to another SYSID.

Example

Create MAST and set up your tables as described earlier in this chapter. Add the rest of your SYSIDS. From the IADF Administration Main Menu, select option "4.3", which is "Table Management" "Copy Function" and you see Figure 7-14.

```
ADFPRI1 ----- TABLES: COPY FUNCTION -----  
COMMAND ==>  
  
    Available Commands: CAN Cancel  REP Replace  
  
FROM:  
  
TABLE Name ==> mastst  
  
DDNAME      ==> mast  
  
DSNAME      ==> 'tsoid1.mast.ispftbls'  
  
TO:  
  
TABLE Name ==> sampst  
  
DDNAME      ==> samp  
  
DSNAME      ==> 'tsoid1.test.ispftbls'  
  
COMPLETE AND PRESS ENTER TO CONTINUE OR END TO EXIT
```

Figure 7-14. Copying From "master" SYSID Table

If you are already working on MAST, all the "From" information is known except the table name, and since the DDNAME is already allocated, you do not need to enter the data set name. If not, enter the DDNAME of the master SYSID, its data set name, and the table name you wish to copy.

Complete the "TO" information: table name, "ISPF Tables" DDNAME, and its data set name as shown in Figure 7-14, so both data sets may be allocated by TSO.

Note: In addition to the member copy, table management changes the identifier stored within the table to reflect the new name. In this example table management changes the identifier to SAMPST from MASTST.

Copy all foreground/batch tables required for your environment to the new SYSID. Change SYSIDS and repeat until all all SYSIDS have been processed.

You can use this same technique if later on you make some changes to the table which affects all others registered to IADF. If the table already exists, you must enter "rep" on the command line to replace it.

Chapter 8. Initialize SYSID Data Bases

You are at the point in your administrative duties that is highlighted in the following list:

1. Add Administrators to this ADFID
2. Add the Application System (SYSID)
3. Add Developers to the Application System (SYSID)
4. Create Static Rule (RULEGEN) data sets
5. Create Dynamic data sets
6. Create BTS data sets
7. **Initialize IMSADF II data bases for the SYSID**

Note: If your installation uses DB2 as the access method for the Dynamic rules data bases, the data base allocation and initialization was performed at installation time. Since there can only be one set of these tables per DB2 system, do NOT attempt to allocate and/or initialize them in IADF. This function is not supported in IADF for DB2 users.

Allocate Data Bases

This chapter shows you how to create and initialize the IMSADF II-supplied data bases your developers will later update for that SYSID. This activity can be done at any time prior to the invocation of IMSADF II functions by IADF.

This series of panels is available for use **ONLY** by these groups:

- Level 3 Administrators (ADFID users)
- Level 2 Administrators (SYSID users)

```

IADFADMN ----- IADF ADMINISTRATION MAIN MENU -----
OPTION ==> 1

ADMINISTRATIVE LEVEL:3
KEY ==>                      SYSID ==> SAMP  LEVEL: 3

  1 SYSTEMS      - Create or modify IMSADF II  systems (SYSID)

  2 MIGRATION    - Perform migration    TYPE ==> M ( M Move  C Copy )

  3 ADMINISTRATORS - Specify administrative users

  4 TABLES      - Create, modify, and/or copy ISPF tables

  5 GLOSSARY     - Update glossary terms

  6 DEVELOPMENT  - Perform IMSADF II  development tasks

  X EXIT        - Terminate IADF

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 8-1. Select SYSTEMS to Initialize Data Bases

From the IADF Administration Main Menu, enter 1 on the option line to select SYSTEMS.

```

IADFSYS ----- IADF SYSTEM DEFINITION MENU -----
OPTION ==> 6

ADMINISTRATIVE LEVEL:3
KEY ==>                      SYSID ==> SAMP  LEVEL: 3

  1 LIBRARIES    - Create and initialize the libraries which
                  comprise an IMSADF II  system

  2 DEVELOPERS   - Specify TSO userids of developers who have
                  access to an IMSADF II  system

  3 STATIC DSNS  - Specify data sets for generation of data layouts,panels,
                  transactions, drivers  TYPE==> F (F=Foreground B=Batch)

  4 DYNAMIC DSNS - Specify data sets for generation of audits, messages,
                  help panels, routes, secondary destinations, userids,
                  and profiles          TYPE==> F (F=Foreground B=Batch)

  5 BTS DSNS     - Specify data sets for BTS execution

  6 INITIALIZE   - Initialize IMSADF II  libraries and data bases

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 8-2. Selecting INITIALIZE Option from IADF System Definition Menu

Enter 6 on the option line to select "INITIALIZE".

```

ADF002 ----- INITIALIZE LIBRARIES/DATA BASES -----
OPTION ==> 2

SYSID ==> SAMP   PGROUP ==> PG   LEVEL:3               USERID   - TS0ID1
                                                    TIME     - 12:00

1 LIBRARIES      - Create or Delete IMSADF II Libraries
2 DATA BASES    - Create or Delete IMSADF II Data Bases

JOB STATEMENT INFORMATION:  (VERIFY BEFORE PROCEEDING)
==> //TS0ID1  JOB (ACCOUNT),'NAME'
==> /*
==> /*
==> /*
==> /*
==> /*

COMPLETE AND ENTER SELECTION NUMBER OR PRESS END TO EXIT

```

Figure 8-3. Initialize Libraries/Data Bases Panel

Enter 2 on the option line to select "DATA BASES".

Enter valid job statement information in the place provided at the bottom of the panel if you wish to initialize in batch.

```

ADFIN0D ----- CREATE IMSADF II DATA BASES -----
COMMAND ==> a

A Allocate Data bases
D Delete Data bases
I Initialize Data bases
X Exclude library from global command
If the command is entered at the top, applies to ALL data sets (GLOBAL)

Execution type ==> 0           (0=online, B=batch)
Unit           ==>           (3330/3350/3380. Required for batch)

CMD  DATA SET NAME           Data base  No. of  VSAM  Unique  Volume
                                type       Tracks  Y|N   Y|N
'    TS0ID1.TEST.AUDIT        Audit     50      n     n     ADFVOL
'    TS0ID1.TEST.MSG          Message    50      n     n     ADFVOL
'    TS0ID1.TEST.SIGNON       Signon     50      n     n     ADFVOL
'    TS0ID1.TEST.WORK         Work       30      n     n     ADFVOL

PRESS ENTER IF MORE FUNCTIONS ARE REQUIRED OR END TO PROCESS AND EXIT

```

Figure 8-4. Allocate Data Bases

Enter the correct data set information for your data bases and A(llocate) on the command line. Allocation includes the creation and cataloging of data sets through ISPF allocation services (ISPF option 3.2). Space allocations shown on this figure are examples only, not recommendations. Track allocation is primary only; secondary is fixed and is about 3/4 of the primary amount.

```

ADFIN0D ----- CREATE IMSADF II DATA BASES ----- Function Complete
COMMAND ==> A

A Allocate Data bases
D Delete Data bases
I Initialize Data bases
X Exclude library from global command
If the command is entered at the top, applies to ALL data sets (GLOBAL)

Execution type ==> 0 (0=online, B=batch)
Unit ==> (3330/3350/3380. Required for batch)

CMD DATA SET NAME Data base No. of VSAM Unique Volume
type Tracks Y|N Y|N
* TS0ID1.TEST.AUDIT Audit 50 n n ADFVOL
* TS0ID1.TEST.MSG Message 50 n n ADFVOL
* TS0ID1.TEST.SIGNON Signon 50 n n ADFVOL
* TS0ID1.TEST.WORK Work 30 n n ADFVOL

PRESS ENTER IF MORE FUNCTIONS ARE REQUIRED OR END TO PROCESS AND EXIT

```

Figure 8-5. Successful Data Base Allocation

When the data bases are allocated, Figure 8-5 is shown again.

For those data bases allocated, an "*" appears in the command column. If the allocation failed, an error code appears, and more information may be obtained by pressing the HELP PF key. The data bases that were excluded by keying "x" in the CMD still have an "X" in that column. The completion message appears on the short message line of the panel.

Initialize Data Bases

The next step is to load the data bases with data; two methods are described. When either is finished, you have concluded the minimum setup phase for IADF. It is now ready for use in application development.

```
ADFIN00 ----- CREATE IMSADF II DATA BASES ----- Function Complete
COMMAND ==> I
```

```
A Allocate Data bases
D Delete Data bases
I Initialize Data bases
X Exclude library from global command
If the command is entered at the top, applies to ALL data sets (GLOBAL)
```

```
Execution type ==> 0          (0=online, B=batch)
Unit           ==>          (3330/3350/3380. Required for batch)
```

CMD	DATA SET NAME	Data base type	No. of Tracks	VSAM Y N	Unique Y N	Volume
'	TSOID1.TEST.AUDIT	Audit	50	n	n	ADFVOL
'	TSOID1.TEST.MSG	Message	50	n	n	ADFVOL
'	TSOID1.TEST.SIGNON	Signon	50	n	n	ADFVOL
'	TSOID1.TEST.WORK	Work	30	n	n	ADFVOL

```
PRESS ENTER IF MORE FUNCTIONS ARE REQUIRED OR END TO PROCESS AND EXIT
```

Figure 8-6. Data Base Initialization

Foreground Initialization.: Enter "I" on the command line for foreground initialization for both the IMSADF II system data bases and work data base as shown in Figure 8-6. If your IADF user profile is set to:

```
Browse Source=N and
Foreground Selection=N
```

the following panels are presented. Otherwise, you view the input in browse mode. You can set your profile at any time (see "Establish ISPF User Profile for IADF" on page 10-1 for instructions).

```

ADFAGF1 ----- DYNAMIC RULES GENERATION FOREGROUND -----

SYSID: SAMP

Help Panel(s):                (One or Range of Panels)
Message(s):                   (One or Range of Messages)
Alternate Routes:              (One or Range of Routes)
Destination(s):                (One or Range of Destinations)
Profile(s)                     (One or Range of Profiles)
User IDs                       (One or Range of User IDs)

                        DATA BASE INFORMATION

MESSAGE Data Base:    TSOID1.TEST.MSG
SIGNON Data Base:     TSOID1.TEST.SIGN

12:00
Now the data sets for Dynamic Rule generation are being allocated...

```

Figure 8-7. Foreground Data Base Initialization

The panel shown in Figure 8-7 is displayed twice in succession: the first with the informational message presented on the lower line of the panel along with the current time, and

```

ADFAGF1 ----- DYNAMIC RULES GENERATION FOREGROUND -----

SYSID: SAMP

Help Panel(s):                (One or Range of Panels)
Message(s):                   (One or Range of Messages)
Alternate Routes:              (One or Range of Routes)
Destination(s):                (One or Range of Destinations)
Profile(s)                     (One or Range of Profiles)
User IDs                       (One or Range of User IDs)

                        DATA BASE INFORMATION

MESSAGE Data Base:    TSOID1.TEST.MSG
SIGNON Data Base:     TSOID1.TEST.SIGN

12:00
Now the Batch Driver Program is being called...

```

Figure 8-8. Foreground Data Base Initialization (Panel 2)

the second time in Figure 8-8. The keyboard is locked for initialization.

The next panel you see contains the output of the Batch Driver program, which loads the IMSADF II System data bases, in Browse mode. Following that is the output of the Work Data Base initialization shown in Browse mode.

When you press END, you see the original panel for this function, Figure 8-6 or Figure 8-3.

Batch Initialization.: If you want to initialize your data bases in batch mode, do the following:

ADFIN0D ----- CREATE IMSADF II DATA BASES ----- Function Complete
COMMAND ==> I

A Allocate Data bases
D Delete Data bases
I Initialize Data bases
X Exclude library from global command
 If the command is entered at the top, applies to **ALL** data sets (GLOBAL)

Execution type ==> **b** (0=online, B=batch)
 Unit ==> (3330/3350/3380. Required for batch)

CMD	DATA SET NAME	Data base type	No. of Tracks	VSAM Y N	Unique Y N	Volume
*	TSOID1.TEST.AUDIT	Audit				
*	TSOID1.TEST.MSG	Message				
*	TSOID1.TEST.SIGNON	Signon				
*	TSOID1.TEST.WORK	Work				

PRESS ENTER IF MORE FUNCTIONS ARE REQUIRED OR **END** TO PROCESS AND EXIT

Figure 8-9. Data Base Initialization

Enter "I" on the command line and "b" beside "Execution Type" as shown in Figure 8-9. Initialization creates a batch job stream to submit to the internal reader.

```
ADFSUBP ----- JOB SUBMISSION -----
OPTION ==>
Source code has been created, choose OPTION 3 to execute the Batch Driver.
SYSID: SAMP      PGROUP: PG      LEVEL:3

      1      EDIT THE FILE BEFORE SUBMISSION
      2      BROWSE THE FILE BEFORE SUBMISSION
      3      SEND THE JOB TO BE EXECUTED
      C      CANCEL THE SUBMISSION

ENTER OPTION NUMBER OR CANCEL/PF3 (END) TO EXIT
```

Figure 8-10. Job Submission Panel

This panel allows you to Edit, Browse, or Send(submit) the job. Enter "3" if you have no changes.

Normal TSO job submission procedures take over at this point.

Chapter 9. Migration Services

The IADF migration function is designed to help coordinate and control phased development and execution environments and to transfer IMSADF definitions between these environments.

Migration of both static and dynamic rules from one set of libraries to another is the topic of this chapter. You are provided a series of panels that guide you through the migration process to ensure that all elements you select are migrated.

This chapter is divided into two parts. The first describes the capabilities and objectives of the migration function within IADF. The second illustrates migration function using specific examples. If you already understand what IADF provides in the way of migration of application SYSIDS, go to “Migration Examples” on page 9-12. Otherwise, you should begin with “Migration Concepts.”

Migration Concepts

In its simplest form, migration means to transfer elements from one entity to another. An example of migration could mean that a SYSID (or any of its elements) is copied from test libraries to production ones.

Migration in IADF encompasses two general operations: **move** and **copy**. Within the migration function panels, IADF uses **copy** to mean that elements are copied to receiving libraries, but remain in their libraries of origin. **Move** is used on the panels to indicate that elements are copied to a set of libraries and are DELETED from the originating set.

In order for an IMSADF II application to be migrated, the following elements must reside in production libraries. Most of them can be migrated directly by IADF.

STATIC Static elements migration is described in “System Elements” on page 9-22, “Segment Element” on page 9-23, “Transaction Element” on page 9-25, “Transaction Drivers Element” on page 9-27. All static elements are fully migrated.

DYNAMIC Migration of audit source is described in “Audit Group Element” on page 9-29. You may then submit a batch job to invoke the HLAL compiler and update the audit data base.

Migration of message data base elements, including messages, HELP screens, and routing information is described in “Update Message Data Base” on page 9-31. Source is generated, so you must invoke the batch driver to update the message data base.

Profiles are NOT migrated for security reasons, since the profile identifiers and users may differ greatly.

FORMATS Format control blocks are not copied. To place them in IMSVS.FORMAT, you may either copy them from your test library or generate them from source. To do that, you should select the “source” form of migration. Refer to “Source” on page 9-2 for more information on source migration.

DL/I

Migration of application DBD, PSB, and ACB is outside the scope of IADF. However, you can accomplish it using a technique described in “Non-Standard Element” on page 9-36.

You must add the IMS/VS transaction to your Stage 1 System Definition yourself.

Migration Function in IADF

Highlights of migration function are described in the following sections. In addition to the services described in this chapter, parts of these administrative tasks affect migration:

1. Initialize SYSID (“Add the Application System” on page 5-2)
2. Add Developers (Chapter 6, “Adding Developers to the SYSID”)

Two-Phase Migration

IADF manages migration in a two-phase operation. Phase I gathers data about the migration desired as you progress through the selection panels. A control file is built with those requests in translated form and is used to build a TSO batch job. (Its format is found in “Format of the Control File” on page 9-6).

The job stream is submitted as Phase II and actually invokes the utilities required to migrate the data from one SYSID to another. The elements are migrated in TSO batch to allow the user to free terminal resources. The utilities may require considerable time if many elements are selected for migration.

The user is notified of the result by recovering the output from this job when it has completed. The results of migration are stored in a centralized log for an audit trail.

Source/Load Migration

IADF supports both source and/or load forms of migration, which are selectable by the user. The source form must be used for messages, the SYSTEM statement, and non-static audits.

Source: All the elements for a SYSID are stored in IADF ISPF tables or libraries, designated when you established the SYSID as described in Chapter 5, “Installing an Application SYSID.”

IADF searches the input library, either rules or dynamic, for the requested member. If the member is not found, temporary source is dynamically created by migration services from the IADF tables and placed in the **input** source library. IADF then migrates the source.

When Phase II migration is complete, you may then generate the required rules and screens. Since each element is copied as a single member, you must create a separate member to be used as the SYSIN to the Rules Generator. You must put in the appropriate INCLUDE statements for the elements you want to generate. Then you use a standard Rules Generator procedure to create the rules and screens.

Note: A migration log shows element names copied during an individual session.

Load: Load modules are migrated directly from the rules library you named when you installed your SYSID to your target library during Phase II. This form of migration is faster, since there is no need to manipulate source. Remember that formats are not copied, so you should not use this option exclusively.

Centralized Log

This data set contains the complete history of an application system. It is named (and allocated, if necessary) during Phase I migration as a permanent data set. Time-stamped data is added for migrated elements during Phase II in order to provide an audit trail. It is always opened by migration services as DISP=MOD.

User Exit

An optional user exit is allowed, which can be used to provide more customization on migration. It can be written as a clist or a program and may be invoked at three points during migration:

- At initialization time, the exit is provided with the list of categories which are to be migrated and the TSO userid of the migrator.
- As each element is prepared for migration, the exit is provided with pertinent information about the category and element to be migrated.
- At termination, the exit is provided with the number of elements migrated and maximum return code of the process.

The exit signals to migration to continue the operation, cancel the operation, or to redo the operation because changes were made in the exit. The exit is provided access to the migration log data set as well as the control file at initialization time. It communicates to migration using return codes, to cause messages to be placed in the log.

The exit can change the list of categories, change the log and control data sets, or cancel the operation. The exit could be used to change data such as the Rules Generator keyword LRULE from ALT to YES when production migration is performed. An example of this is shown in Appendix D, 'Sample Migration User Exit.' The user exit is described more fully in "Migration User Exit" on page 9-9.

Checkpointing Capability

Since Phase II of the process may take time to complete for SYSIDS with many elements, checkpointing is provided and may be requested in Phase I by specifying either a limit in CPU seconds or in the number of elements migrated.

This function is really a 'stop after' situation. Once the limit is reached, Phase II **terminates** after creating a restart record in the control file describing its current position.

The restart may be completed later by the user by resubmitting the Phase II job stream (the initiation of Phase I migration invokes a currently active restart and rebuilds the Phase II job stream).

Migration Authority

Access to the migration functions may be as restrictive or as open as you desire. You may prefer to limit this ability to Administrators; however, any user may be designated (by an Administrator of any level) to perform migration services. You specify them in the task described in Chapter 6, "Adding Developers to the SYSID."

The table that contains authorization for migration is `ssssPRad` in the "ISPF Tables" data set. Each developer authorized in any way for SYSID activities, including migration, must be named either specifically or generically. The table contains two columns which apply to migration, one for migration **from** this SYSID, the other for migration **to** this SYSID. There are four possible values:

Blank No migration is allowed for this developer (default).

M Move form of migration allowed.

C Copy form of migration allowed.

B Both forms allowed.

"Generic" users are supported in the Developers table, so that each individual who may perform migration need not be named.

#(ADM) Grants access for migration to all Administrators with the ability (not a requirement) to specify a level number, such that only Administrators with that level are allowed access.

#(DEV) Is used to grant access to any developer whose level matches the one specified at the `LEVEL` column. Access must be specified for this generic profile (for example, only developers of level 3 might be permitted migration privileges).

Generic Administrative users are automatically established by IADF when the Developers table is initialized, but may be deleted to restrict access if so desired. A generic profile may also specify the `FROM` or `TO` columns to reduce the type of function allowed.

The columns provide the maximum amount of flexibility in change control. You may want to **MOVE** applications to the production system, but to **COPY** them to a maintenance system (movement would delete them from production, resulting in interrupted service). Some users limit migration to Administrators only, in which case the generic Administrator is all that is required; but for a sensitive application, you may want to have only one or two individuals responsible for migration and not to use the generic identifier. You may wish **any** developer to copy applications **from** production, but only your change control team to perform migration **to** production.

During migration, the source SYSID must pass verification that both the user and the migration form match in the Developers table under the "FROM" column, otherwise the function will not be performed.

Once the "FROM" SYSID validation is completed, verification of the same table in the target system is performed where the "TO" column for this user must pass the same criteria for both user and form of migration. If both are not true, migration is denied. If only one version of a SYSID exists, no verification can be performed on the "TO" SYSID.

Multiple Version SYSIDS

You may establish multiple versions of **one** SYSID using IADF, the purpose of which is to facilitate application migration through multiple levels of test to production environments. No restriction on development activities at any level occurs because all elements selected are migrated, along with their related ISPF tables (in the "ISPF Tables" data set). To promote maximum flexibility of change control in any given installation, IADF imposes no limits on the number of versions of a SYSID which may be established. Multiple path selection is also supported should the Administrator select it.

The use of multiple version SYSIDS is optional, but recommended. In addition to your having more than one version of a SYSID on which you can perform development activities or maintenance, migration services can provide more structured change control and authorization verification.

It is possible to "wrap around" versions and overlay them. In this three-level scheme, if

```
SYSID Version BASE ==> SYSID Version TEST and
SYSID Version TEST ==> SYSID Version PROD and
SYSID Version PROD ==> SYSID Version BASE
```

then PROD could overlay TEST.

Though the version paths allow for maximum flexibility, in practice each installation usually follows a proscribed migration pattern, which is known to the Administrator who must set up migration paths.

At SYSID initialization (described fully in Chapter 5, "Installing an Application SYSID"), each version is initialized as a separate SYSID with a version identifier. Each has separate libraries and tables, and the eventual migration path is selected at this time. It is identified in the SYSTEMS table with a key of

```
ADFID  SYSID  VERSION
```

The migration paths are stored in the ADFVER table with a key of

```
ADFID  SYSID  fromversion  toversion
```

At migration time, if a version has more than one possible path, you are presented with the possible migration paths to determine which one of them is the target.

The following controls are imposed by IADF:

- A SYSID version may be migrated to any other version of the **same** SYSID in the **same** ADFID.
- Migration to versions not in the path is disallowed.
- Migration paths for a SYSID may be changed after initialization by a Level 2 Administrator using special utilities.
- ISPF tables are migrated **only** for multiple-version SYSIDS.
- If a SYSID does not have multiple versions, the user is prompted for the target libraries.

"ISPF Tables" Considerations

For multiple version SYSIDS, affected tables are migrated with their selected source/load elements. The tables are deleted from the source on a MOVE.

In the cases of partial migration (MOVE) for segments, the segment and its entry in the control table ssssALL are only deleted if the segment is not referenced by any active transaction.

Tables that contain control information are updated in the target system to reflect accurate information about the migrated segments or the cross references between segments and transactions.

If a complete SYSID/VERSION is MOVED, it is deleted from the SYSTEMS table.

Note: The paths to other versions are not deleted since you may not wish to destroy existing path relationships. Maintenance panels (invoked by the IADF PATHC command) allow you to perform this task yourself, but it is not required.

Library Management Facility

LMF under ISPF is fully supported using the MOVE function, but is not available on COPY. If you use LMF PROMOTE, you must have established the required libraries and architecture prior to IADF migration. You must also have installed the LMF user SVC mentioned in "LMF Support" on page 2-12. All movement to LMF libraries are to the "entry level" library only.

Although LMF services are invoked on behalf of migration functions, you do not have to use the LMF form of library management. If you do not, IADF invokes IEBCOPY.

Note: Since IEBCOPY is a program that must execute in an authorized state, be sure it can be properly invoked in your environment. For more information refer to "IEBCOPY" on page 2-12.

Format of the Control File

The control file is written by Phase I of migration. Its name (unless changed by a user exit) is MIGRATE.CNTL prefixed by the TSO userid. It contains the following records in the following sequence:

1. Restart (record 1)

It is inserted when an error is found and before termination of the process (to aid the user in the resumption of migration, once the cause of the failure has been corrected), or when the preselected checkpoint limit has been reached.

This record is updated after a normal or abnormal termination of Phase II.

If errors occur of which Phase II is unaware, it cannot update the restart record. An example would be job cancellation, or an 'X37' type abend. The user may update this record manually after determining the cause.

Record format:

Column 1	Record ID "5"
Column 3	Restart type
	E Error found
	C Checkpoint restart
	X No restart is currently active.
Columns 5-10	Relative record number to resume
Columns 12-80	Reason for failure or checkpoint

2. Exit (record two)

This record describes the name and type of the exit to be used. Detection of the presence and validity of an exit is verified in Phase I of migration. This record is present in the file even if no exit is selected.

Record format:

Column 1	Record ID "1"
Columns 3-10	Exit name or blanks if no exit is active
Column 12	Exit type: P = program, C = clist
Column 14	Times to be called during Phase II:
	1 For each element
	2 At termination time
	3 Both

3. ID (record three)

This record contains the TSO userid of the requestor and the data set name of the centralized log in use during this migration request. Its availability was verified in Phase I.

Record format:

Column 1	Record ID "3"
Columns 3-10	TSO userid of requestor
Columns 12-44	Data set name of migration log

4. Checkpoint (record four)

This record triggers the automatic ending of the process when the requested number of elements to migrate is reached. Phase II appends a restart record to the control file describing its present position in the process. If this record is active during a run, restart of Phase II is automatically executed, starting at the interrupted element.

Record format:

Column 1	Record ID "4"
Column 3	Checkpoint type
	T Elapsed CPU time in seconds
	N Number of records processed
	X No checkpoint is active.
Columns 5-10	Time in seconds (for T) or number of records (for N)

5. Extraction record

This record contains the data to be migrated. There are as many occurrences of this record as there are number of elements migrated. Category names are stored in the IADFCAT variable. This record may have two formats.

For non-EXT categories (standard migration elements), the format is:

Column 1	Record ID "2"
Columns 3-5	Category type (as described in the IADFCAT variable)
Columns 7-14	IADF table name to use
Columns 16-23	Member name one (source)
Columns 25-32	Member name two (source)
Columns 34-41	Member name one (load)
Columns 43-50	Member name two (load)
Columns 52-77	Promote reason (optional if LMF is used)

For EXT (external) categories there are two records.

Note: EXT categories are those exits and DL/I control blocks which are migrated using panels described in "Non-Standard Element" on page 9-36.

Record one contains:

Column 1	Record ID "2"
Columns 3-5	Category type (as described in the IADFCAT variable)
Columns 16-23	Member name
Columns 25-79	Source library

Record two contains the following if LMF is not used:

Column 1	Record ID "2"
Columns 3-5	Category type (as described in the IADFCAT variable)
Columns 25-79	Target library

Record two contains the following if LMF is used:

Column 1	Record ID "2"
Columns 3-5	Category type (as described in the IADFCAT variable)
Columns 25-32	Project for library
Columns 34-41	Group for library
Columns 43-50	Type for library

Migration User Exit

An optional user exit may be specified in the migration panel. It is verified for existence before the actual migration begins. It may be written as a clist or a program. In the first case, it must exist in the SYSPROC library concatenation; if it is a program, it must reside in a library concatenated to ISPLLIB DDNAME.

A model user exit called ADFMEXIT is supplied with IADF and is shown in Appendix D, “Sample Migration User Exit.” This clist should be used as a model for any user exit, since it contains the basic operations for any exit.

The user exit may be called at three points:

1. Initialization time (called by Phase I of migration). The exit is provided with the following information:
 - a. List of categories to migrate: e.g. segments, transactions, etc. (separated by commas). The values of the list are explained under the IADFCAT variable. This list is stored in the IADFLST variable.
 - b. TSO userid requesting the migration.
 - c. Data set name of the centralized log, fully qualified (stored in the IADFLOG variable)
 - d. Data set name of the Phase II control file, fully qualified (stored in the IADFCNTL variable)

At this time the exit is able to change the list of categories, change the log and control data sets, or cancel the operation. If changes are made to the list or control file, those changes must conform with the allowed values.

The exit sets a return code in IADFRC to signal the action to take:

0	Continue the operation. No changes were made to the list of categories.
1	Continue the operation. Changes were made to the list of categories that conform to the format of Phase II.
2 and greater	Cancel the complete migration process. A reason should be passed back to insert in the log (stored in the IADFRESN variable).

The exit may also set the variable IADFEXT to indicate when the same routine should be called again.

2. Every time an element is prepared for migration (Phase II), the exit is provided with:
 - a. Identification of category and element to be migrated (stored in the IADFCAT variable).
 - b. Name of components (up to four) of the element, e.g. source, load (stored in the IADFCOI variable).

If a member is not present, “NONE” is used for the name.

- c. Table name of the element. The table is opened and accepts changes to it. You could change LRULE keyword from ALT to YES in a production environment (stored in the IADFCATN variable).
- d. Source found indicator (stored in the RS RS1 variables).
- e. Load found indicator (stored in the RL RL1 variables).
- f. Member names of source (stored in the MBR MBRS1 variables).
- g. Member names of load (stored in the MBRL MBRL1 variables).

The exit sets a return code in IADFRC to signal the action to take:

0	Continue the operation. No changes were made to the table.
1	Continue the operation. Changes were made: update the table and recreate the source.
2	Cancel the operation for this element. A reason might be passed back in the IADFRESN variable to insert in the log.
3 and greater	Cancel the complete migration process. A reason should be passed back in the IADFRESN variable to insert in the log.

3. Termination time (Phase II). The exit is provided with the following values (separated by a blank) in the IADFRESN variable:

- Number of elements migrated
- maximum return code of the process

The exit may set a return code in IADFRC greater than zero to cause a termination message stored in IADFRESN to be inserted in the log

The migration routine determines from the IADFRC variable the phase that is being called. If termination is not normal, the reason is available in the IADFRESN variable.

Communication with the User Exit

Only ISPF variables are used for communications between the user exit and the migration programs. They are defined and stored by migration in the shared pool and the names are as follows:

IADFRC On entry contains the phase that is being called:

- "I" for initialization
- "E" for every migration request
- "T" for termination

On exit it contains the return code of the exit routine.

The following return codes are valid for initialization:

- 0 Continue with the process
- 1 Changes were made to the list; update it
- 2 Cancel the process (reason in IADFRESN)

For each element, the codes and meanings are:

- 0 Continue operation
- 1 Continue operation, but update table and recreate source
- 2 Cancel operation
- Any other. Cancel migration (a reason should be placed in IADFRESN).

For termination the codes and meanings are:

- 0 Continue with termination
- Not zero. Write a message in the log (stored in IADFRESN) and continue with termination

If the code is not set up on exit, migration assumes that the exit did not run properly and it will not be called again. This feature may be used by the exit to be called only once to verify the input to migration and never be called again, instead of using the variable IADFEXT.

IADFLST Contains the list of categories to migrate, separated by commas. The names used for this list are the same as in the IADFCAT variable.

IADFLOG Name of the centralized log. This name is fully qualified with no quotes. The name may be changed by the exit at initialization time. If the name is changed, Phase I verifies the new data set and if an error is found, Phase I terminates with an error message.

IADFCNTL Name of the control file. The name is fully qualified with no quotes. The name may be changed by the exit at initialization time. If the name is changed, Phase I verifies the new data set and if an error is found, migration terminates with an error message. The exit is responsible for the contents of this variable, which must conform with the expected format.

IADFCAT Identification of the category being migrated:

- SEG for segment category
- TRX for transaction category
- DRV for transaction driver category
- SYS for system category
- AUD for audit groups
- EXT for non-standard (external) categories
- POM for signon and primary option menu
- SOM for secondary option menu
- STX for secondary transaction
- MSG for messages
- HLP for help panels
- ROU for routing information
- SDT for secondary destinations

IADFCOI Member names of the components being migrated for this category. Four member names separated by blanks may be passed.

1. First source member
2. Second source member
3. First load module name
4. Second load module name

If a member name is not present, "NONE" is used instead.

- IADFCATN** Table name of the opened table to be migrated. The user exit may find the names of the variables and keys by using the TBQUERY instruction against the table. The table should be left open at exit.
- IADFRESN** Used by the exit to pass the reason for the cancellation of a process. It is also used by migration at termination time to pass to the exit information about the termination status.
- It contains three variables, separated by blanks, depending on the termination type:
1. **NORMAL** / number of elements / maximum return code
This format is used for normal terminations. It provides the exit with two numbers: one for the number of elements migrated, and other for the maximum return code found.
 2. **CHKPT** / record number / maximum return code
This format is used when the maximum CPU time or number of elements specified by the user is reached. The number of elements processed and return code are also passed following the literal.
 3. **ERROR** / record number / description of the error
This format is used when an abnormal condition is reached by the process.
- This variable is used to notify the exit at the initialization call that a restart condition from a checkpoint exists. It contains the word **restart** followed by the date and time of this condition.
- IADFEXT** Only used at the initialization call. It shows to Phase I the times that Phase II is to call the exit:
- | | |
|-------------|--|
| zero | Do not change defaults already set up |
| 1 | Call the exit only for each element |
| 2 | Call the exit only at termination time |
| 3 | Call the exit for each element and at termination time |
| 4 | Do not call the exit again |

Migration Examples

Since there are so many panels for this function, a single example is not carried through. Instead, for each sub-function, you see the panel flow displayed. The purpose for the migration function is to COPY/MOVE static and dynamic rules to another set of libraries, so your main task is to select from lists of elements the ones to be copied. Refer to the HELP panels if you wish more detail.

The LMF migration panels differ slightly from standard panels. In this chapter standard migration panels are shown, and the LMF differences are noted in the explanation of the individual panel.

Before you begin migration, you should make sure your user profile, described in “Establish ISPF User Profile for IADF” on page 10-1, has the “Browse Source” option set to N(o). Otherwise you are shown each migrated element in browse mode if IADF has to generate source from the SYSID tables. This can cause you unnecessary screen iterations and significantly slow the process.

Another profile option which improves migration Phase I performance is to set "Bypass Permanent Source" to N(o). If you choose this option, IADF builds all data to be migrated from their ISPF tables repository. It is faster to build from tables than to search libraries for existing source. This option also avoids conflicts caused by the existence of both tables and source which may not be in synchronization.

Element Selection

The elements which are migrated by IADF are all static rules and all dynamic rules except profiles.

Formats/maps and DL/I control blocks (ACBs etc.) are not migrated. For each of the migration element types, there is a section in this chapter, containing the panel flow for that element.

System	Contains the SYSTEM statement, Signon generation statement, Primary and Secondary Option Menu rules. (Panel flow is described in "System Elements" on page 9-22).
Segments	Consisting of Segment Layout, Segment Handler, Pseudo segments, and mapping rules. (Panel flow is described in "Segment Element" on page 9-23).
Transactions	Include Input Transaction rules, Secondary Transactions and composite rules load modules, if you use them. (Panel flow is described in "Transaction Element" on page 9-25).
Transaction Drivers	Are the Mini Drivers for the transactions. (Panel flow is described in "Transaction Drivers Element" on page 9-27).
Audit Groups	Designate the Static Audit load modules only. (Panel flow is described in "Audit Group Element" on page 9-29).
Non-Standard (Exits)	Include audit exits, special processing routines, or exits of other types you may wish to migrate. You can migrate DL/I control blocks as Non-Standard elements. (Panel flow is described in "Non-Standard Element" on page 9-36).
Messages	Include IMSADF II help panels, alternate routing, and destinations, as well as messages. (Panel flow is described in "Update Message Data Base" on page 9-31).

Migration Menu Selection

Migration is reached through the selection of the Administration Main menu or through the Advanced User menu shown in Figure 9-1.

The IADF Main Menu (Advanced User) eliminates menu selections. You can execute all IADF functions with a maximum of two panel levels. In most cases you execute the function directly. To activate the IADF Main Menu (Advanced User), you enter a Y(es) for "IADF Advanced Menu" in your user profile. It is recommended that you use the Advanced User menu for two reasons:

1. On the Advanced User menu you can specify **both the SYSID and VERSION**. Otherwise you must use the VERSION named in your profile (see "Establish ISPF User Profile for IADF" on page 10-1) or change it in the profile.
2. Developers with migration authority can reach migration only with the Advanced User menu.

```

IADFADVM ----- IADF MAIN MENU (Advanced User) -----
OPTION ==> 50

                                USER: TS0ID1
KEY ==>                          SYSID ==> SAMP VERSION ==> BASE LEVEL:3

----- DEFINITION -----
 1 ADFPROF      19 RREK
 2 AUDSEG       20 SDT
 3 AUDSUB       21 SEGMENTS
 4 AUDTBL       22 SIGNON
 5 AUDTRX       23 SYSTEM
 6 DBEK         24 TBL
 7 DBS          25 TRX
 8 DRIVER       26 USER
 9 ESDS
10 HELPPANL
11 KSDS
12 MAP
13 MESSAGE
14 OUT
15 POMENU
16 PS
17 ROUTES
18 RRDS

- GENERATION -
27 AUDITB
28 AUDITF
29 DYNAMICB
30 DYNAMICF
31 FREEDYNA
32 FREESTAT
33 STATICB
34 STATICF

----- TEST -----
35 BTS
36 FREEBTS
37 PROTOTYPE

----- UTILITIES -----
38 ADFIN      51 MFSGEN
39 ADFOUT     52 OUTLIST
40 ADFTRACE   53 PFKS
41 BROWSEF    54 PROFILE
42 DBDGEN     55 PSBGEN
43 DDREPORT   56 REPORTS
44 EDITOR     57 RGLGEN
45 EXTRACT    58 SPF
46 GFIND      59 SUBMIT
47 GLOSSARY   60 SYSID
48 IADFTRACE  61 TERMINAL
49 LOG        62 TUTORIAL
50 MIGRATION

ENTER NAME OR OPTION NUMBER OR PRESS END KEY TO EXIT
  
```

Figure 9-1. IADF Main Menu (Advanced User)

- Type 50 on the option line to select MIGRATION.
- Enter the SYSID and VERSION from which you wish to migrate if it is different from that shown on the panel.
- Press the ENTER key.

The panel with which you begin migration depends on conditions within both your SYSID and the migration process itself. Progress through this section, beginning with the condition described below, and follow the panel flow through to the “Migration Function Menu” on page 9-19, which initializes the migration process.

- If you are in a “restart” condition from a previous migration, that is, your Phase II migration has not completed normally, you start with “Restart Panel.”

If you perform Phase I migration, but never submit Phase II, you receive an error message the next time you invoke migration services. If you never wish to run Phase II, **you** must delete the restart table (ssssRST) found in the “ISPF Tables” data set.

- If your SYSID does not have multiple versions, begin with “SYSID (No Versions)” on page 9-17.
- If your SYSID has multiple versions and contains multiple paths, you must select the desired path. Begin with “Multiple Path Selection” on page 9-18.
- Otherwise, you begin migration with “Migration Function Menu” on page 9-19.

Restart Panel

If Phase I is executed after a checkpoint or error situation in Phase II, Figure 9-2 is displayed to inform you of this condition.

The information shown in Figure 9-2 is for restarting after a checkpoint had been reached, when migration terminates itself. If an error which Phase II was able to trap had caused the termination, you would see “ERROR IN PREVIOUS RUN” in place of the “CHECKPOINT RESTART” line. In addition a “reason code” would have appeared under the “Error Description” line.

If Phase II migration determines that an abnormal condition occurs, such as a non-zero return code from a utility, it updates the restart record in the control file (see 1 on page 9-6) and saves the reason code before terminating. For an error involving a member (when the utility completes normally), a message is issued on the output listing and log, and Phase II continues. You must then look at the listing to resolve the situation.

If an error occurs of which Phase II is unaware, such as its own cancellation, you must determine the restart point and update the control file yourself (see more information in “Phase I Completion” on page 9-38).

```

ADFMGP3 ----- MIGRATION FUNCTION -----
COMMAND ==>

*****
*           MIGRATION RESTART           *
*****

SYSID:  SAMP VERSION:  BASE

A Restart Condition was detected in the Control File caused by

CHECKPOINT RESTART

Error Description (if any)

Record Number where Restart will Begin
0020

New Record Number  ==> 0020    (Change Restart Record Number)

Instructions:
    Press ENTER  Key to Continue with Restart from Record
    Enter  END   Command to Exit Migration
    Enter DEFER  Command to Defer Restart and Exit, or
    Enter CANCEL Command to Cancel Restart and Begin New Migration

```

Figure 9-2. Checkpoint Information for Restarting

When you see Figure 9-2, you may

- Press ENTER to re-execute Phase II.

When you press ENTER, you go through the process described in “Phase I Completion” on page 9-38. During a restart condition, no data collection (Phase I migration) is allowed as it is assumed that the control file contains the same information as in the previous run.

- Press END PF key to exit migration.
- Type DEFER to postpone Phase II and exit Phase I.
- Type CANCEL to reject this restart condition and proceed with a new migration.

When you do this, migration deletes its restart table (ssssRST in “ISPF Tables” data set) and the control file (MIGRATE.CNTL prefixed by TSO userid).

If you see this panel, you should first complete the previous migration and then start over with your new migration request at Figure 9-1.

SYSID (No Versions)

If the SYSID to be migrated consists of only one version, Figure 9-3 appears, requesting the set of target libraries to be used. Since this SYSID is external to the versioning capabilities of IADF, the target libraries are not known. (More structured change control is possible when multiple version SYSIDS are used). The only authorization which is verified is for the existing SYSID.

```
ADFMGPN2----- SYSID (WITHOUT VERSIONING) TARGET LIBRARIES -----
COMMAND ==>                                     SCROLL ==> PAGE

SYSID: SAMP VERSION: BASE

Type      ==>          (C=Copy Function, M=Move Function)
Use LMF ==>          (Y|N Use LMF Promote Function for Move)

OUTPUT LIBRARIES:
  RULES SOURCE      RULES LOAD      DYNAMIC SOURCE
  PROJECT ==> ADF    PROJECT ==> ADF    PROJECT ==> ADF
  GROUP   ==> PROD   GROUP   ==> PROD   GROUP   ==> PROD
  TYPE    ==> RULSRC TYPE    ==> RULLIB TYPE    ==> DYNsrc

OTHER OUTPUT PARTITIONED DATA SETS:
  RULES SOURCE      ==>
  RULES LOAD        ==>
  AUDIT SOURCE      ==>

ENTER = PROCESS AND CONTINUE  END = EXIT WITHOUT PROCESSING
```

Figure 9-3. SYSID without Versioning

Enter the libraries to which the elements are to be migrated. There are two rows of libraries.

1. If you use LMF PROMOTE, you must enter your library names on the first row. All libraries have a three node naming convention, similar to ISPF libraries. The second level ("GROUP") must be the library entry level name in your LMF-defined architecture.
2. The second row is for non-LMF libraries which may have more or fewer nodes in the fully qualified library name. If your libraries have three nodes, you may use the first row of libraries if you find it more convenient.

The input libraries were established at the time you installed this SYSID and are allocated to the migration services at this time.

Note: The ISPF tables are **not** migrated to the target system for any elements selected for migration.

When you have completed all the information required, press ENTER to begin processing the "Migration Function Menu" on page 9-19.

Multiple Path Selection

You begin migration at this point if the SYSID/VERSION from which you are migrating has multiple paths.

If the versions table contains more than one possible target for this version, Figure 9-4 appears.

```
ADFMGPN4----- MULTIPLE TARGET SYSID -----
COMMAND ==>                                SCROLL ==> PAGE

Available command: C Cancel

System ID:    SAMP
Version ID:   BASE

Line Commands: S Select the Target SYSID to Migrate (One Selection Allowed)

C      VERSION
-      ----
S      TEST
'      TES1
***** BOTTOM OF DATA *****
```

Figure 9-4. Multiple Targets (Paths) for A SYSID/VERSION

The selection list presented includes all the possible targets for a particular version from which only one may be chosen.

Once you select the version, verification occurs. You must be authorized for migration in the "TO" version, and the form of migration appearing must be compatible with that of the "FROM" (or current) SYSID. For example, if you were authorized only for COPY in the current SYSID/VERSION, you must have COPY authority in the selected target version (if either COPY or BOTH forms are permitted, your authorization is verified). If the selected version prevents you from performing the migration operation (see "Migration Authority" on page 9-4), a message is displayed, and the migration is terminated.

If your selected target is valid, you continue with "Migration Function Menu" on page 9-19.

Migration Function Menu

Figure 9-5 is displayed.

This panel is the control panel of migration and its appearance signals completion of the preceding sections. You have reached this panel if:

- You have no currently active migration.
- Your SYSID is valid; its path has been selected or derived.
- You are authorized to migrate in both the "FROM" and "TO" versions.

```

ADFMGPM ----- MIGRATE FUNCTION MENU -----
COMMAND ==>
SYSID: SAMP      VERSION: BASE  TO: TEST
Full Migration ==> n      (Y|N Migrate a Complete SYSID)
Type           ==> c      (C=Copy Function, M=Move Function)

Use LMF        ==> n      (Y|N Use LMF Promote Function for Move)
Source         ==> y      (Y|N Migrate Source)
Load          ==> y      (Y|N Migrate Load Modules)

Specify Elements to be Migrated if Full Migration not Selected:
System         ==> y      (Y|N Migrate SYSTEM/SIGNON/SOM)
Segments       ==> y      (Y|N Migrate Segments)
TRXs           ==> y      (Y|N Migrate Transactions/STX/COMPOSITE)
Drivers        ==> y      (Y|N Migrate Transaction Drivers)
AGROUPs        ==> y      (Y|N Migrate Audit Groups)
Messages       ==> y      (Y|N Migrate Messages,Help Panels,Routes,Sec.Dest)
Non-STD        ==> y      (Y|N Migrate Non-standard members)
User Exit      ==> adfmexit (Name of User Exit)
Exit Type      ==> c      (C=Clist, P=Program)
Called at      ==> i e t   (I=Initialization,E=Every Element,T=Termination)
Log Name       ==> initial.log
Checkpoint     ==> y      (Y|N Checkpoints are requested)
ENTER = PROCESS AND EXIT   END = EXIT WITHOUT PROCESSING

```

Figure 9-5. IADF Main MIGRATION Panel

Figure 9-5 is a sample of the MIGRATION panel. Since your responses to the information requested on this panel control all of migration, the requests are described in groups under the following headings.

SYSID/VERSION: Information on this line is non-modifiable. The SYSID presented is the one you carried over from previous panels. The VERSION is either carried over from previous panels or is the one currently in your profile. If no versions exist for the SYSID, the default "BASE" is used and an asterisk(*) appears beside "TO". If multiple versions exist, the "TO" version was determined from the paths table (ADFVER in IMSADF.ADFTLIB) or was selected by you in "Multiple Path Selection" on page 9-18.

Full Migration: This value may be used as a short cut to avoid element selection if you wish to migrate an entire SYSID/VERSION. If this parameter is set to "Y", the list of individual members is ignored, and all the known elements of a SYSID are migrated. If this is a MOVE request, the complete SYSID is deleted at the end of the operation.

Non-standard elements are included in a full migration request. Because it cannot be determined from the "ISPF Tables" data set what is to be migrated, you are shown the panels for this function, which is described in "Non-Standard Element" on page 9-36. Libraries for any non-standard elements must be specifically entered.

Type (form of Migration): Information presented on this line was derived from the Developers table (ssssPRad described in Chapter 6, "Adding Developers to the SYSID") for this SYSID/VERSION from which you are migrating and from the corresponding table for the SYSID/VERSION to which you are migrating. The field may or may not be modifiable, depending on your migration authorization. Your authorization may be specific or generic. The following occurs if these values were extracted or derived for both SYSID/VERSIONs.

M The field is set to "M" and is non-modifiable.

C The field is set to "C" and is non-modifiable.

B Since either form is acceptable, "C" is presented, but you may modify it if you wish.

If there is no only one version of the SYSID, the COPY function is assigned as a default, but the field is modifiable.

Note: If your authorization failed, you never see this panel.

Use LMF: The format of the subsequent panels is determined by your answer to "Use LMF". If your answer is N(o), you see the remainder of the panels in the form presented in this guide. If Y(es), they differ slightly, and are so noted on each panel description (additionally, the migration type must be "M").

Though the panels displayed in the remainder of this chapter are for the COPY function, where it is pertinent, the heading "LMF PROMOTE Differences" denotes the additional information required for LMF.

Source/Load: The values for "Source" and "Load" set initial values for all your migration elements. Each set of element panels contain the same parameters, with the values you enter on the primary panel propagated through them. If you choose to change the parameter values on the element panels, they override the primary.

Element Specification: If you have not chosen "Full Migration", enter the element types pertaining to a SYSID that you wish migrated. You may choose as many migration elements at once as you desire. After you complete the remainder of this panel, each element function is selected in sequence and performed.

The elements selected and their examples in this chapter follow:

- "System Elements" on page 9-22
- "Segment Element" on page 9-23
- "Transaction Element" on page 9-25
- "Transaction Drivers Element" on page 9-27
- "Audit Group Element" on page 9-29
- "Update Message Data Base" on page 9-31
- "Non-Standard Element" on page 9-36

In order to describe each element's migration requirements, the panels are presented to you in the following sections as though you selected only one at a time.

User Exit Information: If you choose to have a user exit, enter here its name, whether it is a clist or program, and the points at which you want it to be invoked. For more information on this topic, see “User Exit” on page 9-3 and “Migration User Exit” on page 9-9.

Log Name: Enter the name of a data set, following TSO naming conventions, on which you wish the migration audit trail saved. If this data set does not exist, migration will allocate it for you. Migration always opens the log using DISP=MOD. At the completion of Phase II, you may browse the migration log. An example of the log is shown in Figure 9-25.

Checkpoint: Enter “Y” to request checkpointing in Phase II of migration. “Checkpoint Panel” describes further information you must enter if you elect this option. Restart from abnormal error conditions is provided, regardless of your checkpoint option.

If checkpoints were requested, at this time you will see “Checkpoint Panel.”

Otherwise element selection begins with “System Elements” on page 9-22.

Checkpoint Panel

If a checkpoint was requested in Figure 9-5, Figure 9-6 is displayed.

```
ADFMGPN5 ----- MIGRATION FUNCTION -----
COMMAND ==>

*****
*          CHECKPOINT REQUEST          *
*****

SYSID:  SAMP VERSION:  BASE

A Checkpoint was Requested for this Run

Checkpoint Type ==> n          (N=Number of Records,T=CPU Elapsed Time)
Criteria       ==> 300        (Number of Records or CPU Seconds)

Instructions:
  Press ENTER   Key to Register Checkpoint Information
  Enter  END    Command to Discard Checkpoint Request
  Enter CANCEL  Command to Cancel Migration Process
```

Figure 9-6. Checkpoint Information

You may choose between a number of records processed or elapsed CPU time as a basis for the checkpoint. This provides a ‘stop after’ function. Phase II verifies this condition, and if the limit is reached, the function is **suspended**. The control file is updated with the type of restart and the starting record number.

A restart will also be scheduled by Phase II if it detects utility errors.

In Figure 9-6, the checkpoint selected is 300 records. Press ENTER.

Element selection begins with "System Elements."

System Elements

You have selected "System" COPY by entering "Y" on Figure 9-5.

SYSMGRP	-----	COPY	FUNCTION	-----
COMMAND ==>				
SYSID: SAMP LEVEL:3				
Specify Elements to be Migrated/Copied:				
System	==>	Y	(Y N SYSTEM Source)	
POM	==>	Y	(Y N Primary Option Menu)	
SOM	==>	Y	(Y N Secondary Option Menu)	
Source	==>	Y	(Y N Migrate Source)	
Load	==>	Y	(Y N Migrate Load Modules)	
COMPLETE AND PRESS ENTER OR END TO PROCESS AND EXIT				

Figure 9-7. System Copy Panel

In Figure 9-7 the SYSTEM source, Primary Option and Secondary Option menu rules are selected for copying. Any combination of them could have been chosen. "Source" and "Load" values are propagated from Figure 9-5.

LMF PROMOTE Differences: If you use LMF, you have an extra row after "System", "POM", and "SOM", which is designated "REASON". On each row you may enter a descriptive reason for the migration activity. The code must not exceed 26 characters; words are separated by periods.

When you have entered all data, press END PF key to invoke the COPY function. After the activity is complete, you are shown either the panel from "Phase I Completion" on page 9-38 or the next element panel you chose in Figure 9-5.

Segment Element

You have selected "Segments" migration by entering "Y" on Figure 9-5.

```
SEGMGRP ----- COPY      FUNCTION FOR SEGMENTS -----  
COMMAND ==>  
  
SYSID: SAMP   LEVEL:3  
  
Enter ALL on the COMMAND Line to Migrate/Copy without selection list:  
  
Source   ==> y           (Y|N Migrate Source)  
Load     ==> y           (Y|N Migrate Load Modules)  
  
  
  
  
  
  
  
  
  
  
COMPLETE AND PRESS ENTER OR END TO PROCESS AND EXIT
```

Figure 9-8. Segment Main Panel

In Figure 9-8 you may select either ALL segments associated with a SYSID or you may request a selection panel listing all the segments and may select them individually. "Source" and "Load" values are propagated from Figure 9-5.

Note: There are no panel differences for LMF users on element main panels.

Pressing ENTER displays the selection list shown in Figure 9-9.

ADFSEGMT ----- COPY
FUNCTION FOR SEGMENTS -----

COMMAND ==>

SCROLL ==>
PAGE

Source: Y
Load: Y

Line Command: Snn select

CMD	ID	Type	Source	Load
s''	M4	DBS	-	N
'''	SB	DBS		
'''	SA	DBS		
'''	M3	DBS		
'''	M1	MAP		
'''	P1	PSE		
'''	F1	BEK		
'''	O1	OUT		
'''	T1	TBL		

***** BOTTOM OF DATA *****

Figure 9-9. Segment Selection Panel

On this panel you may select any of the listed segments. The segment type, data base, mapping, etc., is listed under the "Type" column. Enter under the CMD column an "s" for individual or "snn" where "nn" is a number designating the segments, including the current one, to be copied (for example, "s3" indicates that the current segment and the next two are to be copied). You select the type of migration as shown in the example by entering Y/N under the "Source" and "Load" columns if your choice is different from the values shown at the top of the panel.

LMF PROMOTE Differences: If you use LMF, you have an extra column beside "Load", designated "REASON". On each row you may enter a descriptive reason for the migration activity. The code must not exceed 26 characters; words are separated by periods.

When you have entered all data, press END PF key to invoke the COPY function. After the activity is complete, you are shown either the panel from "Phase I Completion" on page 9-38 or the next element panel you chose in Figure 9-5.

Transaction Element

You have selected “Transaction” migration by entering “Y” on Figure 9-5.

TRXMGRP ----- COPY

FUNCTION FOR TRANSACTIONS -----

COMMAND ==>

SYSID: SAMP LEVEL:3

Enter ALL on the COMMAND Line to Migrate/Copy without selection list:

Source ==> Y

Load ==> Y

Composite ==>

STX ==>

(Y|N Migrate/Copy Source)

(Y|N Migrate/Copy Load Modules)

(Y|N Migrate/Copy Composite Modules)

(Y|N Migrate/Copy Secondary Transactions)

COMPLETE AND PRESS ENTER OR END TO PROCESS AND EXIT

Figure 9-10. Transaction Main Panel

If you want all secondary transactions or composites associated with the transaction to be included enter “Y” beside the appropriate row; otherwise they are not selected.

Press ENTER to select from a list of transactions for this SYSID.

```

ADFTRXMT ----- COPY          FUNCTION FOR TRANSACTIONS          -----
COMMAND ==>                                SCROLL ==> PAGE
Source: Y      Load: Y
Line Command: Snn select
CMD   TRXID  Type  Source  Load  STX  COMP Transaction Name
---   ---   ---   -      -      -      -  -----
'''   CD    C      -      -      -      -  CLOSE/DISBURSE
s'''  CY    C      -      -      -      -  CYCLE COUNT
'''   PA    C      -      -      -      -  PART SEGMENT
'''   IV    C      -      -      -      -  INVENTORY
'''   PD    C      -      -      -      -  STANDARD INFORMATION
***** BOTTOM OF DATA *****

```

Figure 9-11. Individual Transaction Selection Panel

In Figure 9-11 you may select any transactions to migrate by entering "s" for individual or "snn" as described previously.

The value under "Type" indicates a C(onversational) transaction. "Source" and "Load" carry the designations described in the panel. The default values for STX (secondary transaction) and COMP (Composite) is N(o), meaning no STX or COMP transactions are to be copied. Enter Y(es) if you wish any value overridden.

LMF PROMOTE Differences: If you use LMF, the "Transaction Name" column is replaced by "REASON". On each row selected, you may enter a descriptive reason for the migration activity. The code must not exceed 26 characters; words are separated by periods.

When you have entered all data, press END PF key to invoke the COPY function. After the activity is complete, you are shown either the panel from "Phase I Completion" on page 9-38 or the next element panel you chose in Figure 9-5.

Transaction Drivers Element

You have selected "Drivers" migration by entering "Y" on Figure 9-5.

```
DRVMGRP  ----- COPY      FUNCTION FOR TRANSACTION DRIVERS -----  
COMMAND ===>  
  
SYSID: SAMP  LEVEL:3  
  
Enter ALL on the COMMAND Line to Migrate/Copy without selection list:  
  
Source  ===> Y           (Y|N Migrate Source)  
Load    ===> Y           (Y|N Migrate Load Modules)  
  
Drivers will be Migrated/Copied to the RULES LOAD library shown below.  
To override, enter the data set name of another load library.  
  
RULES LOAD  
PROJECT===> ADF  
GROUP   ===> PROD  
TYPE    ===> RULLIB  
  
OTHER OUTPUT PARTITIONED DATA SETS:  
      ===>  
  
COMPLETE AND PRESS ENTER OR END TO PROCESS AND EXIT
```

Figure 9-12. Transaction Driver Main Panel

Pressing ENTER enables you to view the selection list.

Normally drivers in load module form are migrated to the RULES LOAD library displayed (you entered it on the main migration panel). If you want to override the output RULES LOAD library, enter the data set name to which you wish the driver(s) migrated. This option applies only if you requested the load module form of migration.

For performance and preloading reasons you may wish to migrate drivers to a library that will be in the STEPLIB concatenation in your IMS/VS Message Region (the driver must be loaded by the IMS/VS region controller, while rules are loaded by IMSADF II). For information on reasons for separating transaction drivers from other rules, please refer to the *IMS Application Development Facility II Version 2 Release 2 Installation Guide* under the chapter titled "Performance".

Note: Separation of rules is not applicable to CICS users since all modules in CICS are loaded from the //DFHRPL DD concatenation.

```

ADFDRVMT ----- COPY          FUNCTION FOR TRANSACTION DRIVERS -----
COMMAND ==>                      SCROLL ==> PAGE
Source: Y      Load: Y
Line Command: Snn select
CMD      PGMID  Type  Source  Load  Program Description
---      ---   ---   ---    ---   -----
s''      OR     C           STANDARD DRIVER

''       CD     C           CYCLE COUNT/SPR DRIVER
***** BOTTOM OF DATA *****

```

Figure 9-13. Driver Selection Copy Panel

In Figure 9-13 you see the driver name and the type (conversational, non-conversational, or batch). "Source" and "Load" carry the designations shown at the top of the panel.

LMF PROMOTE Differences: If you use LMF, "Program Description" is replaced by "REASON", on which you may enter a descriptive reason for the migration activity. The code must not exceed 26 characters; words are separated by periods.

When you have entered all data, press END PF key to invoke the COPY function. After the activity is complete, you are shown either the panel from "Phase I Completion" on page 9-38 or the next element panel you chose in Figure 9-5.

Audit Group Element

You have selected "AGROUPS" migration by entering "Y" on Figure 9-5. The "Load" form of migration activity affects ONLY static audit rules. If you wish to have the High Level Audit Language (HLAL) source migrated, you must also select "Source", whether on this panel or on Figure 9-5.

AUDMGRP	-----	COPY	FUNCTION	-----
COMMAND	====>			
SYSID: SAMP LEVEL:3				
Enter ALL on the COMMAND Line and/or Elements to be Migrated/Copied:				
Transactions	====>	Y	(Y N Transaction Related Audits)	
Segments	====>	N	(Y N Segment Related Audits)	
Subroutines	====>	N	(Y N Subroutine Audits)	
Other AGROUPS	====>			
	====>			
	====>			
	====>			
Source	====>	Y	(Y N Migrate Source)	
Load	====>	Y	(Y N Migrate Load Modules)	
COMPLETE AND PRESS ENTER OR END TO PROCESS AND EXIT				

Figure 9-14. Audit Group Selection Copy Panel

In Figure 9-14 you may select any combination of transaction or segment audits or audit subroutines to be migrated. Those audits are developed under the IADF naming convention.

If you have existing audit load modules under a standard IMSADF II naming convention, that is, sssnnnn, where 'ssss' is the SYSID and 'nnnn' is audit group, you may enter the latter value beside "Other AGROUPS." If you enter eight-character names, IADF assumes they are subroutine names and accepts them. You may specify up to 20 audit groups at once.

Each type of static audit rule has a selection panel similar to the one shown in Figure 9-15. The panels vary only in the title at the top (such as SEGMENT, TRANSACTION, SUBROUTINE, or OTHER). You are shown as many audit selection panels as the types of audit groups you select on this panel. For brevity, the only one selected here is for transaction audits.

```

ADFMGAG --- COPY      FUNCTION FOR TRANSACTION AUDITS      -----
COMMAND ==>          SCROLL ==> PAGE
Source: Y    Load: Y
Confirm AGROUP Name ==> N
Line Command: Snn select
CMD Member Source Load Description
-----
''' AGT1          TRANSACTION T1 GROUP
''' AGT3
''' AGYY          FIRST AUDIT GROUP
***** BOTTOM OF DATA *****

```

Figure 9-15. Transaction Audit Selection Copy Panel

In Figure 9-15 you may select the audits you wish to be migrated by entering "s" or "snn" under the CMD column. In addition indicate whether you wish to view their actual audit group names by entering "Y" beside "Confirm AGROUPS." The migration function examines each element chosen for the HLAL keyword AGROUP=value in order to migrate the correct audit group(s). Typically, you would only wish this function if you have developed audit rules outside of IADF. "Source" and "Load" need not be entered if the values on the top of this panel are correct.

LMF PROMOTE Differences: If you use LMF, the "Description" column is replaced by "REASON". On each row you may enter a descriptive reason for the migration activity. The code must not exceed 26 characters; words are separated by periods.

When you have entered all data, press END PF key to invoke the COPY function. After the activity is complete, you are shown either the panel from "Phase I Completion" on page 9-38 or the next element panel you chose in Figure 9-5.

Update Message Data Base

You have selected "Message" migration by entering "Y" on Figure 9-5.

```
ADFMRG2P---- MIGRATE COPY      FUNCTION FOR MESSAGES, ROUTES, SDT -----
COMMAND ==>

SYSID              ==> SAMP

Specify Messages, Help, Routes, SDT to Migrate


Help Panel(s)      ==>          (Blank to select, N to skip selection)
Message(s)          ==>          (Blank to select, N to skip selection)
Alternate Routes    ==>          (Blank to select, N to skip selection)
Destination(s)      ==>          (Blank to select, N to skip selection)


COMPLETE AND PRESS ENTER OR END TO PROCESS AND EXIT
```

Figure 9-16. Message Data Base Main Panel

Figure 9-16 is the Message data base control panel on which are listed its four categories.

Leave the value blank if you want the selection list of the category: messages, help panels, alternate routes and secondary destinations. If a category is not desired, enter "N" beside it.

There is a processing distinction between pressing ENTER and END PF key. There are two ways to migrate these message data base elements:

1. Migrate all selected elements as one member.
2. Migrate elements within groups as multiple members.

Some users prefer to group message elements by transaction. To do this, you must know which elements in the four categories apply to a given transaction.

Migration services allow you to migrate Message data base elements in either way.

If you press END PF key, selection panels are presented for each desired category. You select those you wish to migrate in this invocation of migration services, regardless of transaction code or other criteria. All selected elements are migrated as **one** member to the dynamic source library, and Figure 9-16 is never shown to you again.

If you press ENTER, migration assumes that you wish to group elements by transaction, or other criteria. Selection panels are presented for each desired category. You select those you wish to migrate in a given group. After you complete all selected categories, Figure 9-16 is presented again, so you can choose elements for the next group. Each time you press ENTER, a new source member is

created to be migrated to the receiving library. This processing continues until you press END PF key, whereupon you are shown the categories for the last time.

All elements in the Message data base are migrated in source form only. Later, you must invoke the batch driver to add the elements to the Message data base. The names of these source member(s) in the receiving dynamic source library always start with the SYSID, the characters "MS" and a sequential number from one to 99, such as "SAMPMS1".

Note: Any time you select Message data base migration, the member replaces any which currently exist.

In this example it is assumed that blanks were left in all fields and that the END key was pressed. (If you enter "N" beside any categories, you will see a subset of those shown in sequence in this topic).

After the processing of this panel and its dependent panels is complete, you are shown either the panel from "Phase I Completion" on page 9-38 or the next element panel you chose in Figure 9-5.

LMF PROMOTE Differences: There are no differences in this function for LMF.

Help Panels

The first panel (see Figure 9-17) to be displayed contains all the help panels in this version.

```

HLPMPGR ----- MIGRATE HELP PANELS IN SAMP  BASE  -----
COMMAND ===>                                     SCROLL ==> PAGE

Enter ALL on the COMMAND line to Move/Copy all the Help Panels

Line Command: Snn Select
C   ID  S Type TRX  HEADER and first line of panel (58 characters)
--- --- - - - - -
S   0001   6   T1  HELP          TRANSACTION T1          HELP
                        ENTER THE DATA ACCORDING TO THE FOLLOWING RULES:
''' 0002   6   T2  HELP          TRANSACTION T2          HELP
                        ENTER THE FOLLOWING INFORMATION
''' 0003   5   CE  HELP          SECONDARY KEY SELECTION
                        USE THIS PANEL TO ENTER:
''' 0004   2           HELP  FOR SIGNON          HELP
                        HELP
''' 0005   6   02  HELP          CREDIT CARD OPERATION      HELP
                        ----- HELP -----
''' 0006 2   6   02  HELP          CREDIT CARD OPERATION      HELP
                        SECOND PAGE FOR THE CREDIT CARD OPERATION LINE 1
''' 0007 1   6   02  HELP (?)      CREDIT CARD OPERATION      HELP
                        ----- HELP -----
''' 0008 2   6   02  HELP          CREDIT CARD OPERATION      HELP

```

Figure 9-17. Help Panel Migration

Use the "s" or "snn" commands to select any number of elements. They are migrated individually to their corresponding table (if the target is a multiple version SYSID), and as a group to the dynamic source library.

Messages

Next the messages are displayed as shown in Figure 9-18.

```
MSGMGR ----- MIGRATE MESSAGES IN SAMP BASE -----
COMMAND ==> SCROLL ==> PAGE

Enter ALL on the COMMAND line to Move/Copy all the Messages

Line Command: Snn Select
C MSGID Length First two lines of text (62 characters only)
-----
''' 0000 58 PROCESSING LOGIC ERROR FOR TRXID=@@@. DL/I STATUS CODE=--.

''' 0141 110 DATA CONVERSION ERROR OCCURRED DURING MAPPER CALL
AUDITED FIELD IS ! . SEQ# = .
''' 0142 0140 SEGMENT TO BE USED IN MAPPER OR COPYSEG CALL WAS NOT FOUND
AUDITED FIELD IS . SEQ# = .
''' 0143 0140 SEGMENT NOT FOUND WHILE ATTEMPTING TO TEST CHANGE FLAG
AUDITED FIELD IS . SEQ# = .
''' 0444 33 TEST IF AN ERROR @@@@ OCCURS |||

s''' 0500 980 INVALID CODE RETURNED FROM DL/I (**) VERIFY DATA BASE
OTHER MESSAGE MIGHT BE FOUND IN $$$
''' 0501 52 BALANCE MUST BE BETWEEN 1 AND 20 (''''') IS INVALID

''' 0502 46 DATA BASE IN ERROR WHEN INSERTING CUSTOMER: ____

''' 0503 20 NEW MESSAGE TO ADD ?

''' 0504 12 ANY MESSAGES

''' 0505 33 MESSAGE WITHOUT VARLIST OR FIELDS
```

Figure 9-18. Messages Migration

Use the "s" or "snn" commands to select any number of elements. They are migrated individually to their corresponding table (if the target is a multiple version SYSID), and as a group to the current source member.

Alternate Routes

The alternate routes are displayed in Figure 9-19.

```

ROUMGR ----- MIGRATE ROUTES WITHIN SAMP MFC1 -----
COMMAND ==>                                SCROLL ==> PAGE

Enter ALL on the COMMAND line to Move/Copy all the Alternate Routes

Line Command: Snn Select
C ROUTE ID  KEY      PGROUP USERID  MSG1  MSG2  MSG3  MSG4  MSG5  Additional PG2
- - - - -
s'' 01      SS##PG## PG      999999 0400          Y
'' 02      SS##PG## TL      999988 0455          Y
'' 03      SS##PG## XX      777777 0200          Y
'' 04      W0##### TX      111111 0334          Y
'' 21      SS##PG## PG      999999 0400          Y
'' 22      SS##PG## PG      999999 0400          Y
***** BOTTOM OF DATA *****

```

Figure 9-19. Alternate Routes Migration

Use the "s" or "snn" commands to select any number of elements. They are migrated individually to their corresponding table (if the target is a multiple version SYSID). Routing source is added to the current source member produced by help panels and messages.

Secondary Destinations

Finally the secondary destinations are presented as shown in Figure 9-20.

```
SDTMGR ----- MIGRATE SECONDARY DESTINATIONS IN SAMP MFC1 -----
COMMAND ===>                                SCROLL ===> PAGE

Enter ALL on the COMMAND line to Move/Copy all the Secondary Destinations

Line Command: Snn Select
C OUTID LT Destination Comments
-----
s'' 0Q Y LTERM01A ANY
''' 01 Y LTERM01A ANY COMMENT
''' 02 Y LTERM02A SECOND
''' 03 LTERM03A THIRD
''' 04 Y LTERM03B LAST
***** BOTTOM OF DATA *****
```

Figure 9-20. Secondary Destinations Migration

Use the "s" or "snn" commands to select any number of elements. They are migrated individually to their corresponding table (if the target is a multiple version SYSID).

When you have entered all data, press END PF key. The source is added to the current member produced by previous message data base panels. It is migrated during Phase II as a non-standard element to the dynamic source library.

After the activity is complete (you pressed END PF key on Figure 9-16, you are shown either the panel from "Phase I Completion" on page 9-38 or the next element panel you chose in Figure 9-5.

If you pressed ENTER on Figure 9-16, you begin the next iteration of Figure 9-16 processing.

Non-Standard Element

You have selected "Non-STD" migration by entering "Y" on Figure 9-5 or by selecting "Full Migration."

```
ADFMGPNS ----- COPY      FUNCTION FOR NON-STANDARD MEMBERS -----
COMMAND ===>

SYSID ===> SAMP    LEVEL:3

      RULES SOURCE (1)      RULES LOAD (2)      AUDIT SOURCE (3)
PROJECT ===> ADF          PROJECT ===> ADF          PROJECT ===> ADF
GROUP   ===> PROD         GROUP   ===> PROD         GROUP   ===> PROD
TYPE    ===> RULSRC       TYPE    ===> RULLIB       TYPE    ===> DYNsrc

OTHER PARTITIONED DATA SETS:
(1) RULES SOURCE  ===>
(2) RULES LOAD    ===>
(3) AUDIT SOURCE  ===>

Select one of the above Output libraries: (1-3) ===> 1

Or Specify another Output Library:
PROJECT ===>
GROUP   ===>
TYPE    ===>
OTHER PARTITIONED DATA SET:
      ===>
COMPLETE AND PRESS ENTER IF MORE INPUT IS REQUIRED OR END TO PROCESS AND EXIT
```

Figure 9-21. Designate Output Library for Non-Standard Elements

In Figure 9-21 you are to indicate your output libraries for the non-standard elements you wish to copy. This function, a combination of IADF and ISPF function, is similar to ISPF MOVE/COPY. You may migrate only one form of exit or special processing routine at a time. Observe that the two rows of libraries at the top are protected. You entered the appropriate row when you completed the information for Figure 9-5. You must designate the library, which is either one of the numbered libraries displayed or another library you may name now. The libraries of the form PROJECT/GROUP/TYPE must be designated if you use LMF PROMOTE; other users may enter any appropriate type.

When you have finished, press ENTER.

```

ADFMGPN1 ----- COPY      FUNCTION FOR NON-STANDARD MEMBERS -----
COMMAND ==>

SYSID ==> SAMP  LEVEL:3

Specify Input Data Set
PROJECT ==> TS0ID1
GROUP   ==> TEST
TYPE    ==> EXITS

OTHER INPUT PARTITIONED DATA SET:
==>

COMPLETE AND PRESS ENTER TO CONTINUE OR END TO EXIT

```

Figure 9-22. Input Non-Standard Element Library

In Figure 9-22 enter your input library for the type of exit you wish to be copied. The same rules apply for entering the input library as for the output on the previous panel.

When you have finished, press or ENTER.

```

ISRFPM1D: TS0ID1.TEST.EXITS -----
COMMAND ==>

```

										SCROLL ==> PAGE	
NAME	LIB	VV	MM	CREATED	LAST	MODIFIED	SIZE	INIT	MOD	ID	
DBD1	1	01.02	84/02/21	84/03/13	13:21		50	49	0	TS0ID1	
MFC1EX2	1	01.13	84/07/06	84/09/12	18:47		392	378	28	TS0ID1	
MFC1EX5	1	01.05	84/07/09	84/09/12	18:48		378	377	0	TS0ID1	
MFC1W01W	1	01.00	84/07/23	84/07/23	17:54		16	16	0	TS0ID1	
MFC1SAMP	1	01.00	84/08/31	84/08/31	19:35		18	18	0	TS0ID1	
NE	1	01.00	84/06/21	84/06/21	13:18		25	25	0	TS0ID1	
NEP											
PRMOSICD	1	01.01	84/09/12	84/09/12	17:21		59	59	0	TS0ID1	
SAMPSICD	1	01.01	84/09/12	84/09/12	13:46		59	59	0	TS0ID1	
TEST	1	01.01	84/06/14	84/06/14	11:28		2	2	0	TS0ID1	
TESTQ	1	01.01	84/09/04	84/09/04	11:23		59	59	0	TS0ID1	
TEST1	1	01.09	84/03/07	84/03/30	11:48		33	33	0	TS0ID1	
TEST1X											
s TEST2	1	01.01	84/01/23	84/06/21	13:10		43	330	0	TS0ID1	
TEST3	1	01.01	84/01/23	84/06/21	13:10		29	29	0	TS0ID1	
TEST3A											
TEST4	1	01.02	84/01/24	84/03/27	13:11		70	72	0	TS0ID1	
TEST4A											
TEST4B											
TEST5	1	01.03	84/01/24	84/08/30	12:18		50	49	0	TS0ID1	
TEST5C											

Figure 9-23. Selection List for Non-Standard Element COPY

Figure 9-23 shows a directory list from the input library you named. You may select any members you wish to migrate, one at a time. Observe that it is a panel from the ISPF product because it is prefixed by "ISR". Enter an "s" beside any member, then press ENTER. Select each succeeding member.

Note: If you should erroneously select multiple members at one time, all but the first is ignored.

When the members have been selected, you have completed the data gathering portion of migration. Refer to "Phase I Completion" to start Phase II migration.

Since you can use this panel to migrate from any given library to any other, you could use this function to migrate the DBD/PSB/ACB DL/I control blocks. If you wish to do this, you must invoke "Non-Std" service once for each library from which you wish to migrate members.

LMF PROMOTE Differences: The only LMF differences on this activity are in the designation of libraries of the form PROJECT/GROUP/TYPE.

Phase I Completion

Once the data collection is completed, Figure 9-24 is presented, signaling the end of Phase I with no errors.

Note: This panel may be displayed as a result of a restart condition, so that you may complete Phase II.

```
ADFSUBP ----- JOB SUBMISSION -----Phase II ready to run
OPTION ==>
Migrate Phase II is ready to execute, select 3 to submit
SYSID: SAMP      PGROUP: PG      LEVEL:3

1      EDIT THE FILE BEFORE SUBMISSION
2      BROWSE THE FILE BEFORE SUBMISSION
3      SEND THE JOB TO BE EXECUTED
C      CANCEL THE SUBMISSION

ENTER OPTION NUMBER OR CANCEL/PF3 (END) TO EXIT
```

Figure 9-24. Job Submission Panel

If you select "3", the job (Phase II) is submitted to batch.

Phase II performs the actual migration operation, updating the log with the result of the operations. An example of the job stream may be found in Appendix C, "Migration Phase II Sample JCL."

The name of the control file (unless changed by an user exit) is MIGRATE.CNTL prefixed by the TSO userid of the requestor.

You may want to add comments in this file to be placed in the migration output listing. You can edit the control file and update it appropriately (see “Format of the Control File” on page 9-6). Any record that is not type 2 is ignored and printed in the listing.

After the job is completed, you may browse the log file or recover the listing of Phase II to verify the results as shown in Figure 9-25.

If catastrophic errors occur of which Phase II is unaware, it cannot update the control file. An 'X37' abend or the job's being cancelled are examples of this condition.

You must be able to reset the control file to perform restart processing. You do this by editing the control file and updating the restart record yourself, determining where you wish to restart. Refer to “Format of the Control File” on page 9-6.

Migration Log

```
ISRBROBF- TSOID1.INITIAL.LOG ----- LINE 000801 COL    001 080
COMMAND ==>                                SCROLL ==> CSR
***** TOP OF DATA *****
IMSADF II                                COPY FUNCTION
DATE: 86/09/24 TIME: 11:07 ACTIVITY LOG          USER: TSOID1
COPY DONE FOR MEMBER MFC1SAMP 09/24/86 11:07:52
SYSTEM SAMP HAS BEEN COPIED 09/24/86 11:07:53
COPY DONE FOR MEMBER SAMPSYS 09/24/86 11:07:59
COPY DONE FOR MEMBER SAMPPM 09/24/86 11:08:02
PRIMARY OPTION MENU HAS BEEN COPIED 09/24/86 11:08:02
COPY DONE FOR MEMBER SAMPSMSS 09/24/86 11:08:05
SECONDARY OPTION MENU COPIED 09/24/86 11:08:05
COPY DONE FOR MEMBER SAMPBEDK 09/24/86 11:08:11
COPY DONE FOR MEMBER SAMPSRDK 09/24/86 11:08:15
SEGMENT DK DID NOT EXIST IN TARGET SYSTEM, IT WAS INSERTED 09/24/86 11:08:15
SEGMENT DK HAS BEEN COPIED 09/24/86 11:08:16
COPY DONE FOR MEMBER SAMPTRCH 09/24/86 11:08:28
COPY DONE FOR MEMBER SAMPTLCH 09/24/86 11:08:31
TRANSACTION CH HAS BEEN COPIED 09/24/86 11:08:33
SECONDARY TRANSACTION SA01XX01 NOT COPIED 09/24/86 11:08:33
SECONDARY TRANSACTION SAHRXX01 NOT COPIED 09/24/86 11:08:34
SECONDARY TRANSACTION SAXX01 NOT COPIED 09/24/86 11:08:34
COPY DONE FOR MEMBER SAMPTRAA 09/24/86 11:08:42
TRANSACTION AA HAS BEEN COPIED 09/24/86 11:08:44
COPY DONE FOR MEMBER SAMPPGAA 09/24/86 11:08:49
TRANSACTION DRIVER AA COPIED 09/24/86 11:08:50
COPY DONE FOR MEMBER SAMPAGCC 09/24/86 11:08:54
COPY DONE FOR MEMBER SAMP0001 09/24/86 11:08:57
AUDIT GROUP SAMPAGCC COPIED 09/24/86 11:08:57
A MESSAGE, HELP PANEL, SECONDARY DESTINATION OR ROUTE HAS BEEN MOVED OR COPIED 0
A MESSAGE, HELP PANEL, SECONDARY DESTINATION OR ROUTE HAS BEEN MOVED OR COPIED 0
A MESSAGE, HELP PANEL, SECONDARY DESTINATION OR ROUTE HAS BEEN MOVED OR COPIED 0
A MESSAGE, HELP PANEL, SECONDARY DESTINATION OR ROUTE HAS BEEN MOVED OR COPIED 0
COPY DONE FOR MEMBER SAMPMS1 09/24/86 11:09:04
COPY DONE FOR MEMBER SAMPMS1 09/24/86 11:09:04
COPY DONE FOR MEMBER IKEADUMP 09/24/86 11:09:08
COPY DONE FOR MEMBER IKEADUMP 09/24/86 11:09:08
MEMBER MFC1SAMP DELETED FROM LIBRARY TSOID1.TEST.SKELT 09/24/86 11:09:09
MEMBER SAMPBEDK DELETED FROM LIBRARY TSOID1.TEST.SKELT 09/24/86 11:09:09
MEMBER SAMPPGAA DELETED FROM LIBRARY TSOID1.TEST.SKELT 09/24/86 11:09:09
MEMBER SAMPSYS DELETED FROM LIBRARY TSOID1.TEST.SKELT 09/24/86 11:09:09
MEMBER SAMPTRAA DELETED FROM LIBRARY TSOID1.TEST.SKELT 09/24/86 11:09:09
MEMBER SAMPTRCH DELETED FROM LIBRARY TSOID1.TEST.SKELT 09/24/86 11:09:09
NORMAL TERMINATION REACHED, NO CATASTROPHIC ERRORS FOUND 09/24/86 11:09:10
***** BOTTOM OF DATA *****
```

Figure 9-25. Sample Phase II Log

Figure 9-25 is a sample of the migration log at the completion of Phase II, and does not relate to migration activities depicted in this chapter.

This data set was specified in the migration menu and is always accessed by migration services using DISP=MOD. The log also appears as SYSOUT.

Chapter 10. Additional Administration Services

This chapter describes certain administrator functions which are not required at any particular time, but which may prove helpful to you. They are:

- Establish an ISPF User Profile for IADF development
- Convert existing rules source to IADF table format using the Extract function
- View the IMSADF II installed product options
- Extend the Glossary function
- Change an IADF ISPF Table

Establish ISPF User Profile for IADF

The ability to establish a user profile for ISPF/IADF is available to ALL users of all levels.

This step is optional. Product options for IADF generation purposes are extracted from the options specified when the IMSADF II product was installed. The only reason you may want to establish your user profile (ISPF) is if you intend to do development on this SYSID. Otherwise you can leave this task to the developers.

Each user changes his profile by overriding values for the IMSADF II product version through these panels. The results are stored in the ISPPROF data set for his TSO userid in a member called "ADF1PROF" and is used to preserve his IADF options from TSO session to session. Accordingly there is only one profile per user, no matter how many SYSIDs he is allowed to use. The options stored in the profile are primarily Rules Generator values which are used to develop application systems. The developer may still have to enter some values, if all SYSIDs in your installation to which he is assigned do not carry the same overrides.

```

IADFADMN ----- IADF ADMINISTRATION MAIN MENU -----
OPTION ==> 6

ADMINISTRATIVE LEVEL:3
KEY ==>                                SYSID ==> SAMP    LEVEL: 3

  1 SYSTEMS      - Create or modify IMSADF II  systems (SYSID)
  2 MIGRATION    - Perform migration    TYPE ==> C  ( M Move  C Copy )
  3 ADMINISTRATORS - Specify administrative users
  4 TABLES      - Create, modify, and/or copy ISPF tables
  5 GLOSSARY     - Update glossary terms
  6 DEVELOPMENT  - Perform IMSADF II  development tasks
  X EXIT        - Terminate IADF

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 10-1. Select DEVELOPMENT

Enter 6 on the option line of the IADF Administration Main Menu to get to the IADF Main Menu used by IADF developers.

```

IADFMAIN ----- IADF MAIN MENU -----
OPTION ==> 5

KEY ==>                                SYSID ==>          LEVEL:

  1 DEFINITION   - Create, modify or browse application information
                  (Data Layouts, Panels, Transactions, Audits, Messages,
                  Userids and Profiles, Transaction Drivers, SYSIDs)
  2 GENERATION   - Perform application generation
  3 TEST         - Prepare for and perform test (PROTOTYPE, BTS)
  4 UTILITIES    - Perform IADF Utilities
                  (Extract, Submit, Glossary, Documentation, SYSID Info,
                  Trace, GFIND, MFS Utility, Data Dictionary, DB2 RGLGEN)
  5 IADF PROFILE - Create or modify IADF User Profile
  6 TUTORIAL     - Display information about IADF
  X EXIT        - Terminate IADF DEVELOPMENT

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 10-2. IADF Main Menu

To create an IADF user profile, enter 5 on the option line of the IADF Main Menu.

```

ADFPRO2 ----- IADF USER PROFILE----- PAGE 1 OF 2
COMMAND ==>

ADFID:      MFC1
LEVEL:      3
ENVIRONMENT: I      (I=IMS,C=CICS)      DB Access Method: DL/I
System ID    ==> SAMP
Version ID   ==> BASE
Project/Group ==> PG
Key Audit Names ==> S      (S=standard,A=alternate)
Increase No. of Labels ==> N      (Y|N)
Amount of Increment ==> 1
Programming Language ==> C      (C=COBOL,P=PLI,A=ASSEMBLER)
DBDGEN Procedure Name ==> DBDGEN
PSBGEN Procedure Name ==> PSBGEN
ACBGEN Procedure Name ==> TESTACB
Browse Source ==> N      (Y|N when source is generated)
Systems to Search ==> samp prod      ==> mast
Submit Type  ==> f      (B=batch,F=foreground)
USRLANG      ==> E      (User language for SYSID)
Foreground Selection ==> Y      (Y|N Before Execution)
Assembler Source ==> N      (Y|N Source for SOM Rule)
Reset PROFILE ==> N      (Y|N)
COMPLETE AND PRESS ENTER OR END KEY FOR NEXT PAGE, OR ENTER CANCEL TO EXIT

```

Figure 10-3. First Page of User Profile Panel

The first of two panels listing general defaults is displayed in Figure 10-3. Some of the values are not modifiable using IADF (they may only be changed by following procedures in the *IMS Application Development Facility II Version 2 Release 2 Installation Guide* for post-installation customization using the IMSADF II Installation Dialogs). Those values are protected and are designated by a colon(:) to the right of the value name and prior to the value itself. The values are protected in order to prevent you from unintentionally introducing discrepancies between your installed ADFID and your IADF.

The "ENVIRONMENT" and "DB Access Method" (both protected values) refer to one of the four major environments in which IMSADF II can be installed:

- IMS/VS using DL/I for the Dynamic rules data bases
- CICS using DL/I for the IMSADF II Dynamic rules data bases
- IMS/VS using DB2 for the IMSADF II Dynamic rules data bases
- CICS using DL/I for the IMSADF II Dynamic rules data bases

Note: You may develop applications using DB2 regardless of the "DB Access Method" as long as you have that product.

Modifiable values are shown with an arrow to the right of the value name. To modify any of these defaults, new values are keyed over the old. For example, you might wish to change the name of the ACBGEN procedure name, so that you might place your ACBs in a test ACBLIB. The change is shown the example highlighted.

The original values in your user profile are determined by the defaults of the ADFID you access. These values may be modified to establish your profile. This profile is valid across all SYSIDs and is used for saving your preferred values across TSO sessions. Some of the SYSIDs you access may have different overrides. The

overrides are normally cataloged procedure names or values which are named specifically in your Rules Generator input.

"Systems to Search" gives you an opportunity to name up to two SYSID/VERSIONs to be searched for elements in addition to the current one. If you establish a SYSID in which all your "master rules" are placed, developers may access them easily using this parameter. Of course, update access for a "master" should be limited to an administrator only.

Key "F" for submit type to select foreground program invocation. The following parameters are used only with foreground invocation:

- If "Browse Source" is "N", you do not view generated input. This is helpful at migration time.
- Enter "N" for Assembler source to avoid viewing the assembler listing for the foreground invocation of the Rules Generator.
- If the "Foreground Selection" value is "Y", you are presented with a confirmation panel prior to foreground invocation, so that you may elect to abort the invocation.

You may reset your profile by entering "Y" beside "Reset Profile" any time you wish to synchronize it with current options. If your systems programmer changed installation options in the production system, your existing profile does not reflect those changes. Developers may not wish to reset their profiles if, for example, their nodes and procedures name test values.

When you are finished, scroll DOWN.

```

ADFPR03 ----- IADF USER PROFILE ----- PAGE 2 OF 2
COMMAND ==>

ADF SYSTEM ID: MFC1

SCREEN CHARACTERISTICS
Device Name(s)          ==> 2
Device Type(s)          ==> 2
Device Characteristics   ==> 0
MFS Trailer:            1
Format for Literals      ==> D          (L=left,R=right,D=dash)
MFSUTL Procedure Name    ==> MFSUTL

DATA SET CHARACTERISTICS
Node for Shared Files    ==> IMSADF      (ADFNODE)
For Non-shared Files     ==> ADF         (NEWADF)
IMS/VS Reslib Node       ==> IMSVS.TEST (IMSNODE)
IMS/VS DBD/PSB Node      ==> IMSVS.TEST (IMSTEST)
CICS Node                ==>             (CICNODE)
Rules Library DDNAME:    RULES
Composite DDNAME:        COMPS
Esoteric Name (DASD)     ==> SYSDA      (for temporary space)
IADF Advanced Menu       ==> N          (for advanced users)
Bypass Permanent Source ==> N          (do not search for source)
PRESS ENTER FOR NEXT PAGE, END KEY TO PROCESS AND EXIT, ENTER CANCEL TO EXIT

```

Figure 10-4. Second Page of User Profile Panel

The remaining user profile values are displayed on the second panel. For example,

```
DEVNAME=(A2,A4)
DEVTYPE=(4,2)
DEVCHRS=(0,0)
```

overridden in the table for this ADFID result in Rules Generator transaction input with those specific values. In that way, naming of those values specifically by the application developer is eliminated.

The IADF Main Menu (Advanced User) eliminates menu selections. You can execute all IADF functions with a maximum of two panel levels. In most cases you execute the function directly. To activate the IADF Main Menu (Advanced User), you enter a "Y" for "IADF Advanced Menu". It is recommended that you use the Advanced User menu particularly during migration (see "Migration Menu Selection" on page 9-14), because it allows the SYSID and VERSION to be changed directly. It is also the only way an **authorized** developer can invoke migration services, since the option appears on the Administration Main Menu normally.

"Bypass Permanent Source" is an option which can be used to tell IADF to build elements for both Rules Generator and for migration services from tables only. Normally the Rules Generator selects source over tables if source is present during developmental activities. Since IADF activities update tables, there is opportunity for source and tables to be out of synchronization if source is created and generations are performed. Setting this indicator to "Y" avoids possible conflicts for the generation process.

"Bypass Permanent Source" is also used in migration to improve performance. It is faster at that time to build temporary source from tables than it is to search libraries to determine if the source exists or not. If you are involved in migration, you should set this option to "Y". In addition, it avoids the same conflicts between source and tables as the developer encounters, and it is especially important at migration that tables are migrated correctly to the next SYSID/VERSION.

Verify that remaining values are valid for your conditions.

Press END.

```

IADFMAIN ----- IADF MAIN MENU ----- User profile updated
OPTION ==>

KEY ==>                                SYSID ==> SAMP    LEVEL: 3

  1  DEFINITION      - Create, modify or browse application information
                        (Data Layouts, Panels, Transactions, Audits, Messages,
                        Userids and Profiles, Transaction Drivers, SYSIDs)

  2  GENERATION      - Perform application generation

  3  TEST            - Prepare for and perform test (PROTOTYPE, BTS)

  4  UTILITIES       - Perform IADF Utilities
                        (Extract, Submit, Glossary, Documentation, SYSID Info,
                        Trace, GFIND, MFS Utility, Data Dictionary, DB2 RGLGEN)

  5  IADF PROFILE    - Create or modify IADF User Profile

  6  TUTORIAL        - Display information about IADF

  X  EXIT            - Terminate IADF DEVELOPMENT

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 10-5. Successful Product Overrides for SYSID

A confirmation message is displayed.

Extract ISPF Tables from Rules Source

The ability to extract ISPF tables required for IADF development from existing rules source is available to **ALL** users of all levels. This function is particularly appropriate to existing IMSADF II users who have many SYSIDs developed using prior releases. You can select this service to convert static rule source to IADF development format.

Whatever input you provide to Extract is converted to IADF format. That is, if the Rules Generator source input is a complete application system, Extract produces a complete application system. Since the Rules Generator program is invoked on behalf of Extract, the rules you specified for generation are created.

The prerequisite for this function is that you have established your SYSID as has been shown previously in this guide.

If you use screen image members (not imbedded in the Rules Generator input), you must insure that they are present in the screen image library you defined for the SYSID in Figure 5-4 before you invoke Extract.


```

IADFADMN ----- IADF ADMINISTRATION MAIN MENU -----
OPTION ==> 6

ADMINISTRATIVE LEVEL:3
KEY ==>                                SYSID ==> SAMP  LEVEL:3

  1 SYSTEMS      - Create or modify IMSADF II  Systems
  2 MIGRATION    - Perform migration    TYPE ==> M ( M Move  C Copy )
  3 ADMINISTRATORS - Specify Administrative Users
  4 TABLES      - Create, modify, and/or copy ISPF Tables
  5 GLOSSARY     - Update glossary terms
  6 DEVELOPMENT  - Perform IMSADF II  Development tasks
  X EXIT         - Terminate IADF

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 10-6. Select DEVELOPMENT

Enter 6 on the option line of the IADF Administration Main Menu to get to the IADF Main Menu used by IADF developers.

```

IADFMAIN ----- IADF MAIN MENU -----
OPTION ==> 4

KEY ==>                                SYSID ==> SAMP  LEVEL: 3

  1 DEFINITION   - Create, modify or browse application information
                  (Data Layouts, Panels, Transactions, Audits, Messages,
                  Userids and Profiles, Transaction Drivers, SYSIDs)
  2 GENERATION   - Perform application generation
  3 TEST         - Prepare for and perform test (PROTOTYPE, BTS)
  4 UTILITIES    - Perform IADF Utilities
                  (Extract, Submit, Glossary, Documentation, SYSID Info,
                  Trace, GFIND, MFS Utility, Data Dictionary, DB2 RGLGEN)
  5 IADF PROFILE - Create or modify IADF User Profile
  6 TUTORIAL     - Display information about IADF
  X EXIT        - Terminate IADF DEVELOPMENT

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 10-7. IADF Main Menu

To see the Utilities Menu, enter 4 on the option line of the IADF Main Menu.

```

IADFUTIL ----- IADF UTILITIES MENU -----
OPTION ==> 1

KEY ==>                                SYSID ==> SAMP  LEVEL:3

1  EXTRACT      - Update IADF with info from DBD or IMSADF II  source
2  SUBMIT       - Submit generation of rules
3  GLOSSARY     - Display descriptions for IMSADF II  terms
4  DOCUMENTATION - Obtain rules documentation for IMSADF II  definitions
5  SYSID INFO   - Display library names for SYSID
6  TRACE       - IMSADF II  Trace Facility
7  GFIND       - Perform a global search in the members of a PDS
8  MFS UTILITY  - Generate Message Format Service (MFS) Formats
9  DATA DICTIONARY- Perform ADFIN, ADFOUT, DBDOUT, PSBOUT, or reports
10 DB2 RGLGEN   - Create IMSADF II  source from DB2 Catalog
                  RGLGEN Execution Type ==> B  (F=Foreground, B=Batch)

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 10-8. IADF Utilities Menu

Enter 1 on the option line of the IADF Utilities Menu to select "EXTRACT".

There are two basic types of rules source: Static Rules and Dynamic Rules. IADF provides the capability to update the ISPF tables used by IADF to contain the information stored in both types of rules source.

Dynamic Rules Extraction

The dynamic rules extraction process is discussed in this topic. For static rule information, please see “Static Rules Extraction” on page 10-16.

```
ADFUEX ----- IADF EXTRACT MENU -----
OPTION ==> 2

SYSID ==> SAMP   PGROUP ==> PG   LEVEL: 3           USERID - TS0ID1
                                                    TIME   - 10:01

Update IADF with existing info from DBD or IMSADF II source

1 Static Rules - Add Segments, Fields, Transactions from Rules Source
                  Execution Type ==> B (F=Foreground, B=Batch)
2 Dynamic Rules - Add Messages, Help, Routes, Destinations, Users
3 DBD           - Add Segments and Fields from DBD Source

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT
```

Figure 10-9. Extraction Utilities Panel

From the Extraction Menu enter 2 on the option line to select “Dynamic Rules”.

```

ADFDYNT ----- DYNAMIC RULES EXTRACTION -----
COMMAND ==>

SYSID ==> SAMP   PGROUP ==> PG   LEVEL:3

ISPF INPUT LIBRARY:
PROJECT ==>
GROUP   ==>
TYPE    ==>
MEMBER  ==>

OTHER PARTITIONED DATA SET:
DATA SET NAME ==> 'imsadf.jcllib(transin)'

COMPLETE AND PRESS ENTER TO CONTINUE OR END TO EXIT

```

Figure 10-10. Dynamic Rules Extraction Panel

Your screen displays the **Dynamic Rules Extraction** panel.

You must enter the name of the library or sequential data set where the rules source for your messages, help panels, profiles, userids etc. is located.

This data set is **not** the IMSADF II Message or the Signon Profile data base, but a file that contains the transactions to create those segments in the above mentioned data base. This data set is normally used as input to the Batch Driver in order to update the IMSADF II data bases.

This particular example shows the dynamic rules extraction from IMSADF.JCLLIB(TRANSIN). You are not required to perform extract from TRANSIN, as it is shown for illustration purposes only.

The next panel shown is Figure 10-11.

```

ADFDYN1 ----- DYNAMIC RULES EXTRACTION SUMMARY-----
COMMAND ==>

Number of Dynamic Rules Found in: IMSADF.JCLLIB(TRANSIN)

Messages:                143
Help Panels:             7
User Ids:                 2
Profiles:                2
Alternate Routes:        0
Secondary Destinations:  0
Additional Tables:       0      (Pgroups in Routes/LTERMS in Destinations)

Statistics:

Number of errors found:    7
Total number of records:  976
Audit records ignored:    91
Comment lines found:     53
Unknown record type:      6
Delete records ignored:   0

PRESS ENTER TO CONTINUE OR END TO EXIT OR ENTER ERR TO BROWSE ERRORS

```

Figure 10-11. Dynamic Rules Extraction Summary Panel

The process ignores audit operations, delete statements, and marks out of sequence records. A summary report is presented, containing the number and type of elements extracted (i.e. messages, end-user HELP panels, IMSADF II userids, IMSADF II profiles, etc.). The number of records ignored and the number of records in error are also listed in the summary report, as shown in this sample.

The first part of the dynamic rules extraction process scans the specified input file and displays the summary panel.

```

ADFDYN1 ----- DYNAMIC RULES EXTRACTION SUMMARY-----
COMMAND ==> err

Number of Dynamic Rules Found in: IMSADF.JCLLIB(TRANSIN)

Messages:                143
Help Panels:              7
User Ids:                 2
Profiles:                 2
Alternate Routes:         0
Secondary Destinations:   0
Additional Tables:        0      (Pgroups in Routes/LTERMS in Destinations)

Statistics:

Number of errors found:   7
Total number of records: 976
Audit records ignored:   91
Comment lines found:     53
Unknown record type:     6
Delete records ignored:   0

PRESS ENTER TO CONTINUE OR END TO EXIT OR ENTER ERR TO BROWSE ERRORS

```

Figure 10-12. Using the ERR command to display the Extract Error Report

An error report is also produced and can be selected from this panel by the use of the **ERR** command. It is shown in browse mode in Figure 10-13.

```

ISRBROBF - TS0ID1.SPFTEMP1.CNTL ----- LINE 000000 COL 001 080
COMMAND ==>                                SCROLL ==> CSR
***** TOP OF DATA *****

IMSADF II                                DYNAMIC RULES EXTRACTION

DATE: 85/06/06 TIME: 13:43 ERROR LOG      USER: TS0ID1
06/06/85 13:43:15
Unknown Record type, or record out of sequence: MFC1B4HTMFC1@@@10001
06/06/85 13:43:15
Unknown Record type, or record out of sequence:
06/06/85 13:43:15
Unknown Record type, or record out of sequence: HELP INFORMATION IS NOT PROVID
06/06/85 13:43:15
Unknown Record type, or record out of sequence: MFC1B4HTMFC1@@@20001
06/06/85 13:43:16
Unknown Record type, or record out of sequence:
06/06/85 13:43:16
Unknown Record type, or record out of sequence: HELP INFORMATION IS NOT PROVID
06/06/85 13:43:17
Number of records for message 0072 is less than expected
***** BOTTOM OF DATA *****

```

Figure 10-13. Browsing the Dynamic Rules Extraction Error Report

After you browse the error report, press the **END** key to return to the Dynamic Rules Extraction Summary panel (originally shown in “Dynamic Rules Extraction”),

You may then

- View a selection list of elements to be extracted by pressing the **ENTER** key
- **OR**
- Cancel the dynamic rules extraction by pressing the **END** key

If you press the **ENTER** key, you see Figure 10-14.

```
MSGINL ----- MESSAGES IN IMSADF.JCLLIB(TRANSIN) -----
COMMAND ==> SCROLL ==> PAGE

Replace if duplicate ==> N (Y/N)
Line Command: Snn select (message ID can be changed if required)
C MSGID Length First two Lines of Text (65 characters only)
- -----
''' 0000 0070 PROCESSING LOGIC ERROR FOR TRXID=---. DL/I STATUS CODE=---.

s ' 0001 0140 AUDIT ERROR OCCURRED. MESSAGE NUMBER MISSING FROM OPERATION DESCR
    AUDITED FIELD IS . SEQ# = .
''' 0029 0140 *****ALL MESSAGES ON THIS SCREEN ARE CLASSIFIED AS WARNING. EN
    'U' TO PROCEED WITH UPDATE.
''' 0070 0140 THIS TRANSACTION REQUIRES THAT FIELD ( ) CONTAIN A VALUE 0
    THAN BLANK OR ZERO.
''' 0071 0070 NON-ALPHA DATA IN ALPHA FIELD. FIELD NAME IS .
''' 0072 0140 NON-NUMERIC DATA IN NUMERIC FIELD. FIELD NAME IS .
''' 0073 0140 RELATED FIELD REQUIRED FOR AUDIT CAN NOT BE LOCATED.
    AUDITED FIELD IS . SEQ# = .
''' 0079 0140 CHARACTER VALUE FOR CONVERSION TO BIT IS NOT 0 OR 1.
    AUDITED FIELD IS . SEQ# = .
```

Figure 10-14. Selecting a Message to be Extracted

A list is presented for each category of elements to be extracted: messages, **HELP** panels, **userid**s, and **profiles**. This example shows a selection list of messages.

In Figure 10-14, the message with **MSGID** #0001 has been selected for extraction since an 'S' was typed in the line command column for that message. If you wished to change the message number to be inserted in the **ISPF** table used by **IADF**, you could simply type over the **MSGID** field.

You can select more than one message by typing **snn** where **nn** is the number of messages to be selected.

When you press the **ENTER** key, the messages are processed and you see Figure 10-15.

```

MSGINL ----- MESSAGES IN IMSADF.JCLLIB(TRANSIN) -----
COMMAND ==> SCROLL ==> PAGE

Replace if duplicate ==> N (Y/N)
Line Command: Snn select (message ID can be changed if required)
C MSGID Length First two Lines of Text (65 characters only)
-----
''' 0000 0070 PROCESSING LOGIC ERROR FOR TRXID=---. DL/I STATUS CODE=--.

''' 0001 0140 AUDIT ERROR OCCURRED. MESSAGE NUMBER MISSING FROM OPERATION DESCR
ADDED
        AUDITED FIELD IS      . SEQ# =      .
''' 0029 0140 *****ALL MESSAGES ON THIS SCREEN ARE CLASSIFIED AS WARNING. EN
        'U' TO PROCEED WITH UPDATE.
''' 0070 0140 THIS TRANSACTION REQUIRES THAT FIELD (      ) CONTAIN A VALUE 0
        THAN BLANK OR ZERO.
''' 0071 0070 NON-ALPHA DATA IN ALPHA FIELD. FIELD NAME IS      .

''' 0072 0140 NON-NUMERIC DATA IN NUMERIC FIELD. FIELD NAME IS      .

''' 0073 0140 RELATED FIELD REQUIRED FOR AUDIT CAN NOT BE LOCATED.
        AUDITED FIELD IS      . SEQ# =      .
''' 0079 0140 CHARACTER VALUE FOR CONVERSION TO BIT IS NOT 0 OR 1.
        AUDITED FIELD IS      . SEQ# =      .

```

Figure 10-15. Confirmation message for Dynamic Rules Extract

For each selected message, a confirmation appears, indicating if the extract succeeded (ADDED) or failed (REJECTED). The panel shows the ADDED confirmation for MSGID #0001.

Press the END key to continue to the next category of elements to be extracted.

When the last category has been processed, return is made to the Extraction Utilities menu in Figure 10-16.


```
ADFUEX ----- EXTRACTION UTILITIES ----- Function Completed
OPTION ==>

SYSID ==> SAMP   PGROUP ==> PG   LEVEL: 3           USERID   - TS01D1
                                           TIME      - 12:00

1 Static Rules - Create Segments, Fields, Transactions from Rules Source
2 Dynamic Rules - Create Messages, Help, Routes, Destinations, Users
3 DBD          - Create Segments and Fields from DBD Source

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT
```

Figure 10-16. Successful Dynamic Rules Extraction

A confirmation message is displayed and dynamic rules extraction is complete.

Static Rules Extraction

The static rules extraction process is discussed in this topic. For dynamic rule information, please see “Dynamic Rules Extraction” on page 10-9.

```
ADFUEX ----- IADF EXTRACT MENU ----- Function Completed
OPTION ==> 1

SYSID ==> SAMP   PGROUP ==> PG   LEVEL: 3           USERID - TS01D1
                                           TIME    - 16:31

Update IADF with existing info from DBD or IMSADF II source

1 Static Rules - Add Segments, Fields, Transactions from Rules Source
                  Execution Type ==> B (F=Foreground, B=Batch)
2 Dynamic Rules - Add Messages, Help, Routes, Destinations, Users
3 DBD           - Add Segments and Fields from DBD Source

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT
```

Figure 10-17. Successful Extraction

From the Extraction Menu enter 1 on the option line to select “Static Rules”. You can invoke Extract in either foreground or batch; “f” beside “Execution Type” in this example selects foreground.

```

ADFUEXTP ----- STATIC RULES EXTRACTION -----
COMMAND ==>

SYSID ==> samp

Available Commands: CAN Cancel RES Restart

HAS RULE INFORMATION PREVIOUSLY BEEN EXTRACTED FROM SOURCE? ==> N (Y|N)
(N= data set below contains rules generator source)
(Y= data set below is rules generator output for ISPF tables)

ISPF LIBRARY:
PROJECT ==>
GROUP ==>
TYPE ==>
MEMBER ==>

OTHER PARTITIONED OR SEQUENTIAL DATA SET:
DATA SET NAME ==> 'imsadf.rules.source(rglsamp)'

Press ENTER to execute or END to exit

```

Figure 10-18. Load ISPF Tables from Rules Generator Input

Enter the information requested. The example shown in Figure 10-18 extracts the ISPF tables for the distributed sample application, SAMP, from its installation library, in this case IMSADF.RULES.SOURCE. Since the library is in the form of three nodes, you could have entered it in the ISPF format shown first.

The question, "HAS RULE INFORMATION ALREADY BEEN EXTRACTED FROM SOURCE?", refers to the fact that it is possible to execute the Rules Generator in batch, extracting data on permanent data sets. This example assumes that you have not done so; therefore, "N" is the correct response and the Rules Generator will be invoked in foreground by the Extract function.

You already have set up this SYSID earlier in the guide. In doing so, you named data sets to be used for SAMP in Figure 5-4.

Press ENTER.

If you had entered "Y" in response to "HAS RULE INFORMATION ALREADY BEEN EXTRACTED FROM SOURCE?", you would have entered the data set from your batch Rules Generator run (ISPF DDNAME) in place of the Rules Generator input data set. This indicates to Extract that you have run the Rules Generator in batch. When you invoke Extract, the table load function is used to create the SYSID tables, saving you the time necessary to invoke the Rules Generator in foreground.

Note: When you ran the Rules Generator using batch tables, the output rules were placed in the rules load library named in Figure 5-4.

ADFRGF1 ----- STATIC RULES GENERATION FOREGROUND -----	
SYSID: SAMP	
Signon/Pomenu:	(Y N Signon/Pomenu included)
SOM Update:	(Y N Update Secondary Option Menu)
CSOM Indicator:	(Y N Set Composite Indicator in SOM)
LIBRARY INFORMATION	
Rules Source:	TSOID1.TEST.RULSRC
Rules Load:	TSOID1.TEST.RULLIB
Image Source:	TSOID1.TEST.ADFIMG
Exits Object:	
 12:00 Now the work data sets for Rule generation are being deleted...	

Figure 10-19. Static Rules Generation in Foreground

You are shown Figure 10-19. This is the panel you see when you invoke the Rules Generator in foreground mode. The libraries you originally allocated in Figure 5-4 appear under **LIBRARY INFORMATION**. The keyboard is locked during this time.

At the bottom of the panel is a time stamp which changes as the rules generation process continues. Below that is an informational message reporting on the Rules Generator process. This panel is shown as many as three times during rule generation, with the time stamp and the message at the bottom changing in this order:

1. Now the work data sets for Rule generation are being deleted.
2. Now the data sets for Rule generation are being allocated.
3. Now the Rule Generator Program is being called.

At the completion of the foreground invocation of the Rules Generator, the following appears in browse mode:

1. TSOID1.ADF.GENLIST--Rules Generator output
2. TSOID1.ASMLIST--Assembler source from Rules Generator (if you specified on ASMR = YES on Rules Generator input or your profile)
3. TSOID1.ADF.LIST--Linkage Editor output from Rules Generator.

The data set names are those from the RULEGEN table, created in "Establish Foreground Static data sets for the SYSID" on page 7-7.

Press END.

```

ADFUEX ----- IADF EXTRACT MENU ----- Function Completed
OPTION ==>

SYSID ==> SAMP  PGROUP ==> PG  LEVEL: 3          USERID  - TS0ID1
                                           TIME    - 16:43

Update IADF with existing info from DBD or IMSADF II  source

  1 Static Rules - Add Segments, Fields, Transactions from Rules Source
                    Execution Type ==> B (F=Foreground, B=Batch)

  2 Dynamic Rules - Add Messages, Help, Routes, Destinations, Users

  3 DBD           - Add Segments and Fields from DBD Source

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 10-20. Successful Static Rules Extraction

A confirmation message is displayed. Press the RETURN PF key to go back to the IADF Main Menu.

Note: The table creation is not performed unless the Rules Generator has completed normally. Additionally Extract can be run only once successfully for a SYSID. Subsequent invocations do not replace the tables and will fail for duplicate transactions or duplicate segment names which have different segment types.

Additional Tasks to Complete

```

IADFMAIN ----- IADF MAIN MENU -----
OPTION ==> 1

KEY ==>                                SYSID ==> SAMP  LEVEL: 3

  1  DEFINITION      - Create, modify or browse application information
                        (Data Layouts, Panels, Transactions, Audits, Messages,
                        Userids and Profiles, Transaction Drivers, SYSIDs)

  2  GENERATION      - Perform application generation

  3  TEST            - Prepare for and perform test (PROTOTYPE, BTS)

  4  UTILITIES       - Perform IADF Utilities
                        (Extract, Submit, Glossary, Documentation, SYSID Info,
                        Trace, GFIND, MFS Utility, Data Dictionary, DB2 RGLGEN)

  5  IADF PROFILE    - Create or modify IADF User Profile

  6  TUTORIAL        - Display information about IADF

  X  EXIT            - Terminate IADF DEVELOPMENT

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 10-21. IADF Main Menu

Enter 1 on the option line of the IADF Main Menu to select the IADF Definition Menu.

```

IADFDEF ----- IADF DEFINITION MENU -----
OPTION ==>

KEY ==>                                SYSID ==> SAMP  LEVEL:3

  1  Data Layouts      SEGMENT TYPE ==>      Valid Segment Types:
  2  Panels
  3  Transactions
  4  Audits
  5  Messages
  6  IMSADF II  Userids
  7  IMSADF II  Profiles
  8  Transaction Drivers
  9  Applications (SYSIDs)
 10  Routes
 11  Secondary Destinations

DBS = Data Base Segment
TBL = DB2 Table
PS  = Pseudo Segment
MAP = Mapping Segment
OUT = Out Segment
DBEK = Fastpath DB Seg
KSDS = Key Sequenced Data
ESDS = Entry Sequenced Data
RRDS = Relative Record I/Key
RREK = Relative Record E/Key

ENTER OPTION NUMBER AND ANY REQUIRED DATA OR PRESS END KEY TO EXIT

```

Figure 10-22. IADF Definition Menu

Viewing Figure 10-22, you see all the functions that the Extract service performed.

The Extract function does not perform the following:

- Screen Image source conversion
- MFS generation

Screen Image Conversion

If you used screen image format for any of your transactions or for signon, the Rules Generator created the proper MFS source, but the image input is not valid for processing in IADF. Two conditions are discussed:

External screen image library members: The members are referenced in the Rules Generator input. They reside in the screen image source library allocated to this SYSID. The names by which they were known prior to extraction are no longer valid. IADF requires that you rename them following this convention:

```
ssssSItx      where
                ssss = SYSID
                SI is constant
                tx = transaction identifier or
                  constant GN for signon
```

Imbedded screen images: When screen images are imbedded in the Rules Generator input, they are unable to be converted by Extract. You must create members for all requested screen image input in the extracted application. This example shows how to create signon screen image.

Enter 2.2 on Figure 10-22 to select panel definition for Signon/Primary Option Menu.

```
ADFSIG1 ----- SIGNON/PRIMARY OPTION MENU -----
COMMAND ==> pan

Available Commands: CAN Cancel  GEN Generate  PAN Panel Image  SRC Source

SYSID: SAMP

SIGNON      ==> y              (Signon request: Y=yes, N=no, R=RACF)
SHEADING    ==>
SOIMAGE      ==> y              (Y|N screen image used)
DEVNAME      ==> 2,A03,A04
DEVTYPE      ==> 2,7,4
POMENU       ==>              (Primary Option Menu Options)
LINKLIB      ==>              (DDname for Link Edit process)

Description  ==>
COMPLETE AND PRESS END TO PROCESS AND EXIT
```

Figure 10-23. Signon/Primary Option Menu definition

Enter the request for SIGNON and for screen image (SOIMAGE).

Key the request for panel image (PAN) on the Command line and press ENTER.

```
EDIT --- TS0ID1.TEST.IMGSRG(SAMPSIGN) - 01.00 ----- COLUMNS 001 072
COMMAND ===>                                     SCROLL ===> CSR
SYSID: SAMP  PGROUP: PG  LEVEL: 3
Screen type: S      (N=screen image,S=sign-on)
***** ***** TOP OF DATA *****
000001                                     'HEADING.....
000002
000003
000004
000005
000006
000007                                     SYSID-----:&SYSID
000008                                     USERID-----:&USERID
000009                                     PROJECT-----:&PROJECT
000010                                     GROUP-----:&GROUP
000011                                     LOCKWORD--:&LOCKWORD
000012
000013
000014
000015
000016
000017
000018
000019
000020  OPTION :&OPTION TRANSACTION :&TRAN
000021  KEY      :&KEY
000022
000023
000024 &SYSMSG
=NOTE=          REMOVE OPTION AND KEY ROWS IF YOU DON'T WANT TO SELECT
=NOTE=          TRANSACTIONS FROM THE SIGNON PANEL, ALSO SYSID MAY BE REMOVED
=NOTE=
```

Figure 10-24. Model Screen Image Signon Panel in Edit Mode

Figure 10-24 is an IADF "model" for a screen image signon panel. It is displayed in ISPF EDIT mode. The name of the member in your screen image library is SAMPSIGN, created by IADF. Since you are in edit mode:

1. Use the COPY command to copy the Rules Generator input, IMSADF.RULES.SOURCE(RGLSAMP).
2. Delete all lines of the input and model except for the screen image source.
3. SAVE the member.

Follow this procedure for screens for all the transactions extracted.

Enter 3 on the option line for Figure 10-22 to select TRANSACTIONS. From that point on the resulting panel, Figure 10-28, you may select the Panel(P) option for each transaction to accomplish the same thing as you did for signon.

MFS Generation

The Rules Generator invocation during Extract processing created MFS source for any screen requested in the input, including screen image. It was placed in the library named as the SCREENS DD statement in “Establish Foreground Static data sets for the SYSID” on page 7-7. This data set is created for the duration of your TSO session. If you wish to generate MFS blocks, you must do so at this time. You are not required to do this as a result of Extract processing.

```
IADFMAIN ----- IADF MAIN MENU -----
OPTION ==> 4.8

KEY ==>                                SYSID ==> SAMP   LEVEL: 3

  1  DEFINITION      - Create, modify or browse application information
                        (Data Layouts, Panels, Transactions, Audits, Messages,
                        Userids and Profiles, Transaction Drivers, SYSIDs)

  2  GENERATION      - Perform application generation

  3  TEST            - Prepare for and perform test (PROTOTYPE, BTS)

  4  UTILITIES       - Perform IADF Utilities
                        (Extract, Submit, Glossary, Documentation, SYSID Info,
                        Trace, GFIND, MFS Utility, Data Dictionary, DB2 RGLGEN)

  5  IADF PROFILE    - Create or modify IADF User Profile

  6  TUTORIAL        - Display information about IADF

  X  EXIT            - Terminate IADF DEVELOPMENT

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT
```

Figure 10-25. IADF Main Menu

Enter 4.8 on the option line of the IADF Main Menu to skip to the UTILITIES Menu. You may use the MFS utility panels to generate your formats.

Transaction Control Tables

The transaction control table is automatically generated for the extracted transactions. This table is used by various IADF functions that are aids for building screen images and audit rules. It is also used to store Rules Generator override values for the fields and segments in the transaction. The override values stored in the table for each segment and field may be blank.

```

IADFMAIN ----- IADF MAIN MENU -----
OPTION ==> 1

KEY ==>                                SYSID ==> SAMP  LEVEL: 3

  1 DEFINITION      - Create, modify or browse application information
                    (Data Layouts, Panels, Transactions, Audits, Messages,
                    Userids and Profiles, Transaction Drivers, SYSIDs)

  2 GENERATION      - Perform application generation

  3 TEST            - Prepare for and perform test (PROTOTYPE, BTS)

  4 UTILITIES       - Perform IADF Utilities
                    (Extract, Submit, Glossary, Documentation, SYSID Info,
                    Trace, GFIND, MFS Utility, Data Dictionary, DB2 RGLGEN)

  5 IADF PROFILE    - Create or modify IADF User Profile

  6 TUTORIAL        - Display information about IADF

  X EXIT           - Terminate IADF DEVELOPMENT

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 10-26. IADF Main Menu

Enter 1 on the option line of the IADF Main Menu to select the IADF Definition menu.

```

IADFDEF ----- IADF DEFINITION MENU -----
OPTION ==> 3

KEY ==>                                SYSID ==> SAMP  LEVEL:3

  1 Data Layouts      SEGMENT TYPE ==>      Valid Segment Types:
  2 Panels
  3 Transactions      DBS  = Data Base Segment
  4 Audits             TBL  = DB2 Table
  5 Messages           PS   = Pseudo Segment
  6 IMSADF II Userids  MAP  = Mapping Segment
  7 IMSADF II Profiles OUT  = Out Segment
  8 Transaction Drivers DBEK = Fastpath DB Seg
  9 Applications (SYSIDs) KSDS = Key Sequenced Data
 10 Routes            ESDS = Entry Sequenced Data
 11 Secondary Destinations RRDS = Relative Record I/Key
                       RREK = Relative Record E/Key

ENTER OPTION NUMBER AND ANY REQUIRED DATA OR PRESS END KEY TO EXIT

```

Figure 10-27. IADF Definition Menu

Enter 3 on the option line to select TRANSACTION.

```

TRXPNL ----- TRANSACTIONS IN SAMP -----
COMMAND ==>                                SCROLL ==> PAGE

Confirm before Deletion ==> Y (Y/N)
Line Commands: B Browse D Delete G Generate H HLAL I Insert P
Panel

          S Source T Control Table (Overrides)      U Update X STX
C TRXID F Transaction Name      Type Special STX AGROUP P DBPATH
-----
' CD    CLOSE/DISBURSE INVENTORY C YES N YYYY A IV
' CY    CYCLE COUNT              C NO  N YYYY A CY
' IV    INVENTORY                C NO  N YYYY A IV
' PA    PART SEGMENT             C NO  N YYYY A PA
t PD    STANDARD INFORMATION      C NO  N YYYY A PD
***** BOTTOM OF DATA *****

```

Figure 10-28. Transactions for a SYSID

Enter "t" to update a transaction control table for a transaction (Transaction PD is shown in this example). Press ENTER.

```

IADFCTBL ----- TRANSACTION PD      CONTROL TABLE -----
OPTION ==> 3

1 CREATE      - Create a transaction control table from scratch.
               If a previous table is found, it will be
               replaced. Use this option only if you do not
               want to preserve the previous table.

2 RECREATE    - Update the control table when new segment or field
               information needs to be added for this transaction.
               Override information is preserved.

3 UPDATE      - Update values for segment and fields in the control
               table. No new fields or segments will be inserted.
               Use this option for faster response when no new
               information was added.

4 DELETE      - Delete inactive fields in the control table. These
               fields contain no override information. Use this
               option to decrease transaction generation time. If
               you want to restore fields later, use RECREATE.

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 10-29. Transactions for a SYSID


```

TRXOVT1  -- CONTROL INFO FOR SEGMENTS IN TRANSACTION PD ---
COMMAND ===> SCROLL ===> PAGE

Line Commands: Dnn Delete  Unn Update

CMD  ID.SS  KSEL  ASTATUS  COFIELD  KEYBLD  SLEN  DM  SNAME
'''  KEY.PA  K           0017  Y  PART NUMBER
'''  DESC.PA  R           0020  Y  DESCRIPTION
'''  KEY.PD  K           0002  Y  KEY FIELD
'''  PRCD.PD  N           0002  Y  PROC CODE
'''  INVC.PD  N           0001  Y  INVENTORY CODE
'''  PLRV.PD  N           0002  Y  PLAN REV NO
'''  MKDP.PD  N           0004  Y  MAKE DEPT
***** BOTTOM OF DATA *****

```

Figure 10-31. Control Information for Fields in Transaction

In Figure 10-31 the original values for each field are shown in display mode only. Use the "update" line command under the CMD column to enter any values you wish to override for the field, if the override will be required for it in the transaction.

In Figure 10-31, the "update" command is not used on the panel, since no field overrides are required for the transaction.

Press END.

```
TRXPNL ----- TRANSACTIONS IN SAMP ---- Control Table Updated
COMMAND ==>
Control table for PD has been created successfully
Confirm before Deletion ==> Y (Y/N)
Line Commands: B Browse D Delete G Generate H HLAL I Insert P
Panel
          S Source T Control Table (Overrides)      U Update X STX
C TRXID F Transaction Name      Type Special STX AGROUP P DBPATH
-----
' CD    CLOSE/DISBURSE INVENTORY C YES N YYYY A IV
' CY    CYCLE COUNT              C NO  N YYYY A CY
' IV    INVENTORY                C NO  N YYYY A IV
' PA    PART SEGMENT             C NO  N YYYY A PA
' PD    STANDARD INFORMATION      C NO  N YYYY A PD
***** BOTTOM OF DATA *****
```

Figure 10-32. Transactions for a SYSID

Figure 10-32 shows that a transaction control table has been created for the selected transaction.

Repeat the process for all transactions in the extracted SYSID.

Glossary Services

By adding terms used at your location, you can speed the education process of your developers.

The ability to add your terms to the supplied glossary for ISPF/IADF is available to Level 3 Administrators only.

```

IADFADMN ----- IADF ADMINISTRATION MAIN MENU -----
OPTION ==> 5

ADMINISTRATIVE LEVEL:3
KEY ==>                                SYSID ==> SAMP    LEVEL: 3

  1 SYSTEMS      - Create or modify IMSADF II  systems (SYSID)
  2 MIGRATION    - Perform migration    TYPE ==> C  ( M Move  C Copy )
  3 ADMINISTRATORS - Specify administrative users
  4 TABLES      - Create, modify, and/or copy ISPF tables
  5 GLOSSARY     - Update glossary terms
  6 DEVELOPMENT  - Perform IMSADF II  development tasks
  X EXIT        - Terminate IADF

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 10-33. Select GLOSSARY

Enter 5 on the option line to select the Glossary activities.

```

ADFGLOSM ----- IMSADF II GLOSSARY MENU -----
COMMAND ==> a

D = Display Glossary Definitions
A = Add a new TERM to the Glossary
M = Modify an existing TERM in the Glossary
R = Renumber an existing TERM in the Glossary

TERM ==> testterm

Press END key to exit

```

Figure 10-34. Glossary Menu

Enter "A" on the command line to add the term named beside "TERM".

```

ADFGLOSU ----- IMSADF II GLOSSARY UPDATE -----
COMMAND ===>                                SCROLL ===> PAGE

Press END key to exit with update
Enter CANCEL to exit without update

TERM : TESTTERM

Line Commands: Dnn Delete  Inn Insert

i2'

***** BOTTOM OF DATA *****

```

Figure 10-35. Adding a Term to the Glossary

Enter "i2" to insert two lines (or as many as you think you will use for a definition).

```

ADFGLOSU ----- IMSADF II GLOSSARY UPDATE -----
COMMAND ===>                                SCROLL ===> PAGE

Press END key to exit with update
Enter CANCEL to exit without update

TERM : TESTTERM

Line Commands: Dnn Delete  Inn Insert

... This is a test term. This term is used to verify the
... glossary addition function of IADF

***** BOTTOM OF DATA *****

```

Figure 10-36. Adding a Term to the Glossary

Enter the term definition and press END PF key to add the definition.

This function may also be used to modify existing terms.

Table Management

This function allows you to manipulate the IADF/ISPF tables to create, modify, and copy tables.

You may use the table copy to create backup for tables which you expect to alter in some way. This function is also useful for propagation of SYSID tables (after one set has been created). This ability is described in “A Technique for Administering Multiple SYSIDS” on page 7-24.

The occasion calling for table modification in your installation should be rare. It should be used with care. You must know the table structure of the table you wish to modify and how it is used in IADF processes. In addition you should be experienced in the art of table manipulation using the ISPF Dialog Manager.

The ability to manipulate tables is available to

- Level 3 Administrators
- Level 2 Administrators

```
IADFADMN ----- IADF ADMINISTRATION MAIN MENU -----
OPTION ==> 4

ADMINISTRATIVE LEVEL:3
KEY ==>                               SYSID ==> SAMP   LEVEL: 3

  1  SYSTEMS      - Create or modify IMSADF II   systems (SYSID)

  2  MIGRATION    - Perform migration    TYPE ==> C  ( M Move  C Copy )

  3  ADMINISTRATORS - Specify administrative users

  4  TABLES      - Create, modify, and/or copy ISPF tables

  5  GLOSSARY     - Update glossary terms

  6  DEVELOPMENT  - Perform IMSADF II   development tasks

  X  EXIT         - Terminate IADF

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT
```

Figure 10-37. Select TABLES

Enter 4 on the option line to select table management.

```

ADFPR09 ----- TABLE MANAGEMENT -----
OPTION ==>

SYSID ==> SAMP   PGROUP ==> PG   LEVEL:3           USERID   - TS0ID1
                                                    TIME     - 12:00

1 CREATE      - Create ISPF Tables
2 MODIFY      - Add or Delete Columns to ISPF Tables
3 COPY        - Copy ISPF TABLES

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT

```

Figure 10-38. Table Management Menu

The following panels are shown to you to acquaint you with table services. Please refer to HELP panels for details of each function.

Table Create

```

ADFPR10 ----- TABLES: CREATE USING MASTER TABLE DEFINITIONS -----
COMMAND ==>

Available Command: CAN Cancel

TABLE Name ==>

DDNAME      ==> SAMP

DSNAME      ==>

Master Key  ==>

COMPLETE AND PRESS ENTER TO CONTINUE OR END TO EXIT

```

Figure 10-39. Table Create

Table Modify

```
ADFTBCP ----- TABLES: ADD KEYS/COLUMNS, DELETE KEYS/COLUMNS -----
COMMAND ===>
  Available Commands: CAN Cancel  DEL Delete table  REP Replace structure
TABLE Name ===>                                DDNAME: SAMP
                                           Table Structure:

Keys      ===>

Variables  ===>

No. of rows ===>

Additional names (option REP )

Keys      ===>
Variables ===>

COMPLETE AND PRESS ENTER TO CONTINUE OR END TO EXIT
```

Figure 10-40. Alter Columns in a Table

When you enter the "TABLE Name", the structure of the table is displayed to you. The DDNAME is that of the current SYSID table data set.

If the DDNAME is not allocated, you are shown panel ADFT001, which displays the allocation error. You may then use the TSO ALLOC command to allocate the data set to the named DD statement.

Table Copy

```
ADFPRI1 ----- TABLES: COPY FUNCTION -----  
COMMAND ==>  
  
    Available Commands: CAN Cancel  REP Replace  
  
FROM:  
  
TABLE Name ==> sampst  
  
DDNAME      ==> samp  
  
DSNAME      ==>  
  
TO:  
  
TABLE Name ==> bankst  
  
DDNAME      ==> banktab  
  
DSNAME      ==> 'tsoid1.bank.ispftbls'  
  
COMPLETE AND PRESS ENTER TO CONTINUE OR END TO EXIT
```

Figure 10-41. Table Copy Function

The DDNAME of your current SYSID Tables data set appears in both the "From" and "To" entries, but you may key over either. If the DDNAME is already allocated, you do not need to enter the data set name. When you use this panel, normally your "From" data set information is known, but you may need to enter the "To" data.

Enter the name of the table, the "From" and "To" data set names, if necessary, and a new table name if you wish, as shown in Figure 10-41. If the table already exists, you must enter "rep" on the command line if you wish to replace it.

Note: Table copy encompasses more than a data set copy. In addition to copying the member, table management changes the ISPF table identifier, which is contained within the table. If you copy from a table named BANKST to SAMPST, table management changes the identifier in the "to" table to SAMPST.

At the completion of the copy, you see the message TABLE UPDATED appear in the third line of the panel (long message).

Chapter 11. IADF Structure and Flow

This chapter lists some of the important Clists and programs used in IADF. You may find this chapter helpful in understanding the structure of IADF and in following the flow of panels, Clists, and programs. Each Clist named contains a brief description of the functions it performs.

The first six topics within this chapter are options on the IADF Main Menu. The last topic refers to administrative functions which are options on the IADF Administration Main Menu. The Clists and programs are described within these functional groupings:

1. "OPTION 1: Definition" on page 11-2.
2. "OPTION 2: Rules Generation" on page 11-5.
3. "OPTION 3: Test" on page 11-6.
4. "OPTION 4: IADF Utilities" on page 11-8.
5. "OPTION 5: IADF User Profile" on page 11-10.
6. "OPTION 6: Tutorial Function" on page 11-10.
7. "IADF Administration Functions" on page 11-11.

IADF is structured into several logical functions that cross-reference each other to eliminate excessive menu navigation and provide a consistent environment.

Most of the Clists are split into these categories:

1. Dialog Clist

Each of these Clists handles a specific dialog that normally consists of a menu display, an item display, and an entire update process.

2. Generation Clist

The Clists in this section perform rules generation. Temporary rules source is created from the information stored in the ISPF tables.

3. Service Clist

The Clists in this category are always called by another Clist and may or may not present dialog screens.

IADF also contains several programs classified as virtual table generators, which are found in IMSADF.ADFLOAD. Virtual tables are created by IADF in order to reduce MVS enqueueing caused when multiple users access the same set of tables. All the programs scan the table libraries and construct a virtual table in storage based on parameters passed by the calling Clist. One row in a virtual table is built for each matching element found. A virtual table exists only for the duration of the function.

The category is found in parentheses at the end of the description of each Clist or program.

OPTION 1: Definition

Where a specific table is created or updated, its abbreviated name is used in this chapter. To determine the full name, you may refer to Appendix B, "IADF Naming Conventions."

This chapter contains eleven different options.

Option 1.1: Data Layouts

- ADFSEG** Presents a selection list of segments, honors the user's commands, and passes control to the rest of the Clists in the chain. It creates or updates segment related tables. (Dialog Clist)
- ADFSEG2** Generates segment macro source and stores it. (Generation Clist)
- MFC1EX1** A table generator which creates virtual tables in storage based on the input type. (Program)

Option 1.2: Panels

The table affected in this section is "SYS".

- ADFSIPM** Processes the dialog keywords for Signon and Primary Option Menu. It invokes ISPF edit if a signon-tailored screen is defined. (Dialog Clist)
- ADFSIPM1** Generates source for Signon and Primary Option Menu. (Generation Clist)

The table affected in this section is "HLP".

- ADFHLP** Handles the IMSADF II (not IADF) help panel generation and control. It presents a selection list, then updates, creates, or deletes any particular panel on request. On request it calls the dynamic execution Clist, ADFAG, to update the message data base. (Dialog Clist)

Note: The user must provide the end of text terminator DELIM two times: \$\$\$\$\$. DELIM is normally '\$\$' but could be different at individual locations since it is an installation option of IMSADF II.

- ADFHLP2** Converts the panel from ISPF table format to IMSADF II batch transactions and stores the result in a data set. (Generation Clist)

Option 1.3: Transactions

- ADFTRX** The main transaction handler. It presents menu selections, updates an individual transaction, and presents a list of all the transaction extensions. (Dialog Clist)
- ADFTRXN1** Creates the source for the transaction and stores it. (Generation Clist)
- MFC1EXD** Creates the source for the transaction. (Program)
- ADFTRX1** Controls the secondary transactions feature of IADF, presents a table display of secondary transactions, and communicates with ISPF editor to define printer images if required. (Dialog Clist)

ADFTRX2A The transaction control table handler. The transaction control concept was introduced to eliminate the need for an individual user to know which segments are needed for a particular transaction and which fields are valid for reference. The process takes DBPATH, TSEGS, TWINS, etc., keywords and creates a unique list of segment IDs that is later used to search for each of those segment, extracting relevant information from them and its fields. That information is stored in the transaction control table and is referenced anytime the transaction is generated, when the screen image is constructed, or for dynamic rules. (Service Clist)

Option 1.4: Audits

The tables affected in this section are "SB", "SG", and "DG".

ADFD001A Processes the high level audit options: transaction related, segment related and subroutine audits. It presents a selection menu and updates the header for the audit. It calls either ISPF edit to handle the source, or the audit rules executor (batch or foreground) to compile the audit source. (Dialog Clist)

MFC1EX1 A table generator described in "Option 1.1: Data Layouts" on page 11-2. (Program)

Option 1.5: Messages

The table affected in this section is "MSG".

ADFMSG Handles the message generation and control. It presents a selection menu (if no key is supplied) of all the messages within a SYSTEM ID, then updates, inserts, and deletes messages. On request, it communicates with the dynamic execution Clist, ADFAG, to update the message data base. (Dialog Clist)

ADFMSG2 Converts the message from symbolic to IMSADF II batch transaction form and stores the result in a data set (temporary or permanent). (Generation Clist)

Option 1.6: IMSADF II Users

The table affected in this section is "USR".

ADFUSR Displays the list of users (placed in the SR segments of the Signon Profile Data Base) when IMSADF II is invoked) and updates any user that is changed from that selection. It may call the dynamic execution Clist, ADFAG, to update the Signon Profile Data Base. (Dialog Clist)

ADFUSR2 Converts the user IDs from ISPF table format to IMSADF II batch transactions and stores the result in a data set. (Generation Clist)

Option 1.7: IMSADF II Profiles

The table affected in this section is "PFL".

ADFPRO Displays the list of profiles (found in the PR segments of the Signon Profile Data Base when IMSADF II is invoked) and updates any which are changed from that selection. It may call the dynamic execution Clist, ADFAG, to update the Signon Profile data base. (Dialog Clist)

ADFPROA Converts the profiles from ISPF table format to IMSADF II batch transactions and stores the result in a data set. (Generation Clist)

Option 1.8: Transaction Drivers

The table affected in this section is "SG".

ADFDRVR Is the driver handler, displaying a selection menu and updating a specific driver. (Dialog Clist)

ADFDRVR1 Generation Clist for link-edit rules. (Generation Clist)

Option 1.9: Applications (SYSIDs)

The table affected in this section is "SYSTEMS".

ADFSYST Presents all the keywords for a SYSTEM statement in two panels and updates the table if changes were made. (Dialog Clist)

ADFSYST1 Generates the source for a SYSTEM statement and stores it. (Generation Clist)

Option 1.10: Routes

The table affected in this section is "ROU".

ADFROUT Presents a selection list of all the ROUTE IDs within a system. The user must select the proper format of the key, depending on the routing type: transaction, global by SYSID, by project group etc. Additionally, options may be selected to add additional projects and users to a given route. This Clist calls the dynamic execution Clist, ADFAG, (on request) to add those routes into the data base. (Dialog Clist)

ADFROU2 Converts the routes from ISPF table format to IMSADF II batch transactions and stores the result in a data set. (Generation Clist)

Option 1.11: Secondary Destinations

The table affected in this section is "SDT".

ADFSDT Handles the selection of secondary destinations and specifications for additional input terminals plus the general maintenance of those destinations. It may call the dynamic execution Clist, ADFAG, to update the data base. (Dialog Clist)

ADFSDT2 Converts the destinations from ISPF table format to IMSADF II batch transactions and stores the result in a data set. (Generation Clist)

OPTION 2: Rules Generation

This section contains three different options.

Option 2.1: Static Rules Generation

ADFRG Controls static rules generation. It presents dialog panels for either batch or foreground environment. Both panels contain the same input fields except for the bottom part. The Clist processes requests for rules generation, which are handled by using calls to the appropriate generation Clist (see "OPTION 1: Definition" on page 11-2).

The Clist contains additional logic in transactions to verify the segments which have already been generated, to avoid possible duplications if the same segment is also included in a transaction. It additionally handles the foreground process of calling the Rules Generator.

This Clist can also be called from the definition option selected from the IADF Main Menu, passing as parameter the segment, transaction, driver, etc. to be generated. (Service Clist)

Option 2.2: Dynamic Rules Generation

ADFAG Controls dynamic rules generation. It presents dialog panels for either batch or foreground environment. Both panels contain the same input fields except for the bottom part. The Clist processes requests for dynamic generation, which are handled by using calls to the appropriate generation Clist (see "OPTION 1: Definition" on page 11-2).

The Clist contains additional logic in transactions to separate requests for high and low level dynamic rules. It additionally handles the foreground process of calling the High Level Audit Language and IMSADF II batch driver.

This Clist can also be called from the definition option selected from the IADF Main Menu, passing as parameters the segment audits, messages, etc. to be generated. (Service Clist)

Option 2.3: Audit Rules Generation

ADFAUDM Handles the conversion from data base audit rules to static rules. This process can be executed in batch or foreground and contains logic to handle audit groups names of 4 or 8 characters. If the name contains four characters, the SYSID is prefixed before the request is passed. (Service Clist)

OPTION 3: Test

This section contains two major options: Prototyping and BTS Testing.

Option 3.1: PROTOTYPING

Option 3.1.1: Screen Image Creation

ADFPRO1 Displays a panel for the user to specify the data set where the screen image exists or where it is to be stored. Usually the data set name is known to IADF through the SYSTEMS table, and it is supplied in the data set name line. This Clist uses ISPF edit services and a macro to provide models for the different types of screen images. (Dialog Clist)

Option 3.1.2: Screen Image Translation

ADFPRO2 Examines the screen image and constructs the control used by the second translation Clist. It verifies the existence of the input data set and member and the existence of the output data set. After the translation is completed by the second Clist, it displays (if possible) the panel. (Dialog Clist)

Note: The panel can only be displayed if it is stored in a library that is concatenated in DDNAME ISPLIB. Otherwise, it is translated, but not displayed, and the user must move it to the required library if the prototyping function is used.

ADFPRO2B Accesses the information constructed by the ADFPRO2 Clist and completes the translation of the panel. (Service Clist)

Option 3.1.3: Data Base

ADFPRO8C Allows the developer to define and update simulated data bases for prototyping. (Dialog Clist)

Option 3.1.4: Transaction

ADFPRO9C Allows the developer to prepare a transaction for prototype test. This option automatically builds a prototype dialog for the developer.

Option 3.1.5: Dialog Creation

The table affected in this section is "DG".

ADFPRO4 Presents a selection dialog and provides maintenance functions for it. It performs validity checking on most operations and provides a default skeleton that the user may customize. (Dialog Clist)

MFC1EX1 A table generator described in "Option 1.1: Data Layouts" on page 11-2. (Program)

Option 3.1.6: Dialog Test

The table which is read in this section is "DG".

ADFPRO5 Dialog executor which interprets the instructions in the dialog. It provides support for the following:

- Arithmetic operations
- IF decisions
- GOTO capability
- Panel display

- HELP panel display
- Message generation
- Variable assignment
- Loop detection

Standard IMSADF II screens are provided for functions such as Signon, Primary Option Menu, Secondary Option Menu, and Primary Key Selection. (Dialog Clist)

MFC1EX1 A table generator described in “Option 1.1: Data Layouts” on page 11-2. (Program)

MFC1EX5 Constructs data base keys for prototyping. (Program)

Option 3.1.7: Prototype Control

ADFPROB Controls all prototyping resources and keeps a log.

Option 3.2: BTS

ADFBTS1 Handles all but the “Specify BTS User Data sets” option. It presents appropriate screens for the functions and calls the required service Clist to perform the function. (Service Clist)

ADFBTS Handles the creation and maintenance of BTS user data sets such as BTSIN and user data bases. (Service Clist)

OPTION 4: IADF Utilities

The Clists are described within the functional groups listed in this topic.

Option 4.1: Extract

This section is used to generate ISPF tables from IMSADF II static rules source and from IMS DBDGEN macro source. This function is performed by four Clists. “Extract ISPF Tables from Rules Source” on page 10-6 describes the panel flow from an external standpoint.

ADFSEGEN Converts from source DBD to data base segments and fields. It is necessary to provide the segment and field ID's to be used as identifiers in the ISPF tables. It creates the tables "DB" and "DF". (Dialog Clist)

ADFUEXT Controls the dialog for the conversion from existing rules source statements to ISPF tables in two modes.

In the first mode, it calls a service Clist ADFXT, followed by the Rules Generator Clist ADFRG to generate the extract file.

In the second mode, it assumes that the extract file already exists and calls MFC1TABL directly. (Dialog Clist)

ADFUEXTB Performs the same function in batch as ADFUEXT does in foreground. (Dialog Clist)

ADFXT Scans the source file for the extraction for two reasons: to verify that it contains IMSADF II valid source, and to detect if a request for extraction is present on the file. If that request is not found, it is inserted and later removed by a second call to this same Clist. (Service Clist)

MFC1TABL Creates from the extract file the ISPF tables for:

- segments
- fields
- transactions
- drivers
- secondary transactions
- Primary Option Menu.

The program ignores screen images. (Program)

Option 4.2: SUBMIT Utility

Submit process, used when a batch job is to be send to the internal reader.

ADFSUBM Sends the required file to three possible places: static rules generation, dynamic rules generation, direct to batch services. (Service Clist)

Option 4.3: Glossary Function

This function presents a glossary of IMSADF II terms that can be scrolled sequentially or by term.

MFC1GLOS May be called from either the IADF Main Menu or from anywhere in IADF by using the command GLOSSARY and the term that requires explanation. (Program)

Option 4.4: Rules Documentation

This section contains three options. Reports for static rules, for dynamic rules and cross references between segment-transactions.

ADFRP Processes the report options. It produces commands based upon the user's choices and submits those requests to the internal reader. (Dialog Clist)

ADFSEGCR Displays cross references between segments and transactions. Those references are constructed when the transaction control table for a particular transaction is requested. (Dialog Clist)

Option 4.5: Library Display

This option allows developers to browse the list of libraries used by a specified SYSID.

IADFDISP Displays the library list.

Option 4.6: IMSADF II Trace

ADFTRACC Processes the options for IMSADF II macro, ADFTRACE. Once the options are generated, it calls the assembler and linkage editor to create the required module (???FLLM). (Dialog Clist)

Option 4.7: Global Find and Edit

The Global Edit function allows the user to search and replace a string in a library (PDS). A list of all the members where the string was found (and replaced) is presented at the end of the process. This utility is useful outside the IADF environment whenever a mass change affecting many or all members of a partitioned data set is required.

This option is handled by two Clists.

GFIND Identifies the data set to be used as target for the search, creates the work parameters, and presents the summary of the search. (Dialog Clist)

MFC1EX2 Performs the actual search and replace in the member selected by GFIND. (Program)

Option 4.8: MFS Utility

ADFFMTL Handles MFS generation. This Clist calls the MFS Utility to generate MFS control blocks (DIF,DOF,MID,MOD) from macro source that probably comes from the rules generation process. (Service Clist)

Option 4.9: Data Dictionary Utilities

ADFDD Processes the following options:

1. DBD extraction from Data Dictionary and generation
2. PSB extraction from Data Dictionary and generation
3. Report generation (specific and glossary) from Data dictionary

All the processes are for batch or BMP executions of the IMS/VS Data Dictionary.

ADFOUT Processes requests for ADFIN and ADFOUT services. It provides conversion from dialog specifications to dictionary command inputs for the services mentioned above.

Option 4.10: DB2 RGLGEN process

It is handled by one Clist and one program:

ADFDBTWO Constructs a work table containing the list of DB2 tables to be extracted, consisting of member name, table name, table id, and the output library.

MFC1Y25 The IMSADF II RGLGEN utility, a program linked in IMSADF.ADFLOAD and customized by ADFID. This utility runs under DB2 command DSN. It can be executed in batch or foreground. In both cases, it runs under TSO's terminal monitor Clist. (Program)

OPTION 5: IADF User Profile

IADFPROF The controlling Clist for the user profile option of the menu. It presents two screens with all the variables sorted in the user's profile, allowing updates or refreshing the profile to reflect the most recent changes made to the IMSADF II version through the use of the Installation dialogs. The information is stored in the ISPPROF DDNAME as member ADF1PROF. (Dialog Clist)

OPTION 6: Tutorial Function

This function is implemented by the use of ISPF's Clist ISPTUTOR, contains several menu sections, one top panel, and no index.

IADF Administration Functions

This option presents different alternatives depending on the user level as registered in IADF. If the user is found in the SPECUSER table with Level 3, the IADF Administration Main Menu (IADFADMN) is presented when the user first logs on to IADF.

OPTION 1: Systems

Option 1.1: Libraries

In this section a particular row of the SYSTEMS table is inserted, updated, or deleted.

ADFPROF1 Presents a selection of installed systems under the same ADFID. The list contains the SYSID, the version, the DDNAME, and data set name that is used to store the ISPF tables from that SYSID. An example of the selection panel is found in Figure 5-7. Operations supported against a system are:

1. Initialize

This step displays the panel (shown in Figure 5-4) where all the information required for a system is collected: DDNAMES, data set names for all the libraries used in IADF processing, and the description of that SYSID. After the list is completed, the request is queued for execution. Upon exit from the Clist, request(s) are executed. A second Clist is invoked to perform SYSID initialization.

2. Reinitialize

This process is essentially the same as that described in 1 with the exception of the initial panel, which appears with the library names previously established.

3. Update

This step only changes library information and does not perform table creation or modification for the system.

4. Delete

This operation deletes the system from the list shown in Figure 5-7 and erases all the tables with the same SYSID.

If a multiple version SYSID is selected, it calls ADFVRV to for verification. (Dialog Clist)

ADFVRV This Clist performs the following functions:

- Verify that data sets required for migration are unique to this version.
- Insure that the DDNAME for the "ISPF Tables" data set is unique.
- Select the migration path(s) for this version.

In addition, it performs maintenance against the table ADFVER, which contains migration information for each version of a given SYSID. (Service Clist)

ADFCREAT Creates the basic tables that define a SYSID to IADF. Refer to “Table Initialization” on page 5-13 for their names and descriptions. If the libraries specified for the system do not exist, it calls the ADFINIT service.

Option 1.2: Developers

The affected table in this section is ssssPRad.

ADFPF02 Presents a selection of registered users for the system (these are Developers, not Administrators). It allows updates to the user entries if the Developer level accessing the Clist is 2 or 3; otherwise, no changes are permitted.

This table also contains two columns for migration authority with multiple version SYSIDs.

- The “FROM” column is used to verify the authority of the individual requesting migration in the source (“FROM”) SYSID. If the user has an entry in this column that agrees with the form of migration (COPY or MOVE), the request is allowed.
- The “TO” column is used in the target (“TO”) SYSID/VERSION to verify the authority of the requester. If the entry of the column coincides with the form of migration, the function is allowed.

Note: If the SYSID only consists of the “BASE” version, no verification of the “TO” system is performed. The reason is that the migration must then occur across SYSIDs and not as much control can be enforced.

The two columns may contain values denoting forms of migration permitted:

Blank	No migration is allowed for this developer (default)
M	Move form of migration allowed
C	Copy form of migration allowed
B	Both forms allowed

(Dialog Clist)

Option 1.3: Static DSNS

The affected table in this section is “ST”.

ADFRG1 Presents the contents of the Rules Generator foreground procedure if the table is found; otherwise it provides a default.

This table contains all the data sets required for a foreground execution of both the Rules Generator and the IMS/VS Message Format Services utility. The Clist provides maintenance function against the list to:

- Concatenate data sets
- Add new data set names
- Delete user changes
- Change data set names
- Update parameters passed to invoking programs

Most of the data set names are handled with symbolic references that are resolved at execution time of the table. (Dialog Clist)

Option 1.4: Dynamic DSNS

The affected table in this section is "DY".

ADFRG1 This Clist is described in "Option 1.3: Static DSNS" on page 11-12 and is also used to update the dynamic data sets table.

Option 1.5: BTS DSNS

The affected table in this section is "BTS".

ADFBTS Presents the contents of BTS utility foreground procedure if the BTS data set table is found or provides a default. The table contains all the data sets required for a foreground execution of the BTS program product. The Clist provides maintenance function against the list to:

- Concatenate data sets
- Add new data set names
- Delete user changes
- Change data set names
- Update parameters passed to invoking programs

Most of the data set names are handled with symbolic references that are resolved at execution time of the table. (Dialog Clist)

Option 1.6: Initialization

ADFINIT Presents a dialog panel depending on the option used: "L" is used for library initialization, "D" is for data base. The main functions of this Clist are:

1. Data set creation. It creates the required data sets either in batch or in foreground. It requires Administrator Level (2 or 3).
2. Browse data set. It is available only for libraries in foreground.
3. Delete data set. It performs either in batch or foreground, requiring Administrator Level (2 or 3).
4. Initialize data base. It loads IMSADF II data bases in batch or foreground via the IMSADF II batch driver.

(Service Clist)

OPTION 2: ADMINISTRATORS

The affected table in this section is SPECUSER.

ADFPROF4 Presents a selection list of all the Administrators by ADFID. It performs maintenance against the list to add, delete, or replace users. Each user has two keys: ADFID and TSOID. If a user is an Administrator in two versions, he must be registered separately for each one. This function is described in Chapter 4, "Add ADFID Administrators." (Dialog Clist)

OPTION 3: MIGRATION

These two options are handled by the same group of Clists. The distinction occurs upon disposition of the source: copy retains it, while migration deletes it.

- ADFMRG0** This initial control Clist (Phase I) verifies the requestor's authority to perform migration in the source, or "FROM", SYSID (all migrations) and the target ("TO") SYSID/VERSION (multiple version SYSIDs only). It also determines if a restart condition is active and calls a user exit if one was selected. It prepares the job stream which is submitted as Phase II of migration. (Dialog Clist)
- ADFMRGII1** This Clist is invoked in Phase II of migration. It runs in batch TSO, invoking IADF, and uses the control file built in Phase I data collection to perform the actual move or copy operations. It has checkpoint and restart capabilities and may call a specified user exit for each entry which was migrated. It can also call the exit at termination time if the user requested it. The results of all migration activities are recorded in a user-specified log data set. (Dialog Clist)
- ADFMRG** Primary data collection Clist for Phase I. It displays multiple panels depending on the options selected in the previous Clist and constructs a control file with the records selected by the user. (Dialog Clist)
- ADFMRG0A** Migration (move or copy) of non-standard members within the SYSID (those who do not follow IADF naming conventions). Migration of these members are described in "Non-Standard Element" on page 9-36. (Dialog Clist)
- ADFMRG2** Creates user source member(s), similar to TRANSIN for all the messages, help panels, routes, secondary destinations within the system. Member(s) are stored in a library and are automatically passed to Phase II of migration. (See "Update Message Data Base" on page 9-31.) (Service Clist)

OPTION 4: TABLES

The panels for the Clists described in this section are found in "Table Management" on page 10-31.

ADFTBCR Creates a table in the library to which it is directed by the current IADF SYSID. It obtains the table description (Keys, names etc.) from the IADF master table. (Service Clist)

ADFTBCOP Allows dynamic modifications to table structures to:

- Add or delete columns or keys
- Delete the table
- Create a key for a non-keyed table.

It verifies the user is an Administrator of Level 3 before allowing those changes to take place. (Service Clist)

ADFTBCP1 Copies a given table in the same system giving it a new name. (Service Clist)

OPTION 5: GLOSSARY

The affected tables in this section are GLOSINDEX and GLOSTERM.

MFC1GLOS Is a program linked in IMSADF.ADFLOAD which processes glossary terms. This service is described in “Glossary Services” on page 10-28. A menu is selected from the IADF Administration Main Menu and is presented with these options:

1. Display a glossary term.
2. Add a new term to the glossary
3. Modify an existing term in the glossary
4. Renumber an existing term. This option is used when many update operations against the term have polluted the sequence numbers, and no more insertions are available until renumbering takes place.

(Program)

Appendix A. Converting IADF Tables

Converting IADF Tables from Prior IMSADF II Releases

You may have SYSIDS defined under IADF in a previous IMSADF II release which you wish to use in IMSADF II Version 2 Release 2. The structure of several IADF tables has been modified in Version 2 Release 2. A conversion process has been provided for this purpose.

If you have never used IADF or you do not wish to use any SYSIDS registered under IADF in other IMSADF II releases, the following instructions do not apply. To add SYSIDS to the current release, refer to Chapter 5, “Installing an Application SYSID.”

If you are a former IADF user, there are **three** administrative tasks you must perform. The current task is highlighted:

1. Add Administrators to this ADFID
2. **Convert Application Systems (SYSIDS) to Version 2 Release 2 Format**
3. Convert SYSTEMS Table for Versioning. (See “Converting Tables to Versioning” on page A-6)

When all tasks are complete, development begins with the same libraries, data bases, users etc. which existed under the former release. Rows and columns pertinent to Version 2 Release 2 have been added to the new tables.

DO NOT USE IEBCOPY or ISPF Option 3.3 (Utilities/Copy) to copy your old SYSTEMS table or any members of the “ISPF Tables” libraries.

The table conversion process, which ensures that the structure of the tables is correct for Version 2 Release 2, performs the following functions:

- Reads the old SYSTEMS table
- Selects a SYSID to be converted
- Allocates a data set to contain the converted Version 2 Release 2 tables
- Reads the input “ISPF Tables” data set (name stored in SYSTEMS)
- Converts/writes the tables into the new “ISPF Tables” data set
- Adds the SYSID to the Version 2 Release 2 SYSTEMS table
- Produces a conversion log

The original table libraries are preserved.

You may convert **all** SYSIDS for **all** ADFIDS at one time, or you may convert them selectively. It is recommended that you perform conversion at one time, immediately after you establish the Administrators for the ADFID (see Chapter 4, “Add ADFID Administrators”).

Note: If you have multiple ADFIDS, you may still convert tables before all ADFIDS have been installed. The installation process does not update the SYSTEMS table.

To invoke the table conversion process, you **must** be designated as a Level 3 Administrator in Version 2 Release 2.

Invoke IADF under Version 2 Release 2 and you see Figure A-1.

```
IADFADMN ----- IADF ADMINISTRATION MAIN MENU -----
OPTION ==> TSO %IADFCONV

ADMINISTRATIVE LEVEL:3
KEY ==>                SYSID ==>        LEVEL:

  1 SYSTEMS          - Create or modify IMSADF II  systems (SYSID)
  2 MIGRATION        - Perform migration    TYPE ==> C ( M Move  C Copy )
  3 ADMINISTRATORS   - Specify administrative users
  4 TABLES          - Create, modify, and/or copy ISPF tables
  5 GLOSSARY         - Update glossary terms
  6 DEVELOPMENT      - Perform IMSADF II  development tasks
  X EXIT            - Terminate IADF

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT
```

Figure A-1. Invoke Table Conversion Process

On the OPTION line of the IADF Administration Main Menu, type:

TSO %IADFCONV

and press the ENTER key.

The IADF Table Conversion Panel is displayed, as shown in Figure A-2.

```

IADFCNVP ----- IADF TABLE CONVERSION -----
COMMAND ==>>

Data Set Name for:

ADFTLIB Version 2 Release 1 ==>>
ADFTLIB Version 2 Release 2 ==>>

SYSID Selection?          ==>> N      (Y = Select SYSIDS
                                         N = All SYSIDS processed)

If a SYSID exists in the SYSTEMS table for Version 2 Release 2

Replace SYSID?            ==>> N      (Y = Replace existing V2R2 SYSID
                                         N = Do not convert SYSID)

Copy IADF Glossary Tables? ==>> N      (Y = Copy V2R1 Glossary to V2R2
                                         N = Do not copy Glossary)

COMPLETE AND PRESS ENTER TO PROCESS OR END TO EXIT

```

Figure A-2. IADF Table Conversion Panel

Type the fully qualified name of the Version 2 Release 1 ADFTLIB data set (without quotes).

Type the fully qualified name of the Version 2 Release 2 ADFTLIB data set (without quotes). If the data set does not exist, it will be allocated for you.

If you wish **all** SYSIDS (determined by reading your input SYSTEMS table) to be converted to the output "ISPF Tables" data set, enter "N".

If you have already defined SYSIDS to IADF Version 2 Release 2 which were registered in the previous IADF release, they will be overlaid (in the SYSTEMS table) in the conversion process unless you respond "N" to "Replace SYSID?"

Normally you should convert existing SYSID tables from one release to another. A good procedure is to convert all SYSIDS initially, adding any new ones later. If you typed "Y" after "SYSID Selection List", you see Figure A-3.

You may have added your own terms to the glossary in the previous version of IMSADF II. To incorporate them in Version 2 Release 2, enter "y" beside "Copy IADF Glossary Tables?" You should only select this function once per release.

Once you have completed the IADFCNVP panel, press the ENTER key.

If you chose to see a SYSID Selection List, Figure A-3, is presented.

If all SYSIDS are to be converted, Figure A-4 is displayed.

IADFCSYS -----SYSTEMS TO BE CONVERTED -----			
COMMAND ==>			SCROLL ==> PAGE
Line Commands: S Select			
CMD	ADFID	SYSID	DDNAME Data Set Name
-	-	-	-
s	MFC1	SAMP	SAMPDD IMSADF.SAMP.ISPFTBLS
			IMSADF II SAMPLE
'	MFC1	BANK	BANKDD IMSADF.BANK.ISPFTBLS
			BANKING APPLICATION
'	ADF2	ABCD	DDABCD IMSADF.ADF2ABCD.ISPFTBLS
			ABCD SYSTEM
***** BOTTOM OF DATA *****			

Figure A-3. IADF SYSID Selection Panel

Enter "S" in the Line Command (CMD) column to select the SYSIDS you wish to be converted. If you have followed recommendations made in this guide, you have separate "ISPF Tables" data sets for each SYSID.

Note: If two or more SYSIDS share the same "ISPF Tables" library (not recommended by this guide), you **must** use separate libraries in the table conversion process.

Once you have selected all the SYSIDS you wish to be converted, press the END key.

You then see Figure A-4.

IADFCNVL ----- IADF TABLE CONVERSION -----	
COMMAND ==>	
ADFID:	MFC1
SYSID to be converted:	SAMP
INPUT ISPF TABLES DATA SET: TS0ID1.TEST.ISPFTBLS	
ENTER OUTPUT ISPF TABLES DATA SET:	
	==> tsoid1.v2r2.ispftb1s
(This data set will be allocated NEW)	
COMPLETE AND PRESS ENTER TO PROCESS OR END TO EXIT	

Figure A-4. New "ISPF Tables" Data Set panel

As a SYSID is selected for conversion, this panel is displayed, containing the SYSID name and the input "ISPF Tables" data set name. You must enter the new data set name for the output "ISPF Tables" data set for Version 2 Release 2. It cannot currently exist.

When you press ENTER,

- The IADF Table Conversion panel, originally shown in Figure A-2, is displayed with a message at the bottom indicating the SYSID currently being processed. The keyboard is locked until the conversion process is complete for the selected SYSIDS.
- Then Figure A-4 is shown, displaying the next SYSID to be converted and requesting its Version 2 Release 2 data set name.

The process described above continues until all requested SYSIDS have been converted. **All input "ISPF Tables" data sets remain unchanged.**

This function is concluded by displaying a complete report of all SYSIDS and all their tables, as shown in Figure A-5.

```

ISRBROBF TSOID1.SPFTEMP1.CNTL ----- LINE 000000 COL 001 080
COMMAND ==> SCROLL ==> PAGE
***** TOP OF DATA *****
-----
                IMSADF II
        CONVERSION ROUTINE FOR THE ISPF TABLES USED BY IADF
                FROM: VERSION 2 RELEASE 1
                TO: Version 2 Release 2
-----
SYSID SAMP WILL BE REPLACED IN IMSADF.ADFTLIB SYSTEMS TABLE.
CONVERSION PROCESS BEGINS FOR SYSID SAMP
        CONVERTED TABLES WILL BE PLACED IN THE TSOID1.V2R2.ISPFTBLS LIBRARY.
        TABLE SAMPBTS HAS BEEN SUCCESSFULLY CONVERTED.
        TABLE SAMPDBPA HAS BEEN COPIED, NO CONVERSION NECESSARY.
        TABLE SAMPDFPA HAS BEEN COPIED, NO CONVERSION NECESSARY.
        TABLE SAMPDY HAS BEEN SUCCESSFULLY CONVERTED.
        TABLE SAMPMSG HAS BEEN SUCCESSFULLY CONVERTED.
        TABLE SAMPOTPA HAS BEEN SUCCESSFULLY CONVERTED.
        TABLE SAMPPG01 HAS BEEN SUCCESSFULLY CONVERTED.
        TABLE SAMPPRMF HAS BEEN COPIED, NO CONVERSION NECESSARY.
        TABLE SAMPST HAS BEEN SUCCESSFULLY CONVERTED.
        TABLE SAMPSYS HAS BEEN SUCCESSFULLY CONVERTED.
        TABLE SAMPTRPA HAS BEEN COPIED, NO CONVERSION NECESSARY.
CONVERSION PROCESS COMPLETE FOR SYSID SAMP
-----
        TABLE TSOID1 HAS BEEN SUCCESSFULLY CONVERTED.
***** BOTTOM OF DATA *****

```

Figure A-5. IADF Table Conversion Report

These SYSIDS are now ready for development under Version 2 Release 2.

The Version 2 Release 2 SYSTEMS table has been updated with current information.

Converting Tables to Versioning

Multiple Version SYSIDS are added in Version 2 Release 2 using PTFs. Existing SYSTEMS and Developers tables must be converted when the PTFs are installed. For a discussion of versioning please refer to “Multiple Version SYSIDS” on page 5-1 and “Multiple Version SYSIDS” on page 9-5.

If you are a current user of IADF, you must convert your tables to Version 2 Release 2 using instructions found in “Converting IADF Tables from Prior IMSADF II Releases” on page A-1 **before** you perform this task.

If you are a new user of IMSADF II Version 2 Release 2 and install the PTFs **after** you install IMSADF II, you must perform this task only.

If you are a new user of IMSADF II Version 2 Release 2 and install the PTFs **before** you install IMSADF II, do not perform this task, since the SYSTEMS table is formatted properly. As you register SYSIDS, the developers tables will be built correctly.

To invoke this conversion process, you **must** be designated as a Level 3 Administrator.

Invoke IADF and you see Figure A-6.

```
IADFADMN ----- IADF ADMINISTRATION MAIN MENU -----
OPTION ==> TSO %IADFTBFX

ADMINISTRATIVE LEVEL:3
KEY ==>                SYSID ==>                LEVEL:

  1 SYSTEMS          - Create or modify IMSADF II  systems (SYSID)
  2 MIGRATION        - Perform migration    TYPE ==> C ( M Move  C Copy )
  3 ADMINISTRATORS   - Specify administrative users
  4 TABLES          - Create, modify, and/or copy ISPF tables
  5 GLOSSARY         - Update glossary terms
  6 DEVELOPMENT      - Perform IMSADF II  development tasks
  X EXIT             - Terminate IADF

ENTER OPTION NUMBER OR PRESS END KEY TO EXIT
```

Figure A-6. Invoke Versioning Conversion Clist

On the OPTION line of the IADF Administration Main Menu, type:

TSO %IADFTBFX

and press the ENTER key.

The IADF Table Maintenance panel is displayed, as shown in Figure A-7.

In order to perform this process you must be within IADF, because the tables are allocated automatically as you need them.

This clist may be invoked to convert **any** table in IMSADF II Version 2 Release 2. The table may be in IMSADF.ADFTLIB or in the "ISPF Tables" data set for the SYSID shown in Figure A-7. **The table is converted in its original library.** If you have added your own columns, they are dropped in the converted version. If an error is encountered in the process, you will see

"Error in conversion while processing table xxxxxxxx"

where the x's represent the name of the table.

```

ADFTBFX ----- TABLE MAINTENANCE: ADD COLUMNS -----
COMMAND ==>

Available Command: CAN Cancel

SYSID      ==> SAMP
VERSION    ==>
Table Type ==> SYST

DB Data Base Segments  AG Transaction Audits    DYB Dynamic Rules Files(B)
DF Data Base Fields    SG Segment Audits      STB Static Rules Files(B)
MA Map Segments        SB Subroutine Audits    HLP Help Panels
MF Map Fields          DG Dialogs                PFL Sysid Profiles
OU Output Segments     PR User Access Table    SDT Secondary Destinations
OF Output Fields       RT Additional Routes    USR Sysid Users
PS Pseudo Segments     SD Logical Terminals    ROU Message Routing
PF Pseudo Fields       TH SQLUSER Information  ALL Unique Segment IDs
BE Fast Path Segments  TR Transaction Tables   BTS BTS System Data Sets
BF Fast Path Fields    TX Secondary Transactions D2 RGLGEN Member List
TB DB2 Tables          OT Transaction Control  MSG Messages Table
TF DB2 Columns         DY Dynamic Rules Files(F) SYS Signon Pomenu Table
PG Program Driver Rule ST Static Rules Files(F) CROS References Segments/Trx
PROD Product Table     SYST Systems Table      SPEC Special Users
COMPLETE AND PRESS ENTER TO CONTINUE OR END TO EXIT

```

Figure A-7. Convert the SYSTEMS Table

Enter "SYST" since the SYSTEMS table is to be converted to the versioning format.

Press ENTER.

When the conversion is complete, you will see Figure A-7 repeated with the following message "Table SYSTEMS Converted."

This means that the column "VERSION" with the value "BASE" is added to each SYSID registered to form a base version.

Next you must convert the list of developers authorized within each SYSID (described in Chapter 6, "Adding Developers to the SYSID") to add columns for migration authority.

```

ADFTBFX ----- TABLE MAINTENANCE: ADD COLUMNS -----
COMMAND ==>

Available Command: CAN Cancel

SYSID      ==> SAMP
VERSION    ==> BASE
Table Type ==> PR

DB Data Base Segments  AG Transaction Audits      DYB  Dynamic Rules Files(B)
DF Data Base Fields    SG Segment Audits          STB  Static Rules Files(B)
MA Map Segments        SB Subroutine Audits      HLP  Help Panels
MF Map Fields          DG Dialogs                PFL  Sysid Profiles
OU Output Segments     PR User Access Table      SDT  Secondary Destinations
OF Output Fields       RT Additional Routes      USR  Sysid Users
PS Pseudo Segments     SD Logical Terminals      ROU  Message Routing
PF Pseudo Fields       TH SQLUSER Information    ALL  Unique Segment IDs
BE Fast Path Segments  TR Transaction Tables      BTS  BTS System Data Sets
BF Fast Path Fields    TX Secondary Transactions  D2   RGLGEN Member List
TB DB2 Tables          OT Transaction Control    MSG  Messages Table
TF DB2 Columns         DY Dynamic Rules Files(F)  SYS  Signon Pmenu Table
PG Program Driver Rule ST Static Rules Files(F)  CROS References Segments/Trx
PROD Product Table     SYST Systems Table        SPEC Special Users
COMPLETE AND PRESS ENTER TO CONTINUE OR END TO EXIT

```

Figure A-8. Convert Developers Table

On Figure A-8 you must enter the **SYSID** whose Developers table you wish to convert, and the **table type ("PR")**. Since you have just converted the SYSTEMS table, the VERSION entered is "BASE" for all SYSIDS.

Press ENTER.

When the conversion is complete, you will see Figure A-8 repeated with the following message "Table PR Converted."

Invoke this function as many times as you have SYSIDS registered under IADF.

Conversion adds a "B" to both the "FROM" and "TO" columns of the ssssPRad table, indicating that both MOVE and COPY forms of migration are permitted. It also adds a generic Administrator row to insure compatibility with migration architecture. Subsequently you may follow Chapter 6, "Adding Developers to the SYSID" to restrict access and form if you desire.

Appendix B. IADF Naming Conventions

Distributed Tables in ADFTLIB

Rule	Table Description
ADFDYTB	Model table for foreground dynamic rules (SYSID)
ADFDYTBB	Model for dynamic batch tables (SYSID)
ADFEDIAG	Store prototyping information
ADFMASR	Store definitions of IADF tables
ADFSTTB	Model table for foreground static rules (SYSID)
ADFSTTBB	Model static batch tables (SYSID)
ADFVER	Store migration paths for SYSID/VERSIONs
ADF1CMDS	IADF command table
BTSSAMP	Base table for BTS System table (SYSID)
EDIALOG	Base table for prototyping
GLOSDEFN	Glossary table
GLOSINDX	Glossary index table
INSTMEM	Customizing installation control table
PRODUCT	Store IMSADF II options (installation)
SPECUSER	Store the Administrators of IADF (installation)
SYSTEMS	Store IADF defined systems (installation)

Figure B-1. IMSADF II Master Table in ADFTLIB

Table Naming Conventions

Rule	Table Description
ssssnnnD	Store prototyping simulated data base (Actual data)
ssssnnnP	Store prototyping simulated data base (Pointer to data)
ssssAGtx	Store transaction audits
ssssALL	Store Unique Segment IDs
ssssBExx	Store a Fast Path segment (DBEK)
ssssBFxx	Store the fields of a Fast Path segment
ssssBTS	Store BTS Systems Data Sets
ssssCROS	Store cross-reference between segments
ssssDBxx	Store a data base segment (DBS, ESDS, KSDS, RRDS, RREK)
ssssDFxx	Store the fields of a data base segment
ssssDGdd	Store dialog simulation
ssssDY	Store Dynamic Rules Data Sets (Foreground)
ssssDYB	Store Dynamic Rules Data Sets (Batch)
ssssD2	Store members to be extracted from DB2 catalog
ssssHLP	Store the help panels
ssssMAxx	Store a MAP segment
ssssMFxx	Store the fields of a MAP segment
ssssMSG	Store Messages
ssssOFxx	Store the fields of a OUT segment
ssssOTtx	Store Override table for transaction
ssssOUxx	Store a OUT segment
ssssPFxx	Store the fields of a Pseudo segment
ssssPFL	Store Profiles for a SYSID
ssssPGpp	Store Transaction drivers
ssssPRaa	Developers Table - lists all the users registered to SYSID
ssssPSxx	Store a Pseudo segment (PS)
ssssROU	Store Alternate Routes in a SYSID
ssssRST	Migration restart data
ssssRTyy	Additional routes for a given route id (YY)

Figure B-2 (Part 1 of 2). Naming Conventions for SYSID Tables

Rule	Table Description
ssssSBzz	Store Audit subroutines
ssssSDzz	Logical input terminals for a given destination (ZZ)
ssssSDT	Store Secondary Destinations
ssssSGxx	Store segment audits
ssssSMpg	Store Secondary Option Menu (for prototyping)
ssssST	Store Static Rules Data Sets (Foreground)
ssssSTB	Store Static Rules Data Sets (Batch)
ssssSYS	Store Signon/POM information
ssssTBxx	Store a DB2 Table (TBL)
ssssTFxx	Store the fields of a DB2 Table
ssssTHxx	Store Table Handler User Information (DB2)
ssssTRtx	Store transactions
ssssTXtx	Store Secondary Transactions
ssssUSR	Store the users in a SYSID (Profile data base)
USERID	Store User data sets for BTS

Figure B-2 (Part 2 of 2). Naming Conventions for SYSID Tables

Lower-case characters above represent:

aaaa ADFID
aa First two characters of ADFID
dd Dialog ID
n Single digit for multiple panel transaction display screens
nnn Prototype data base ID
pp Cluster code for transaction drivers
pg Project Group
ss First two characters of System ID (SYSID)
ssss System ID (SYSID)
tx Transaction ID (TRXID)
vv Destination ID
xx Segment ID
yy Route ID
zz Subroutine ID

Other symbols are literals.

Rule Source Naming Conventions

Rule	Description
aaaaSSSS	System macro
ssKSSxxn	ISPF Prototype panel for Primary Key Selection for Prototype data base segment definition
ssKSTtxn	ISPF Prototype panel for Primary Key Selection for Prototype transaction execution
ssSISxxn	ISPF Prototype panel for default transaction display screen for Prototype data base segment definition
ssSITtxn	ISPF Prototype panel for default transaction display screen for Prototype transaction execution (no SIMAGE)
ssSitxn	ISPF Prototype panel for transaction display using screen image source
ssssAGtx	Transaction audits
ssssBExx	Fast Path segment
ssssBFxx	Fields of a Fast Path segment
ssssDBxx	Data base segment
ssssDFxx	Fields of a data base segment
ssssHHHH	Output for the user-defined help panel
ssssMAxx	MAP segment
ssssMFxx	Fields of a MAP segment
ssssOFxx	Fields of a OUT segment
ssssOUxx	OUT segment
ssssPFxx	Fields of a Pseudo segment
ssssPGpp	Transaction drivers
ssssPRaa	Profile
ssssPSxx	Pseudo segment
ssssROU	Store Alternate Routes in a SYSID
ssssRTyy	Route output for a transaction
ssssSBzz	Audit subroutines
ssssSDvv	Secondary Destinations
ssssSGxx	Segment audits
ssssSitx	Screen Image for transaction display panel

Figure B-3 (Part 1 of 2). Naming Conventions for Rule Source

Rule	Description
ssssTBxx	DB2 Table
ssssTFxx	Columns of a DB2 Table
ssssTRtx	Transactions
ssuserid	User for signon profile data base

Figure B-3 (Part 2 of 2). Naming Conventions for Rule Source

Lower-case characters above represent:

aaaa ADFID
aa First two characters of ADFID
dd Dialog ID
n Single digit for multiple panel transaction display screens
nnn Prototype data base ID
pp Cluster code for transaction drivers
pg Project Group
ss First two characters of System ID (SYSID)
ssss System ID (SYSID)
tx Transaction ID (TRXID)
vv Destination ID
xx Segment ID
yy Route ID
zz Subroutine ID

Other symbols are literals.

Appendix C. Migration Phase II Sample JCL

This appendix lists a sample of the job stream built by Phase II migration.

The job stream shown performs several tasks:

1. It copies the ISPF Profile data set (found in DDNAME ISPPROF) so that no ISPF enqueueing problems occur during migration. The ISPF profile information is required during migration. In addition the ISPPROF DDNAME must have data sets allocated as DISP=SHR.
2. It invokes ISPF through TSO batch and allocates the ISP- data sets. The data set names generated for the ISP- DDNAMES are customized with both ISPNODE and ADFNODE (values from the PRODUCT table or from your profile). If you have to do extensive editing of the job stream to accommodate your installation naming standards, you may wish to modify the skeleton used, ADFMIJCL.
3. It invokes IADF and passes its required dialog variables. The values of the variables are determined by the Phase I data collection process (see “Migration Examples” on page 9-12).

```

//TSOID1  JOB ('acctg'), job statement information appears here
//*
//*
//STEP1  EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*
//SYSUT3  DD UNIT=VIO,SPACE=(TRK,(1,1))
//SYSUT4  DD UNIT=VIO,SPACE=(TRK,(1,1))
//IN      DD DSN=TSOID1.ISPF.ISPPROF,DISP=SHR
//OUT     DD DSN=&&PROFILE,DISP=(NEW,PASS),UNIT=SYSDA,
//        SPACE=(TRK,(2,2,10)),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//SYSIN   DD *
        COPY I=IN,O=OUT
/*
//STEP2  EXEC PGM=IKJEFT01,COND=(4,LT)
//SYSPROC DD DISP=SHR,
//        DSN=IMSADF.ADFCLIB
//ISPPROF DD DSN=&&PROFILE,DISP=(OLD,DELETE)
//ISPPLIB DD DISP=SHR,
//        DSN=IMSADF.ADFPLIB
//        DD DISP=SHR,
//        DSN=SYS4.ISPF.ISPPLIB
//ISPSLIB DD DISP=SHR,
//        DSN=IMSADF.ADFSLIB
//        DD DISP=SHR,
//        DSN=SYS4.ISPF.ISPSLIB
//ISPLIB  DD DISP=SHR,
//        DSN=IMSADF.ADFMLIB
//        DD DISP=SHR,
//        DSN=SYS4.ISPF.ISPLIB
//ISPLLIB DD DISP=SHR,
//        DSN=IMSADF.ADFLOAD
//ISPTLIB DD DISP=SHR,
//        DSN=IMSADF.ADFTLIB
//        DD DISP=SHR,
//        DSN=SYS4.ISPF.ISPTLIB
//ISPFIL  DD DISP=(,PASS),UNIT=SYSDA,
//        DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),SPACE=(TRK,(5,1,40))
//ISPLOG  DD SYSOUT=*,DCB=(LRECL=133,BLKSIZE=133,RECFM=FB)
//ADFTLIB DD DISP=SHR,
//        DSN=IMSADF.ADFTLIB
//SYSTSPRT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSOUT  DD SYSOUT=*
//SYSPRINT DD SYSOUT=*,DCB=(LRECL=133,BLKSIZE=133,RECFM=FB)

```

Figure C-1 (Part 1 of 2). Migration Phase II Sample JCL

```

//SYSTSIN DD *
PROFILE PREFIX(TS0ID1)
ISPSTART CMD(%ADFMRII1 TH(N) -
    ADFID('MFC1') -
    SYSID('SAMP') -
    TDSN1('TS0ID1.TESN.SKELT') -
    TDSN2('TS0ID1.TESN.LOAD') -
    TDSN3('TS0ID1.TESN.SKELP') -
    TDSN4('') -
    SVSYSTO('SAMP') -
    XDDLIB8('SAMT') -
    XDSN8('TS0ID1.TESN.TABLES') -
    IADFLOG('TS0ID1.INITIAL.LOG') -
    IADFCNTL('TS0ID1.MIGRATE.CNTL') -
    SVFR('BASE') -
    TO('TEST') -
    FUN('C') -
    FULLM('N') -
    LMF('OFF') -
    SRC('Y') -
    LOAD('Y') -
    PRJM1('') -
    PRJM2('') -
    PRJM3('') -
    PRJM4('') -
    LIBM1('') -
    LIBM2('') -
    LIBM3('') -
    LIBM4('') -
    TYPM1('') -
    TYPM2('') -
    TYPM3('') -
    TYPM4('') -
    MIGTABLE('ON')) -
    NEWAPPL(ADF1)
/*

```

Figure C-1 (Part 2 of 2). Migration Phase II Sample JCL

Appendix D. Sample Migration User Exit

The following clist is distributed with IMSADF II for your use as a model for a migration user exit. It can be invoked in any phase (initialization, for each element, and termination).

Note: Sequence 00116000 contains an authorization check of "nnnnnnnn". Substitute any valid TSO userids if you want to specifically restrict restart authorization.

Exit processing is described in "Migration User Exit" on page 9-9.

```

PROC 0 TH(Y)                                00001000
/*-----USER EXIT SAMPLE-----*/          00002000
/*      THIS IS JUST AN EXAMPLE OF A MIGRATION EXIT      */ 00003000
/*      -----*/                                         */ 00004000
/* USE:                                                    */ 00005000
/* 1 - THIS EXIT IS A MODEL TO USE DURING MIGRATION FUNCTION. */ 00006000
/*   THE EXIT MAY BE CALLED AT THREE TIMES: INITIALIZATION, */ 00007000
/*   FOR EACH ELEMENT MIGRATED AND AT TERMINATION TIME.     */ 00008000
/* 2 - THIS EXIT IS ONLY AN EXAMPLE AND MUST BE TAILORED TO YOUR*/ 00009000
/*   INDIVIDUAL REQUIREMENTS. IF YOU MODIFY DATA IN THE TABLES*/ 00010000
/*   YOU MUST BE AWARE OF THE FORMAT OF THOSE TABLES AND   */ 00011000
/*   ENSURE THAT VALID DATA IS STORED IN THEM.             */ 00012000
/* 3 - VARIABLES USED FOR COMMUNICATION:                    */ 00013000
/*   IADFRS                                              */ 00014000
/*   ON ENTRY CONTAINS THE PHASE BEING CALLED:            */ 00015000
/*     I INITIALIZATION                                  */ 00016000
/*     E ELEMENT                                         */ 00017000
/*     T TERMINATION                                     */ 00018000
/*   ON EXIT THIS VARIABLE MUST BE CHANGED TO A NUMERIC    */ 00019000
/*   RETURN CODE, OTHERWISE THE EXIT WILL BE CANCELLED     */ 00020000
/*   VALID CODES DURING INITIALIZATION ARE:                */ 00021000
/*     0 CONTINUE THE OPERATION, NO CHANGES MADE TO THE LIST */ 00022000
/*     OF CATEGORIES                                       */ 00023000
/*     1 CONTINUE THE OPERATION, CHANGES MADE TO THE LIST  */ 00024000
/*     2 CANCEL THE MIGRATION RUN (REASON STORED IN IADFRESN)*/ 00025000
/*   CODES DURING ELEMENT MIGRATION ARE:                   */ 00026000
/*     0 CONTINUE, NO CHANGES MADE TO THE IADF TABLE     */ 00027000
/*     1 CONTINUE, CHANGES WERE MADE TO THE TABLE, UPDATE AND*/ 00028000
/*     RECREATE THE SOURCE                                  */ 00029000
/*     2 CANCEL THE MIGRATION FOR THIS ELEMENT, REASON MAY BE*/ 00030000
/*     PASSED IN IADFRESN VARIABLE                         */ 00031000
/*     3 CANCEL THE MIGRATION RUN (REASON STORED IN IADFRESN)*/ 00032000
/*   CODES DURING TERMINATION ARE:                        */ 00033000
/*     0 CONTINUE WITH TERMINATION                        */ 00034000
/*     NOT ZERO PUT TERMINATION MESSAGE (IN IADFRESN) IN THE */ 00035000
/*     LOG                                                  */ 00036000

```

Figure D-1 (Part 1 of 4). Model for a Migration User Exit

/*	IADFLST	*/	00037000
/*	CONTAINS THE LIST OF CATEGORIES TO BE PROCESSED	*/	00038000
/*	SEPARATED BY A COMMA: VALID NAMES ARE:	*/	00039000
/*	SYS - SYSTEM	*/	00040000
/*	SEG - SEGMENTS	*/	00041000
/*	TRX - TRANSACTIONS	*/	00042000
/*	DRV - DRIVERS (LINK EDIT)	*/	00043000
/*	AUD - AUDIT GROUPS	*/	00044000
/*	MSG - MESSAGES, HELP PANELS, ROUTES AND SECONDARY	*/	00045000
/*	EXT - EXTERNAL MEMBERS	*/	00046000
/*	IADFLOG	*/	00047000
/*	CONTAINS THE NAME OF THE MIGRATION LOG FULLY QUALIFIED*	*/	00048000
/*	IF THE EXIT CHANGES THIS NAME, IT SHOULD ALSO BE	*/	00049000
/*	RESPONSIBLE FOR ITS EXISTANCE	*/	00050000
/*	IADFCNTL	*/	00051000
/*	CONTAINS THE NAME OF THE MIGRATION CONTROL FILE	*/	00052000
/*	IF THE EXIT CHANGES THIS NAME, IT SHOULD ALSO BE	*/	00053000
/*	RESPONSIBLE FOR ITS EXISTANCE AND CONTENTS.	*/	00054000
/*	IADFCAT (ONLY USED FOR ELEMENT CALL)	*/	00055000
/*	IDENTIFICATION OF THE CATEGORY BEING MIGRATED:	*/	00056000
/*	SYS - SYSTEM	*/	00057000
/*	POM - PRIMARY OPTION MENU	*/	00058000
/*	SOM - SECONDARY OPTION MENU	*/	00059000
/*	SEG - SEGMENTS	*/	00060000
/*	TRX - TRANSACTIONS	*/	00061000
/*	STX - SECONDARY TRANSACTIONS	*/	00062000
/*	DRV - DRIVERS (LINK EDIT)	*/	00063000
/*	AUD - AUDIT GROUPS	*/	00064000
/*	MSG - MESSAGES	*/	00065000
/*	HLP - HELP PANELS	*/	00066000
/*	ROU - ALTERNATE ROUTES	*/	00067000
/*	SDT - SECONDARY DESTINATIONS	*/	00068000
/*	EXT - EXTERNAL MEMBERS	*/	00069000
/*	IADFCOI (USED ONLY IN ELEMENT CALL)	*/	00070000
/*	IDENTIFICATION OF THE COMPONENTS BEING MIGRATED. FOUR	*/	00071000
/*	NAMES SEPARATED BY BLANKS WILL BE PRESENT, IF THE	*/	00072000
/*	MEMBER NAME IS "NONE", THE MEMBER IS NOT PRESENT.	*/	00073000
/*	THE FIRST TWO NAMES APPLY TO SOURCE DATA, THE LAST TWO*	*/	00074000
/*	TO LOAD MODULES	*/	00075000
/*	IADFCATN (USED ONLY IN ELEMENT CALL)	*/	00076000
/*	ISPF TABLE ID OF THE ELEMENT BEING MIGRATED, THE NAMES*	*/	00077000
/*	OF KEYS AND DATA CAN BE RETRIEVED WITH AN ISPEXEC	*/	00078000
/*	TBQUERY INSTRUCTION. IF THE ELEMEMENT IS NOT STORED	*/	00079000
/*	IN A TABLE, THIS VARIABLE WILL CONTAIN BLANKS. IF	*/	00080000
/*	CHANGES ARE MADE TO THE DATA, THE TABLE SHOULD BE	*/	00081000
/*	UPDATED BEFORE EXITING AND MUST BE LEFT OPENED	*/	00082000

Figure D-1 (Part 2 of 4). Model for a Migration User Exit

```

/*      IADFRESN                                     */      00083000
/*      USED BY MIGRATION AT INITIALIZATION TO NOTIFY A      */      00084000
/*      RESTART CONDITION AND DATE TIME OF THIS RESTART      */      00085000
/*      AT TERMINATION TIME TO PASS INFORMATION ABOUT THE     */      00086000
/*      CONDITION OF THE PROCESS. ITS HAS THREE FORMATS:      */      00087000
/*      NORMAL NUMBER OF ELEMENTS MAXIMUM RETURN CODE        */      00088000
/*      CHKPT RECORD NUMBER MAXIMUM RETURN CODE              */      00089000
/*      ERROR RECORD NUMBER ERROR MESSAGE                    */      00090000
/*      USED BY THE EXIT TO PASS THE REASON OF THE CANCELATION*/      00091000
/*      OF THE PROCESS OR AN ENTRY                           */      00092000
/*      IADFEXT                                             */      00093000
/*      USED BY THE EXIT TO CHANGE THE NUMBER OF TIMES THAT  */      00094000
/*      PHASE II WILL CALL THIS SAME EXIT:                   */      00095000
/*      0 DO NOT CHANGE THE DEFAULTS SELECTED                */      00096000
/*      1 CALL THE EXIT ONLY FOR EACH ELEMENT                 */      00097000
/*      2 CALL THE EXIT ONLY AT TERMINATION                   */      00098000
/*      3 CALL THE EXIT TERMINATION AND FOR EACH ELEMENT      */      00099000
/*      4 DO NOT CALL THE EXIT IN PHASE TWO                   */      00100000
/*-----*/      00101000
CONTROL NOMSG NOFLUSH END(ENDO)      00102000
IF &TH = Y THEN CONTROL LIST CONLIST SYMLIST NOFLUSH MSG END(ENDO) 00103000
ISPEXEC CONTROL ERRORS RETURN /* TELL ISPF TO RETURN CONTROL */ 00104000
/* IF ERRORS ARE FOUND */ 00105000
ISPEXEC VGET (IADFRC IADFLST IADFLOG IADFCNTL IADFRESN) - 00106000
        SHARED /* GET COMMUNICATION VARIABLES FROM ISPF POOL 00107000
IF &LASTCC > 0 THEN DO /*VARIABLES ARE NOT INITIALIZED,EXIT*/ 00108000
    WRITE THIS PROGRAM ONLY RUNS AS AN EXIT FOR IADF MIGRATION 00109000
    EXIT CODE(12) 00110000
ENDO 00111000
IF &IADFRC = I THEN DO /* THIS IS INITIALIZATION TIME CALL*/ 00112000
    SET SYSDVAL = &STR(&IADFRESN) /* SEE IF A RESTART IS ACTIVE */ 00113000
    READDVAL CAUSE DATE TIME /* PARSE THE VALUES */ 00114000
    IF &CAUSE = RESTART THEN DO /* IF THIS IS A RESTART CALL */ 00115000
        IF &SYSUID ^= nnnnnnn THEN DO /* ONLY THIS USER IS AUTHORIZED */ 00116000
            SET IADFRC = 2 /* TO PERFORM RESTARTS. OTHER */ 00117000
            SET IADFRESN = + /* MUST BE CANCELLED */ 00118000
            &STR(INVALID USER SELECTED RESTART 00119000
            ISPEXEC VPUT (IADFRC IADFRESN) SHARED /* UPDATE VARIABLES */ 00120000
            EXIT CODE(0) /* MUST EXIT WITH ZERO CODE */ 00121000
        ENDO 00122000
    ENDO 00123000
ENDO 00124000

```

Figure D-1 (Part 3 of 4). Model for a Migration User Exit

```

ELSE IF &IADFRC = E THEN DO          /* ELEMENT MIGRATION          */ 00125000
    ISPEXEC VGET (IADFCAT IADFCATN) SHARED /* RETRIEVE TABLE INFO      */ 00126000
    IF &IADFCAT = SYS THEN DO        /*IF SYSTEM CAT CHANGE LRULE TO YES */ 00127000
        ISPEXEC TBSKIP &IADFCATN      /* FIRST READ ALL DATA IN    */ 00128000
        IF &LASTCC = 0 THEN DO        /* IF READ WAS SUCCESSFUL     */ 00129000
            SET LRULE = YES           /* CHANGE LRULE VALUE         */ 00130000
            ISPEXEC TBMOD &IADFCATN    /* CHANGE THE SYSTEM TABLE   */ 00131000
        ENDO                          00132000
        SET IADFRC = 1                /* NOTIFY THAT CHANGES WERE MADE */ 00133000
        ISPEXEC VPUT IADFRC SHARED    /* SAVE VALUE IN POOL         */ 00134000
        EXIT CODE (0)                 /* EXIT WITH ZERO             */ 00135000
    ENDO                              00136000
ELSE DO                              /* CONTINUE THE PROCESS       */ 00137000
    SET IADFRC = 0                    /* NO CHANGES WERE MADE      */ 00138000
    ISPEXEC VPUT IADFRC SHARED        /* SAVE VALUE IN POOL         */ 00139000
    EXIT CODE (0)                     /* ALWAYS EXIT WITH ZERO      */ 00140000
ENDDO                                00141000
ENDO                                  00142000

                                     /* THIS IS TERMINATION TIME   */ 00143000
SET IADFRC = 0                       /* CONTINUE WITH TERMINATION  */ 00144000
ISPEXEC VPUT IADFRC SHARED           /* SAVE RETURN CODE IN POOL   */ 00145000
EXIT CODE (0)                         /* EXIT WITH ZERO             */ 00146000

```

Figure D-1 (Part 4 of 4). Model for a Migration User Exit

Index

A

ADFFIELD IADF command
 See commands
ADFID
 administrators 4-1, A-1
ADERVER
 See tables, distributed in ADFTLIB
ADF1PROF
 See tables
administration main menu 3-12
administrative IADF users
 See administrators
administrators
 level 2 3-2, 4-1, 4-2, 5-2, 5-8, 6-1, 7-1, 8-1, 10-1, 10-31
 level 3 3-2, 4-1, 4-2, 5-2, 5-8, 6-1, 7-1, 8-1, 10-1, 10-28, 10-31, A-2, A-6
advanced user main menu
 See panels
 See panels listed by name, IADFADVM
allocate SYSID libraries 5-11
allow copy parameter 7-5, 7-17, 7-22, 7-23
AUDFIELD IADF command
 See commands

B

batch job
 data base initialization 8-7
 migration phase II 9-38
batch tables for SYSID
 See SYSID
bringing up administrator screens 3-3
browse source profile option 8-5, 9-12, 10-4
BTS data set tables
 See steps
 See SYSID

C

CICS
 considerations 7-2, 7-10, 7-14, 9-27
CICS/OS/VSE
 See CICS
clists
 ADFAG 11-2, 11-3, 11-4, 11-5
 ADFAUDM 11-5
 ADFBTS 11-7, 11-13
 ADFBTSA 2-11
 ADFBTS1 11-7
 ADFCOPY 2-11
 ADFCREAT 5-13, 11-12
 ADFDBTWO 11-10
 ADFDD 11-9

clists (*continued*)

ADFDRVR 11-4
ADFDRVR1 11-4
ADFD001A 11-3
ADFFMTL 11-9
ADFHLP 11-2
ADFHLP2 11-2
ADFINIT 2-11, 11-13
ADFMEXIT 9-3, D-1
ADFMRG 11-14
ADFMRG0 11-14
ADFMRG0A 11-14
ADFMRG2 11-14
ADFMRII1 11-14
ADFMMSG 11-3
ADFMMSG2 11-3
ADFOUT 11-10
ADFPRO 11-3
ADFPROA 11-4
ADFPROB 11-7
ADFPROF1 11-11
ADFPROF2 11-12
ADFPROF4 11-13
ADFPRO1 11-6
ADFPRO2 2-11, 11-6
ADFPRO2B 2-11, 11-6
ADFPRO4 11-6
ADFPRO5 11-6
ADFPRO8C 11-6
ADFPRO9C 11-6
ADFRG 11-5, 11-8
ADFRG1 11-12, 11-13
ADFROUT 11-4
ADFROU2 11-4
ADFRP 11-9
ADFSDT 11-4
ADFSDT2 11-4
ADFSEG 11-2
ADFSEGCR 11-9
ADFSEGEX 11-8
ADFSEG2 11-2
ADFSIPM 11-2
ADFSIPM1 11-2
ADFSUBM 11-8
ADFSYST 11-4
ADFSYST1 11-4
ADFTBCP1 11-14
ADFTBCR 11-14
ADFTBOP 11-14
ADFTRACC 11-9
ADFTRX 11-2
ADFTRXN1 11-2
ADFTRX1 11-2
ADFTRX2A 11-3

clists (*continued*)

- ADFUEXT 11-8
- ADFUEXTB 11-8
- ADFUSR 11-3
- ADFUSR2 11-3
- ADFVRV 11-11
- ADFXT 11-8
- audit generation utility 11-5
- dynamic generation utility 11-5
- EDGET 2-11
- GFIND 11-9
- IADF 2-1, 2-3, 2-14, 5-11
- IADF user profile 11-10
- IADFCONV A-2
- IADFDISP 11-9
- IADFTBFX A-7
- IAD2 2-1
- IMSADFM 2-1
- invoked from ISPF menu 2-1
- migration user exit 9-3, 9-21, D-1
- modifiable 2-11
- performance 2-12
- SHOW 2-11
- SHOWISPF 2-11

CMD column 4-3, 4-5, 5-12, 7-4

commands

- ADFFIELD 10-26
- AUDFIELD 10-26
- copy 7-5, 7-17, 7-22, 7-23
- FREEBTS 2-7
- FREEDY 2-7
- FREEST 2-7
- PATHC 5-8, 5-16, 9-6

conversion from prior IMSADF II releases 5-1, A-1

conversion to versioning 5-2, 6-3, A-6

convert existing Rules Generator source

See also Extract

MFS generation 10-23

screen image conversion 10-21

transaction overrides 10-23

copy command 7-5, 7-17, 7-22, 7-23

D

data bases

- allocate SYSID 8-1
- allocating 8-3, 8-4
- batch initialization 8-7
- foreground initialization 8-5
- initializing 8-5

data set allocation levels for TSO 2-7

data sets required for SYSID

- dynamic source 5-5, 5-6, 5-15
- ispf tables 5-6, 5-15, 6-1, A-1, A-6
- ispf tables data set 5-14
- rule source 5-5, 5-6, 5-15

DATABASE 2

See DB2

DBRC parms 2-11, 7-6

DB2

considerations 7-1, 7-2, 7-9, 7-11, 7-14, 8-1

DDNAMES 2-7, 9-9

deleting a SYSID 5-14

developers

level 1 3-2, 5-12, 6-1, 10-1

level 2 3-2, 6-1, 10-1

level 3 3-2, 6-1, 10-1

migration authority 5-1, 6-1, 6-2, 6-3, 6-6, 9-2, 9-4, 9-5, 9-17, 9-18, 9-20, 11-12, 11-14, A-8

table

conversion to versioning 6-3, A-8

generic user for migration 6-2, 6-5, 6-6, 9-4, 9-5, A-9

verification 11-12

development users

See developers

dynamic data set tables

See steps

See SYSID

dynamic rules conversion to IADF

See Extract

E

exits data set (user) 7-10

extract

data set name 7-20

dynamic rules 10-9

generate formats 10-23

prerequisites 10-6

screen image conversion 10-21

static rule conversion 10-6

static rules 2-12, 10-16

transaction control table creation 10-23

F

foreground selection profile option 8-5

foreground tables for SYSID

See SYSID

FREEBTS IADF command 2-7

FREEDY IADF command 2-7

FREEST IADF command 2-7

G

GFIND utility 2-13

glossary 10-28, A-3

I

- IADF logic description
 - administrators 11-13
 - audit rules 11-3
 - BTS data sets table 11-13
 - BTS utility 11-7
 - data layouts 11-2
 - data set initialization utility 11-13
 - DB2 generation utility 11-10
 - definition 11-2
 - dictionary utility 11-9
 - dynamic data sets table 11-13
 - extract 11-8
 - global edit utility 11-9
 - glossary 11-8, 11-15
 - IADF developer profile 11-11
 - IADF developers 11-12
 - IADF special profile management 11-11
 - IMSADF II help panels 11-2
 - IMSADF II trace utility 11-9
 - library display 11-9
 - message routing 11-4
 - message rules 11-3
 - mfstest utility 11-9
 - migration utility 11-13
 - primary option menu 11-2
 - profile data base 11-3
 - profile data base user 11-3
 - prototype control 11-7
 - prototyping dialog 11-6
 - prototyping overview 11-6
 - RGLGEN utility in IADF 11-10
 - rulegen data sets table 11-12
 - rules documentation 11-9
 - rules generation utility 11-5
 - screen image 11-6
 - screen image to ISPF 11-6
 - secondary destination 11-4
 - signon 11-2
 - submit 11-8
 - system statement control 11-4
 - table management service 11-14
 - transaction control 11-2
 - transaction driver control 11-4
 - tutorial 11-10
 - utilities overview 11-8
- IADF panel format 4-5
- IADF table naming conventions B-1
- IEBCOPY requirements 2-12, 9-6
- IMS/VS release considerations 2-11, 7-6, 7-14
 - affected skeletons 2-10
- installation data required for IADF 3-2
- installing
 - application system (SYSID) 5-1
- introduction to IADF 1-1
- IRLM parms 7-6
- ISPCTL0
 - See ISPF, DDNAMES

- ISPCTL1
 - See ISPF, DDNAMES
- ISPF
 - DDNAMES 2-2, C-1
 - ISPPREP utility 2-13
 - modifying Model panel(ISREMCLS) 2-7
 - modifying Primary Menu(ISR@PRIM) 2-3
 - modifying Tutorial panel(ISR00003) 2-3
 - modules in LPA 2-12
 - preparation for IADF 2-1
 - profile data set (ISPPROF) 10-1, C-1
 - variables 2-2
- ISPFIL0
 - See ISPF, DDNAMES
- ISPPROF 10-1, C-1
- ISR@PRIM ISPF primary menu modifications 2-3
- ISREMCLS ISPF model menu modifications 2-7
- ISR00003 ISPF tutorial panel modifications 2-3

K

- keyed table format 4-5

L

- levels of IADF users 3-2, 4-1, 4-2, 5-2, 5-8, 5-12, 6-1, 7-1, 8-1, 10-1, 10-28, 10-31, A-2, A-6
- LMF PROMOTE migration
 - See migration
- LMF requirements 2-12, 9-6
 - exit 9-9

M

- master rules SYSID 7-24, 10-4, 10-31
- MFS generation 10-23
- migration 2-12
 - "wrap" versions 9-5
 - ACBS 9-1, 9-38
 - alternate routes (messages) 9-34
 - audit exits 9-20, 9-36
 - audit load module processing 9-20, 9-29
 - checkpoint 9-3, 9-7, 9-15, 9-21, 11-14
 - concepts 9-1
 - control file 9-6, 9-38, 11-14
 - data sets required 5-6, 5-7, 9-17, 11-11
 - DBDS 9-1, 9-38
 - developers can invoke 10-5
 - dynamic rules 9-1
 - element selection 9-13
 - error conditions 9-15, 9-39
 - examples 9-12
 - exits (DL/I etc.) 9-20, 9-36
 - format library 9-1, 9-2
 - help panels (messages) 9-32
 - LMF PROMOTE differences 9-22, 9-24, 9-26, 9-28, 9-30, 9-32, 9-38
 - load form 9-2, 9-20
 - log 9-3, 9-7, 9-21, 9-39, 9-40, 11-14

migration (*continued*)

- message data base 9-31
- messages 9-33
- multiple version concepts 9-5
- multiple version SYSID 5-1, 5-2, 5-6, 5-7, 5-8, 5-16, 6-2, 6-3, 6-6, 9-4, 9-5, 9-6, 9-18, 9-19, 11-11, 11-12, 11-14
- no version SYSID 9-17
- non-standard element 9-8, 9-20, 9-36, 9-38
- paths 5-6, 5-8, 9-5, 9-6, 9-18, 9-19, 11-11
- Phase I (data collection) 9-2, 9-3, 9-6, 9-15, 9-17, 9-18, 9-19, 9-22, 9-23, 9-25, 9-27, 9-29, 9-31, 9-36, 9-38, 11-14
- Phase II 2-2, 2-10, 9-2, 9-3, 9-6, 9-7, 9-15, 9-21, 9-38, 9-40, 11-14
- phase II sample JCL C-1
- PSBS 9-1, 9-38
- restart 9-3, 9-6, 9-15, 9-39, 11-14
- secondary destinations (messages) 9-35
- segment processing 9-20, 9-23
- source form 9-2, 9-20
- standard element 9-8
- static rules 9-20
- SYSTEM processing 9-20, 9-22
- tables 9-6, 9-17
- transaction driver processing 9-20, 9-27
- transaction processing 9-20, 9-25
- two-phase 9-2
- user exit 9-3, 9-7, 9-21, 11-14, D-1

models 2-7

modifiable skeletons and clists

- data dictionary 2-10
- IADF data base allocations 2-10
- IADF library allocations 2-10
- IMS/VS batch procedure 2-10

modifying IADF/IAD2 clists 2-3

modifying ISPF supplied panels 2-3, 2-7, 2-14

Multiple ADFID environment 2-2, 2-14, 4-5, A-1

multiple version SYSID

- See conversion to versioning
- See developers, migration authority
- See migration, multiple version
- See SYSID, multiple version

MVS/XA 2-12

P

panels

- administration main 3-12, 4-2, 4-4, 5-3, 6-3, 7-3, 8-2, 10-2, 10-29, 10-31, A-2, A-7
- administrators 4-2
- advanced user main menu 3-11, 9-14, 10-5
- allocation error 5-9
- any table conversion A-8, A-9
- background job submission 8-7, 9-38
- bts data sets 7-15
- checkpoint request 9-21
- data base allocation 8-2

panels (*continued*)

- data base initialization 8-5
- definition menu 10-20, 10-24
- deletion of SYSID 5-14, 5-15
- developer table conversion A-9
- developers of SYSID 6-4, 6-6
- dynamic data sets 7-12
- dynamic tables background 7-21
- exits (DL/I etc.) 9-36
- extract menu 10-9, 10-16
- generate override fields 10-26
- generate segment control table 10-26
- generate transactions 10-24
- glossary menu 10-29
- HELP 3-8
- IADF main menu 3-6, 3-7, 3-9, 10-2, 10-6, 10-7, 10-20, 10-23, 10-24
- installation of SYSID 5-5
- library allocation 5-11, 5-12
- migration audit detail 9-29
- migration audit menu 9-29
- migration driver menu 9-27
- migration driver selection 9-27
- migration log 9-40
- migration menu 9-19
- migration message data base 9-31
- migration non-standard members 9-36
- migration of help panels 9-32
- migration of messages 9-33
- migration of routes in message data base 9-34
- migration of secondary destinations 9-35
- migration phase II 9-38
- migration restart 9-15
- migration segment detail 9-23
- migration segment menu 9-23
- migration system statement 9-22
- migration transaction detail 9-26
- migration transaction menu 9-25
- multiple path sysid 9-18
- Rules Generator background 7-18
- Rules Generator foreground 8-5, 10-17
- screen image model 10-22
- signon menu 10-21
- special processing routines 9-36
- sysid target libraries (no versioning) 9-17
- SYSIDS in ADFID 5-8
- system definition menu 5-3, 5-13, 6-4, 7-7, 7-11, 7-12, 7-14, 7-15, 7-18, 7-20, 8-2
- SYSTEMS table conversion A-8
- table copy 7-25, 10-34
- table create 10-32
- table menu 10-31
- table modify 10-33
- tutorial 3-10
- update data sets 7-4
- user profile 10-2

panels for data set activities explained 7-3

panels listed by name

ADFAGF1 8-6
 ADFDL01 5-14, 5-15
 ADFDRVMT 9-28
 ADFDYNT 10-10
 ADFDYN1 10-11, 10-12
 ADFGLOSM 10-29
 ADFGLOSU 10-30
 ADFIN0D 8-3, 8-4, 8-5, 8-7
 ADFIN0L 5-11, 5-12
 ADFMGAG 9-30
 ADFMGPM 9-19
 ADFMGPN5 9-36
 ADFMGPN1 9-37
 ADFMGPN2 9-17
 ADFMGPN3 9-16
 ADFMGPN4 9-18
 ADFMGPN5 9-21
 ADFMRG2P 9-31
 ADFPR02 10-3
 ADFPR03 10-4
 ADFPR04 5-4, 5-9
 ADFPR05 5-5
 ADFPR07 6-5, 6-6
 ADFPR08 4-2, 4-3, 4-4
 ADFPR09 10-32
 ADFPR10 10-32
 ADFPR11 7-25, 10-34
 ADFRGF1 10-18
 ADFSEGMT 9-24
 ADFSIG1 10-21
 ADFSUBP 8-8, 9-38
 ADFTBCP 10-33
 ADFTBFX A-8, A-9
 ADFTRXMT 9-26
 ADFT001 5-10
 ADFUEX 10-9, 10-15, 10-16, 10-19
 ADFUEXTP 10-17
 ADFU002 8-3
 ADFVRV1 5-16
 ADFVRV2 5-17
 ADFVRV3 5-7
 ADFVRV4 5-8
 AUDMGRP 9-29
 BTSPNLS 7-16
 DRVMGRP 9-27
 EDIT 10-22
 GENUPDS 7-4, 7-5
 HAGD8010 3-10
 HAGMD0 3-8
 HLPMPGR 9-32
 IADFADMN 3-12, 4-2, 4-4, 5-3, 6-3, 7-3, 8-2, 10-2,
 10-7, 10-29, 10-31, A-2, A-7
 IADFADVM 3-11, 9-14
 IADFCNVL A-5
 IADFCNVP A-3
 IADFCSYS A-3
 IADFCTBL 10-25

panels listed by name (*continued*)

IADFDEF 10-20, 10-24
 IADFMAIN 3-6, 3-7, 3-9, 10-2, 10-6, 10-7, 10-20,
 10-23, 10-24
 IADFSYS 5-3, 5-13, 6-4, 7-7, 7-11, 7-12, 7-15, 8-2
 IADFUTIL 10-8
 ISR@PRIM 2-4
 ISRBROBF 9-40, 10-12, A-6
 ISREMCLS 2-8, 2-9
 ISRFPM 9-37
 ISR00003 2-6
 MSGINL 10-13, 10-14
 MSGMGR 9-33
 ROUMGR 9-34
 RULPNLD 7-13, 7-21
 RULPNLS 7-8, 7-18
 SDTMGR 9-35
 SEGMGRP 9-23
 SYSMGRP 9-22
 TRXMGRP 9-25
 TRXOVRT 10-26
 TRXOVT1 10-27
 TRXPNL 10-25, 10-28
 parms
 allow copy 7-5, 7-17, 7-22, 7-23
 updating program invocation 7-5, 7-6, 7-10
 PATHC
 See commands
 See migration.paths
 performance 2-7, 2-12, 7-17, 10-26
 preparing ISPF for IADF 2-1
 profile menu, description of 3-5
 programs
 MFC1EXD 11-2
 MFC1EX1 11-2, 11-3, 11-6, 11-7
 MFC1EX2 11-9
 MFC1EX5 11-7
 MFC1GLOS 11-8, 11-15
 MFC1TABL 11-8
 virtual table generators description 11-1

R

rules conversion to IADF
 See Extract
 Rules Generator tables
 See steps
 See SYSID
 rules source
 extract ISPF tables from
 See Extract

S

- skeletons 7-17, 7-24
 - affected by IMS/VS release changes 2-10
 - data dictionary modifications 2-10
 - modifiable IADF 2-10
- special administrative users
 - See administrators
- SPECUSER
 - See tables, distributed in ADFTLIB
- ssssBTS
 - See tables, SYSID
- ssssDY
 - See tables, SYSID
- ssssDYB
 - See tables, SYSID
- ssssPRad
 - See tables, SYSID
- ssssST
 - See tables, SYSID
- ssssSTB
 - See tables, SYSID
- static rule data set tables
 - See steps, RULEGEN
 - See SYSID, RULEGEN
- static rules conversion to IADF
 - See Extract
- steps
 - adding administrators to ADFID 3-1, 4-1
 - adding SYSID 3-1, 5-1
 - adding SYSID developers 3-1, 6-1
 - converting developers table to versioning 6-3, A-6
 - converting SYSID from prior release 5-1, A-1
 - converting SYSTEMS table to versioning 5-2, A-6
 - establishing BTS data sets for SYSID 2-7, 3-1, 7-1, 7-14
 - establishing dynamic data sets for SYSID 2-7, 3-1, 7-1, 7-11, 7-20
 - establishing RULEGEN data sets for SYSID 2-7, 3-1, 7-1, 7-7, 7-18
 - initializing data bases 3-1, 8-1
- structure of IADF
 - See IADF logic description
- SYSID
 - See also tables, SYSID
 - "wrap" versions 5-2
 - adding users 6-1
 - batch tables 2-7, 7-17, 7-18, 7-20
 - BTS data sets 7-14
 - converting from prior release A-1
 - converting to multiple versions 5-2, A-6
 - developers table (PR) 6-3, A-6
 - distributed model tables B-1
 - dynamic data sets 2-7, 2-12, 7-11
 - establishing data set activities 2-7, 2-12, 7-1, 10-31
 - foreground tables 2-7, 2-12, 7-7, 7-11, 7-14
 - IADF profiles 10-1
 - installing on IADF 5-1
 - migration paths 5-1

SYSID (continued)

- multiple version 5-1, 5-5, 5-6, 5-7, 5-8, 5-16, 6-2, 6-3, 6-6, 9-2, 9-4, 9-5, 9-6, 9-18, 9-19, 11-11, 11-12, 11-14, A-8
- multiple version concepts 5-2
- RULEGEN data sets 2-7, 2-12, 7-7
- table initialization 5-13
- table maintenance 7-25
- table migration 9-6, 9-17
- table naming conventions 5-13, B-2
- table propagation 7-25, 10-31
- users
 - See developers
- system libraries
 - IMSADF.ADFCLIB 2-2
 - IMSADF.ADFMLIB 2-2
 - IMSADF.ADFPLIB 2-2, 2-13
 - IMSADF.ADFSLIB 2-2, 7-17, 7-24
 - IMSADF.ADFTLIB 2-2, 5-1, 5-16, 9-5, A-2, A-7, B-1
- SYSTEMS
 - See tables, distributed in ADFTLIB

T

tables

- ADF1PROF 10-1
- conversion from prior releases A-1
- conversion to versioning 5-2, 6-3, A-6
- distributed in ADFTLIB B-1
 - ADFVER 5-1, 5-8, 5-16, 9-5, 9-18, 9-19, 11-11
 - SPECUSER 4-1
 - SYSTEMS 5-1, 5-2, 9-5, A-1, A-6
- glossary A-3
- management 10-31
 - copy 7-25, 10-34
 - create 10-32
 - modify 10-33
- naming conventions B-4
- naming conventions for rule source B-4
- size 2-13
- SYSID
 - See also SYSID
 - copying foreground to batch 7-22
 - Initialization 5-13
 - naming conventions 5-13, B-2
 - ssssALL 9-6
 - ssssBTS 2-7, 2-12, 7-2, 7-14
 - ssssCTxx 10-26
 - ssssDY 2-7, 2-12, 7-2, 7-11
 - ssssDYB 7-2, 7-20
 - ssssPRad 6-1, 6-2, 9-4, 9-20, 11-12, A-6, A-8
 - ssssRST 9-15, 9-16
 - ssssST 2-7, 2-12, 7-2, 7-7, 10-34
 - ssssSTB 7-2, 7-18
 - table migration 9-6, 9-17
- transaction control table 2-12, 10-23

- TSO data set allocation levels 2-7, 2-12, 7-17
- TSO performance
 - See performance
- TSO region size 2-7, 7-17
- TSO/E 2-11, 2-12
- tutorial menu, description of 3-5

U

- user exit
 - See migration
- user profile update 10-1
- users in SYSID (developers)
 - See tables, SYSID
- utilities menu, description of 3-5

V

- variables used in migration user exit 9-9
- versioning
 - See multiple version SYSID
- Virtual I/O (VIO) restrictions 2-2

Z

- ZTEMPF
 - See ISPF, variables
- ZTEMPN
 - See ISPF, variables

You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

Your comments will be sent to the author's department for whatever review and action, if any, are deemed appropriate.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name, company, mailing address, and date:

What is your occupation? _____

Number of latest Newsletter associated with this publication: _____

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail directly to the address in the Edition Notice on the back of the title page.)

Reader's Comment Form

Cut or Fold Along Line

Fold and tape

Please Do Not Staple

Fold and tape



NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, N.Y.

POSTAGE WILL BE PAID BY ADDRESSEE:

International Business Machines Corporation
Department 8D8
220 Las Colinas Boulevard
Irving, Texas 75039-5513



Fold and tape

Please Do Not Staple

Fold and tape



Publication Number
SH20-6602-02

File Number
S/370/4300-32

Program Number
5665-348



SH20-6602-02

