Event Replicator for Adabas

Using the Adabas Event Replicator Subsystem

Version 3.6.1

July 2018

ADABAS & NATURAL

This document applies to Event Replicator for Adabas Version 3.6.1 and all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Document ID: ARF-GUI-361-20190409

Table of Contents

Preface

This document describes the Adabas Event Replicator Subsystem and how to maintain Replication definitions in the Replicator system file using the Adabas Event Replicator Subsystem.

It covers the following topics:

About the Adabas Event Replicator Subsystem	Describes basics of the Adabas Event Replicator Subsystem screens, including how to access them and how to control access to the Adabas Event Replicator Subsystem. Global values and methods of loading and unloading the Replicator system file are also described.
Maintaining Initial-State Definitions	Describes maintenance tasks for initial-state definitions using the Adabas Event Replicator Subsystem screens.
Maintaining Destination Definitions	Describes maintenance tasks for destination definitions using the Adabas Event Replicator Subsystem screens.
Maintaining Input Queue (IQUEUE) Definitions	Describes maintenance tasks for IQUEUE definitions using the Adabas Event Replicator Subsystem screens.
Maintaining Resend Buffer Definitions	Describes maintenance tasks for resend buffer definitions using the Adabas Event Replicator Subsystem screens.
Maintaining Transaction Filter Definitions	Describes maintenance tasks for transaction filter definitions using the Adabas Event Replicator Subsystem screens.
Maintaining Subscription Definitions	Describes maintenance tasks for subscription definitions using the Adabas Event Replicator Subsystem screens.
Maintaining SFILE Definitions	Describes maintenance tasks for SFILE definitions using the Adabas Event Replicator Subsystem screens.
Maintaining GFB Definitions	Describes maintenance tasks for global format buffer (GFB) definitions using the Adabas Event Replicator Subsystem screens.
Initiating a Replay Request Using the Adabas Event Replicator Subsystem	Describes how you can initiate synchronized and replay-only replay processing using the Adabas Event Replicator Subsystem and the ADARPL utility.
Reviewing and Managing the PLOG Data Set List	Describes how you can use the Adabas Event Replicator Subsystem to review and maintain the list of PLOG data sets stored in the Replicator system file. It also describes the information stored for each PLOG data set.
Submitting Event Replicator Target Adapter Requests	Describes how you can populate, refresh (clear), and delete RDBMS tables using the Event Replicator Target Adapter once it is activated. You can submit requests to populate RDBMS tables with initial-state data, clear tables, or drop tables.

1 About this Documentation

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
`Monospace font`	Identifies service names and locations in the format *`folder.subfolder.service`*, APIs, Java classes, methods, properties.
`Italic`	Identifies: Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
`Monospace font`	Identifies: Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
|	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the | symbol.
[ ]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [ ] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information and Support

Software AG Documentation Website

You can find documentation on the Software AG Documentation website at **http://documentation.softwareag.com**. The site requires credentials for Software AG's Product Support site Empower. If you do not have Empower credentials, you must use the TECHcommunity website.

Software AG Empower Product Support Website

If you do not yet have an account for Empower, send an email to empower@softwareag.com with your name, company, and company email address and request an account.

Once you have an account, you can open Support Incidents online via the eService section of Empower at **https://empower.softwareag.com/**.

You can find product information on the Software AG Empower Product Support website at **https://empower.softwareag.com**.

To submit feature/enhancement requests, get information about product availability, and download products, go to **Products**.

To get information about fixes and to read early warnings, technical papers, and knowledge base articles, go to the **Knowledge Center**.

If you have any questions, you can find a local or toll-free number for your country in our Global Support Contact Directory at **https://empower.softwareag.com/public_directory.asp** and give us a call.

Software AG TECHcommunity

You can find documentation and other technical information on the Software AG TECHcommunity website at **http://techcommunity.softwareag.com**. You can:

- Access product documentation, if you have TECHcommunity credentials. If you do not, you will need to register and specify "Documentation" as an area of interest.
- Access articles, code samples, demos, and tutorials.
- Use the online discussion forums, moderated by Software AG professionals, to ask questions, discuss best practices, and learn how other customers are using Software AG technology.
- Link to external websites that discuss open standards and web technology.

Data Protection

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

2 About the Adabas Event Replicator Subsystem

The Adabas Event Replicator Subsystem provides an online interface you can use to maintain definitions for the Event Replicator for Adabas. These definitions are stored in the Replicator system file associated with a specific Event Replicator Server.

Accessing the Adabas Event Replicator Subsystem

You can access the Adabas Event Replicator Subsystem with or without the Adabas Online System (AOS). The procedure described in this section describes how to access it without AOS. For information on accessing it from AOS, read *Accessing the Adabas Event Replicator Subsystem from AOS*, in *Event Replicator for Adabas Administration and Operations Guide*.

To access the Adabas Event Replicator Subsystem without using AOS:

Verify that the Event Replicator Server and Replicator system file have been defined appropriately, as described in the Event Replicator for Adabas installation documentation (in the *Event Replicator for Adabas Installation Guide*.

1 Use ADALOD to load a Replicator system file into the Event Replicator Server. The Replicator system file is an Adabas system file. For more information about the ADALOD functions and parameters pertinent to Event Replicator for Adabas, read *ADALOD LOAD Parameters* in *Event Replicator for Adabas Reference Guide*. For more information about Adabas system files, read about the FILE parameter of the ADALOD LOAD function in your Adabas utilities documentation (the *Adabas Utilities Manual*).

   Specifically, you need to specify the REPLICATOR parameter on the ADALOD LOAD function:

   ```
   ADALOD LOAD FILE=nnnn,REPLICATOR
   ```

   where *nnnn* is the number of the Replicator system file. For example:

   ```
   ADALOD LOAD FILE=89,REPLICATOR
   ```

   The Replicator system file, when loaded, stores the Event Replicator Server initialization parameters. These parameters are defined in the Replicator system file using the online Adabas Event Replicator Subsystem. If the Replicator system file is not present, the Event Replicator initialization parameters are read from the DDKARTE of the Event Replicator Server job.

2 Start Natural with the Natural `LFILE` parameter set as follows:

```
LFILE=(89,dbid,file)
```

where *dbid* is the database ID of an Event Replicator Server you created during installation and *file* is a Replicator system file. The LFILE parameter can be specified either as a dynamic parameter or inside a Natural SYSPARM profile.

3 Logon to the SYSRPTR library by entering:

```
LOGON SYSRPTR
```

4 Enter the following command:

```
MENU
```

The Adabas Event Replicator Subsystem Main Menu appears.

Controlling Access to the Adabas Event Replicator Subsystem

You can control access to the Adabas Event Replicator Subsystem using the initialization exit, subprogram N-IEXIT. This exit is a Natural subprogram that runs automatically, whenever a user attempts to access the Adabas Event Replicator Subsystem. Based on Natural code you supply in the exit using the exit parameters, you can:

- restrict specific users from accessing the Adabas Event Replicator Subsystem
- identify the Replicator system file initially used by a user or users.

For complete information on coding Natural subprograms, read your Natural documentation. Once you have supplied code in the N-IEXIT subprogram, the ID of any user attempting to access the Adabas Event Replicator Subsystem is passed to the exit. If the #RESPONSE parameter is set to a non-zero value for that user, they cannot access the Adabas Event Replicator Subsystem .

A sample of the N-IEXIT is shown below.

```
0010 ***************************
0020 *   INITIALIZATION EXIT   *
0030 ***************************
0040 DEFINE DATA PARAMETER
0050 01 #USER              (A08)   /* USER ID
0060 01 #REPTOR-DBID       (N05)   /* REPTOR ID
0070 01 #REPTOR-FNR        (N05)   /* REPTOR SYSTEM FILE
0080 01 #PARM-1            (A40)   /* DATA
0090 01 #RESPONSE          (B02)   /* USER EXIT RESPONSE CODE
0100 01 #VERSION           (A04)   /* ONLINE SYSTEM VERSION
0110 END-DEFINE
0120 *
0130 #RESPONSE = H'0000'           /* NON-ZERO WILL TERMINATE
```

```
0140 ESCAPE ROUTINE
0150 END                                                    ↩
```

You can use the following parameters while coding the Natural subprogram:

Parameter	Description
#PARM-1	Reserved for future use.
#REPTOR-DBID	The database ID of the Event Replicator Server whose Replicator system file you want to maintain when the Adabas Event Replicator Subsystem starts. Once you are using the Adabas Event Replicator Subsystem, you can change this setting using the **Set LFILE Parameters** screen.
#REPTOR-FNR	The file number of the Replicator system file you want to maintain when the Adabas Event Replicator Subsystem starts. Once you are using the Adabas Event Replicator Subsystem, you can change this setting using the **Set LFILE Parameters** screen.
#RESPONSE	A non-zero response code will cause the Adabas Event Replicator Subsystem to terminate. By setting this to a non-zero number you can restrict access to the Adabas Event Replicator Subsystem for the user identified in the USER parameter or for the Adabas Event Replicator Subsystem version specified in the VERSION parameter..
#USER	The user ID of a potential Adabas Event Replicator Subsystem user.
#VERSION	The version of the Adabas Event Replicator Subsystem.

The Adabas Event Replicator Subsystem Main Menu

From the main menu of the Adabas Event Replicator Subsystem screens, you can select options that allow you to maintain any Event Replicator for Adabas definitions you need. Definitions can be added, reviewed, modified, copied, or deleted.

```
14:49:23      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
Vers 3.4.1                        Main Menu                          M-RP0010
DBID 1954  File 89

                  Code    Function
                  ----    --------------------------------
                    A     Administrator Functions
                    D     Destination Definitions
                    F     Transaction Filter Definitions
                    G     Global Format Buffer Definitions
                    I     Initial-State Definitions
                    Q     Input Queue Definitions
                    R     Resend Buffer Definitions
                    S     Subscription Definitions
                    ?     Help
                    .     Exit
                  ----    --------------------------------

           Code ... _

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                      ↩
```

The following table describes the options on this menu. To select an option, enter its associated code in the **Code** field on the screen.

Code	Allows you to:
A	Specify global values for the Adabas Event Replicator Subsystem, identify Adabas databases to which connections should be attempted (or not be attempted), identify the Replicator system file you wish to maintain using the Adabas Event Replicator Subsystem, and load or unload definitions from that Replicator system file if you choose.
D	Maintain destination definitions.
F	Maintain transaction filter definitions.
G	Maintain global format buffer definitions.
I	Maintain initial-state definitions.
Q	Maintain IQUEUE definitions.
R	Maintain resend buffer definitions.
S	Maintain subscription definitions.
?	Get help on this menu.
.	Exit the Adabas Event Replicator Subsystem.

Maintaining Replication Definitions Using Adabas Event Replicator Subsystem

You can use the Adabas Event Replicator Subsystem to perform add, copy, modify, and delete replication definitions in a Replicator system file.

The relevant sections describing the individual maintenance tasks are the following:

Maintaining Initial-State Definitions	Describes maintenance tasks for initial-state definitions using the Adabas Event Replicator Subsystem screens.
Maintaining Destination Definitions	Describes maintenance tasks for destination definitions using the Adabas Event Replicator Subsystem screens.
Maintaining Input Queue (IQUEUE) Definitions	Describes maintenance tasks for IQUEUE definitions using the Adabas Event Replicator Subsystem screens.
Maintaining Resend Buffer Definitions	Describes maintenance tasks for resend buffer definitions using the Adabas Event Replicator Subsystem screens.
Maintaining Transaction Filter Definitions	Describes maintenance tasks for transaction filter definitions using the Adabas Event Replicator Subsystem screens.
Maintaining Subscription Definitions	Describes maintenance tasks for subscription definitions using the Adabas Event Replicator Subsystem screens.
Maintaining SFILE Definitions	Describes maintenance tasks for SFILE definitions using the Adabas Event Replicator Subsystem screens.
Maintaining GFB Definitions	Describes maintenance tasks for global format buffer (GFB) definitions using the Adabas Event Replicator Subsystem screens.

Identifying, Loading, and Unloading the Replicator System File

The Event Replicator for Adabas definitions are stored in the Replicator system file associated with a specific Event Replicator Server. While using the Adabas Event Replicator Subsystem, you can:

- Change which Replicator system file's definitions you are updating.
- Unload Replicator system file definitions to Natural workfile 1.
- Load definitions from Natural workfile 1 to a Replicator system file.

Before you attempt any of these functions, verify that all Event Replicator Servers and Replicator system files are defined, as described in *Event Replicator for Adabas Installation Steps for z/OS Systems* in *Event Replicator for Adabas Installation Guide*.

This section covers the following topics:

- Changing the Event Replicator Server and Replicator System File

- Unloading a Replicator System File
- Loading a Replicator System File

Changing the Event Replicator Server and Replicator System File

To start working on the replication definitions for a different Replicator system file and Event Replicator Server:

1 Select option **A** from the Adabas Event Replicator Subsystem Main Menu.

The Administration menu appears.

```
19:30:31      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2017-03-15
                                Administration                        M-RP1100


                       Code    Function
                       ----    ------------------------------
                         D     Database ID
                         I     Perform Initial-State
                         P     PLOG Information
                         R     Initiate Replay
                         S     System Functions
                         T     Target Adapter
                         V     Global Values
                         X     SYSAOS Replicator Management
                         ?     Help
                         .     Exit
                       ----    ------------------------------
               Code ...  _


Command ==>



Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                    Menu
```

2 Select option S from the Administration menu.

The System Functions submenu appears.

```
14:54:26      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                              System Functions                       M-RP1000


                         Code    Function
                         ----    -------------------------------
                           L     Set LFILE Parameters
                           U     Unload Replication Definitions
                           I     Load Replication Definitions
                           ?     Help
                           .     Exit
                         ----    -------------------------------




                  Code ...  _



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                              Menu  ↩
 ↩
```

3 Select option L on the System Functions submenu.

The Set LFILE Parameters screen appears.

```
14:54:58      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                            Set LFILE Parameters                     M-RP1010



                  Replicator Server DBID ......... _1954
                  Replicator System File ......... __89
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save                                    Menu  ↩
 ↩
```

4 Tab to the **Replicator Server DBID** field and enter the database ID of the Event Replicator Server whose Replicator system file you want to maintain. The database must have been previously defined. For more information on defining an Event Replicator Server, read *Event Replicator for Adabas Installation Steps for z/OS Systems* in *Event Replicator for Adabas Installation Guide*.

5 Tab to the **Replicator system file** field and enter the number of the Replicator system file you want to maintain. This must be the number of the Replicator system file loaded into the Event Replicator Server you specified in the previous step.

6 Press PF5 to save your settings.

All updates you make using the Adabas Event Replicator Subsystem will now be made to the Replicator system file you specified on this screen.

Unloading a Replicator System File

This section provides instructions for running the RPULD utility from the Adabas Event Replicator Subsystem. For information on running it in batch mode, read *RPULD and RPLOD Utilities* in *Event Replicator for Adabas Reference Guide*.

To unload Replicator system file definitions to Natural workfile 1:

1 Select option **A** from the Adabas Event Replicator Subsystem Main Menu.

The Administration menu appears.

```
19:30:31      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2017-03-15
                               Administration                        M-RP1100 ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
                        Code    Function                                        ↩
                        ----    --------------------------------                ↩
                          D     Database ID                                     ↩
                          I     Perform Initial-State                           ↩
                          P     PLOG Information                                ↩
                          R     Initiate Replay                                 ↩
                          S     System Functions                                ↩
                          T     Target Adapter                                  ↩
                          V     Global Values                                   ↩
                          X     SYSAOS Replicator Management                    ↩
                          ?     Help                                            ↩
                          .     Exit                                            ↩
                        ----    --------------------------------                ↩
                 Code ...  _                                                    ↩
                                                                                ↩
                                                                                ↩
Command ==>                                                                     ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                   Menu
```

2 Select option S from the Administration menu.

The System Functions submenu appears.

```
14:54:26      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                              System Functions                       M-RP1000


                         Code    Function
                         ----    -------------------------------
                           L     Set LFILE Parameters
                           U     Unload Replication Definitions
                           I     Load Replication Definitions
                           ?     Help
                           .     Exit
                         ----    -------------------------------




                 Code ...  _



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                              Menu  ↩
 ↩
```

3 Select option U on the System Functions submenu.

The Unload Replication Objects screen appears.

```
14:59:47      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                      Unload Replication Objects                     RPULD



Entity ........ __
Name .......... *_______


Replicator System File
   DBID ....... 1954__    Fnr ........ 89____
   Password ...           Cipher .....

Options
   Unload related objects ......... Y
   Only subscriptions of status ... _




Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Exec                                  Menu  ↩
 ↩
```

4 Fill in the fields on the Unload Replication Objects screen, as described in the following table.

Field Name	Instructions	Required?
Entity	Optionally, identify a specific definition type to unload. Valid values are: ■ blank: all definitions are selected ■ "DE": destination definitions are selected ■ "GF": global format buffer definitions are selected ■ "IQ": input queue definitions are selected ■ "IS": initial-state definitions are selected ■ "RB": resend buffer definitions are selected ■ "SB": subscription definitions and their associated SFILE definitions are selected ■ "TF": transaction filter definitions are selected If no definition type is selected, all definitions with names that match the **Name** field specifications are unloaded.	No
Name	Specify the name of the definitions to select. Wildcards *, >, and < can be used. For example, `AB4*` selects all definitions with names starting with the letters "AB4". As another example, `AB4>` selects all definitions with names that sort (alphanumerically) before names starting with the letters "AB4".	No

Field Name	Instructions	Required?
	If no **Name** specifications are made, all definitions of the types specified by the **Entity** field are unloaded.	
DBID	Specify the database ID of the database with the Replicator system file containing definitions you want to unload to Natural workfile 1.	Yes
Fnr	Specify the file number of the Replicator system file containing definitions you want to unload to Natural workfile 1.	Yes
Password	Specify the password necessary to access the Replicator system file identified in the **DBID** and **Fnr** fields. If no password is specified, no password is used to access the Replicator system file.	No
Cipher	Specify the cipher code necessary to access the Replicator system file identified in the **DBID** and **Fnr** fields. If no cipher code is specified, no cipher code is used to access the Replicator system file.	No
Unload related objects	Indicate whether or not related definitions called by a selected definition should also be unloaded. Valid values are "Y" (all related definitions should also be unloaded) or "N" (all related definitions should not be unloaded unless they are selected directly through other field settings).	No
Only subscriptions of status	Identify the version of the selected definitions that should be unloaded. Valid values are "C" (current versions) or "S" (scheduled versions). If no value is entered, all versions of the selected definitions are unloaded.	No

5 When all fields have been specified appropriately, press the PF5 key to run the program to unload the specified Replicator system file definitions to Natural workfile 1.

Loading a Replicator System File

This section provides instructions for running the RPLOD utility from the Adabas Event Replicator Subsystem. For information on running it in batch mode, read *RPULD and RPLOD Utilities* in *Event Replicator for Adabas Reference Guide*.

To load a Replicator system file with definitions from Natural workfile 1:

1 Select option **A** from the Adabas Event Replicator Subsystem Main Menu.

The Administration menu appears.

```
19:30:31      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2017-03-15
                                Administration                        M-RP1100 ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
                        Code    Function                                        ↩
                        ----    --------------------------------                ↩
                          D     Database ID                                     ↩
                          I     Perform Initial-State                           ↩
                          P     PLOG Information                                ↩
                          R     Initiate Replay                                 ↩
                          S     System Functions                                ↩
                          T     Target Adapter                                  ↩
                          V     Global Values                                   ↩
                          X     SYSAOS Replicator Management                    ↩
                          ?     Help                                            ↩
                          .     Exit                                            ↩
                        ----    --------------------------------                ↩
                 Code ...  _                                                    ↩
                                                                                ↩
                                                                                ↩
Command ==>                                                                     ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                    Menu
```

2 Select option S from the Administration menu.

The System Functions submenu appears.

```
14:54:26      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                              System Functions                       M-RP1000


                         Code    Function
                         ----    -------------------------------
                           L     Set LFILE Parameters
                           U     Unload Replication Definitions
                           I     Load Replication Definitions
                           ?     Help
                           .     Exit
                         ----    -------------------------------




                 Code ...  _



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                              Menu  ↩
 ↩
```

3 Select option I on the System Functions submenu.

The Load Replication Objects screen appears.

```
15:00:41        ***** A D A B A S  REPLICATION SUBSYSTEM *****       2013-02-28
                          Load Replication Objects                   RPLOD



Entity ........ __
Name .......... *_______


Replicator System File
   DBID ....... 1954__    Fnr ........ 89____
   Password ...           Cipher .....

Options
   Replace objects ................ N
   Only subscriptions of status ... _




Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Quit        Exec                                    Exit  ↩
 ↩
```

4 Fill in the fields on the Load Replication Objects screen, as described in the following table.

Field Name	Instructions	Required?
Entity	Optionally, identify a specific definition type to load. Valid values are: ■ blank: all definitions are selected ■ "DE": destination definitions are selected ■ "GF": global format buffer definitions are selected ■ "IQ": input queue definitions are selected ■ "IS": initial-state definitions are selected ■ "RB": resend buffer definitions are selected ■ "SB": subscription definitions and their associated SFILE definitions are selected ■ "TF": transaction filter definitions are selected If no definition type is selected, all definitions with names that match the **Name** field specifications are loaded.	No
Name	Specify the name of the definitions to select. Wildcards "*", ">", and "<" can be used. For example, `AB4*` selects all definitions with names starting with the letters "AB4". As another example, `AB4>` selects all definitions with names that sort (alphanumerically) before names starting with the letters "AB4".	No

Field Name	Instructions	Required?
	If no **Name** specifications are made, all definitions of the types specified by the **Entity** field are loaded.	
DBID	Specify the database ID of the Replicator system file to which you want to load definitions.	Yes
Fnr	Specify the file number of the Replicator system file to which you want to load definitions.	Yes
Password	Specify the password necessary to access the Replicator system file identified in the **DBID** and **Fnr** fields. If no password is specified, no password is used to access the Replicator system file.	No
Cipher	Specify the cipher code necessary to access the Replicator system file identified in the **DBID** and **Fnr** fields. If no cipher code is specified, no cipher code is used to access the Replicator system file.	No
Replace Objects	Indicate whether or not existing definitions should be replaced with the newly loaded definitions. Valid values are "Y" (existing definitions should be replaced) or "N" (existing definitions should not be replaced).	No
Only subscriptions of status	Identify the version of the selected definitions that should be loaded. Valid values are "C" (current versions) or "S" (scheduled versions). If no value is entered, all versions of the selected definitions are loaded.	No

5 When all fields have been specified appropriately, press the PF5 key to run the program to load the Replicator system file with the definitions from Natural workfile 1.

Controlling Connections to Adabas Databases

By default, Event Replicator for Adabas attempts to connect with any Adabas database encountered during an Event Replicator Server session. You may, however, prefer to control these connection attempts using an Adabas database connection definition. This section describes how to maintain these definitions using the Adabas Event Replicator Subsystem.

- Listing Adabas Database Connection Definitions
- Adding a Connection Setting for an Adabas Database
- Modifying the Connection Definition for an Adabas Database
- Activating and Deactivating Databases and Files
- Copying the Connection Definition for an Adabas Database

Listing Adabas Database Connection Definitions

To list the connection definitions for Adabas databases in theAdabas Event Replicator Subsystem:

1 Select option **A** from the Adabas Event Replicator Subsystem Main Menu.

The Administration menu appears.

```
19:30:31      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2017-03-15
                                 Administration                      M-RP1100 ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
                        Code    Function                                        ↩
                        ----    --------------------------------                ↩
                          D     Database ID                                     ↩
                          I     Perform Initial-State                           ↩
                          P     PLOG Information                                ↩
                          R     Initiate Replay                                 ↩
                          S     System Functions                                ↩
                          T     Target Adapter                                  ↩
                          V     Global Values                                   ↩
                          X     SYSAOS Replicator Management                    ↩
                          ?     Help                                            ↩
                          .     Exit                                            ↩
                        ----    --------------------------------                ↩
                 Code ...  _                                                    ↩
                                                                                ↩
                                                                                ↩
Command ==>                                                                     ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                   Menu
```

2 Select option D from the Administration menu.

The Database IDs screen appears showing all of the Adabas database connection definitions in the Adabas Event Replicator Subsystem.

```
15:01:27      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                                 Database IDs                        M-RP1160

 Sel DBID Connect   Sel DBID Connect   Sel DBID Connect   Sel DBID Connect
 ---------------    ----------------   ----------------   ----------------
  _     42    Y      _                  _                  _
  _     62    Y      _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit  Add                -     +                     Menu  ↩
 ↩
```

Adding a Connection Setting for an Adabas Database

To add a connection definition for an Adabas database:

1 Select option **A** from the Adabas Event Replicator Subsystem Main Menu.

The Administration menu appears.

```
19:30:31      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2017-03-15
                                 Administration                      M-RP1100

                          Code    Function

                          ----    --------------------------------

                            D     Database ID

                            I     Perform Initial-State

                            P     PLOG Information

                            R     Initiate Replay

                            S     System Functions

                            T     Target Adapter

                            V     Global Values

                            X     SYSAOS Replicator Management

                            ?     Help

                            .     Exit

                          ----    --------------------------------

                     Code ...  _



Command ==>



Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                  Menu
```

2 Select option D from the Administration menu.

The Database IDs screen appears showing all of the Adabas database connection definitions in the Adabas Event Replicator Subsystem.

3 Press PF4 on the Database IDs screen.

A dialog appears.

```
Enter Database ID and Connect

Database ID ..... _____
Connect ......... _  (Y or N)

PF3 to Exit without saving
                                ↩
```

4 Tab to the **Database ID** field on the pop-up window and specify the database ID of the Adabas database to which this connection definition applies. This is the equivalent of specifying the `DATABASE ID` parameter in the Event Replicator Server startup job.

5 Tab to the **Connect** field and specify whether or not the Event Replicator Server should initiate an attempt to connect to the Adabas database at Event Replicator Server startup. Valid values are "Y" or "N".

If you specify "Y", the Event Replicator Server will initiate an attempt to connect to the Adabas database after the Event Replicator Server starts.

If you specify "N", the Event Replicator Server will *not* initiate an attempt to connect to the Adabas database. The database may connect with the Event Replicator Server at some point, but the Event Replicator Server will not initiate the connection at Event Replicator Server startup.

This is the equivalent of specifying the `DBCONNECT` parameter in the Event Replicator Server startup job.

6 Press the Enter key to save the connection definition and return to the Database IDs screen.

The database connection definition is added.

Modifying the Connection Definition for an Adabas Database

To modify a connection definition in the Adabas Event Replicator Subsystem:

1 List the connection definitions in the Adabas Event Replicator Subsystem, as described in *Listing Adabas Database Connection Definitions*, elsewhere in this guide.

The destinations are listed on the Database IDs screen.

2 Locate the definition you want to modify on the screen and enter an **M** in the **Sel** column for that definition.

A dialog pops up displaying the settings for the connection definition you selected. For information on modifying this screen, read the description of *Adding a Connection Setting for an Adabas Database*, elsewhere in this section.

3 When all modifications have been made, press Enter to save the changes and return to the Database IDs screen.

Activating and Deactivating Databases and Files

You can use Adabas Online System (AOS) to activate and deactivate databases or individual files within a database. For more information, read *Activating and Deactivating Replication Definitions and Databases*, in the *Event Replicator for Adabas Administration and Operations Guide*

Caution: Be careful when you activate and deactivate replication definitions and databases, especially if replication is ongoing at the time. Whenever you activate or deactivate definitions or databases, you run the risk of altering what data is replicated and how that replication occurs. If the Event Replicator Server receives data from an Adabas database for which it has no active definitions, replication simply does not occur.

Copying the Connection Definition for an Adabas Database

To copy a connection definition in the Adabas Event Replicator Subsystem:

1 List the connection definitions in the Adabas Event Replicator Subsystem, as described in *Listing Adabas Database Connection Definitions*, elsewhere in this guide.

The destinations are listed on the Database IDs screen.

2 Locate the definition you want to copy on the screen and enter a **C** in the **Sel** column for that definition.

A dialog appears.

```
Enter Database ID and Connect

Database ID ..... ___12
Connect ......... N  (Y or N)

PF3 to Exit without saving
                                  ↩
```

3 Specify a new, unique Adabas database ID for the copy of the definition and press Enter.

The connection definition is copied and the copy appears on the Database IDs screen.

Deleting the Connection Definition for an Adabas Database

To delete a connection definition in the Adabas Event Replicator Subsystem:

1 List the connection definitions in the Adabas Event Replicator Subsystem, as described in *Listing Adabas Database Connection Definitions*, elsewhere in this guide.

The destinations are listed on the Database IDs screen.

2 Locate the definition you want to delete on the screen and enter a **D** in the **Sel** column for that definition.

A pop-up dialog appears prompting you to indicate whether you really want the definition deleted or not.

```
Delete 10?       (Y or N) _

(PF3 to exit without deleting)

                                  ↩
```

3 Specify either "Y" or "N" to indicate whether you really want the definition deleted. Then press Enter.

The definition is deleted if you specify "Y"; otherwise it is not.

Populating a Database With Initial-State Data

In addition to submitting requests for initial-state data from your application, you can submit initial-state requests from the Adabas Event Replicator Subsystem. (For information on submitting initial-state data requests from your application, read *Event Replicator Client Requests* in *Event Replicator for Adabas Application Programmer's Reference Guide*.)

To submit a request for initial-state data from the Adabas Event Replicator Subsystem:

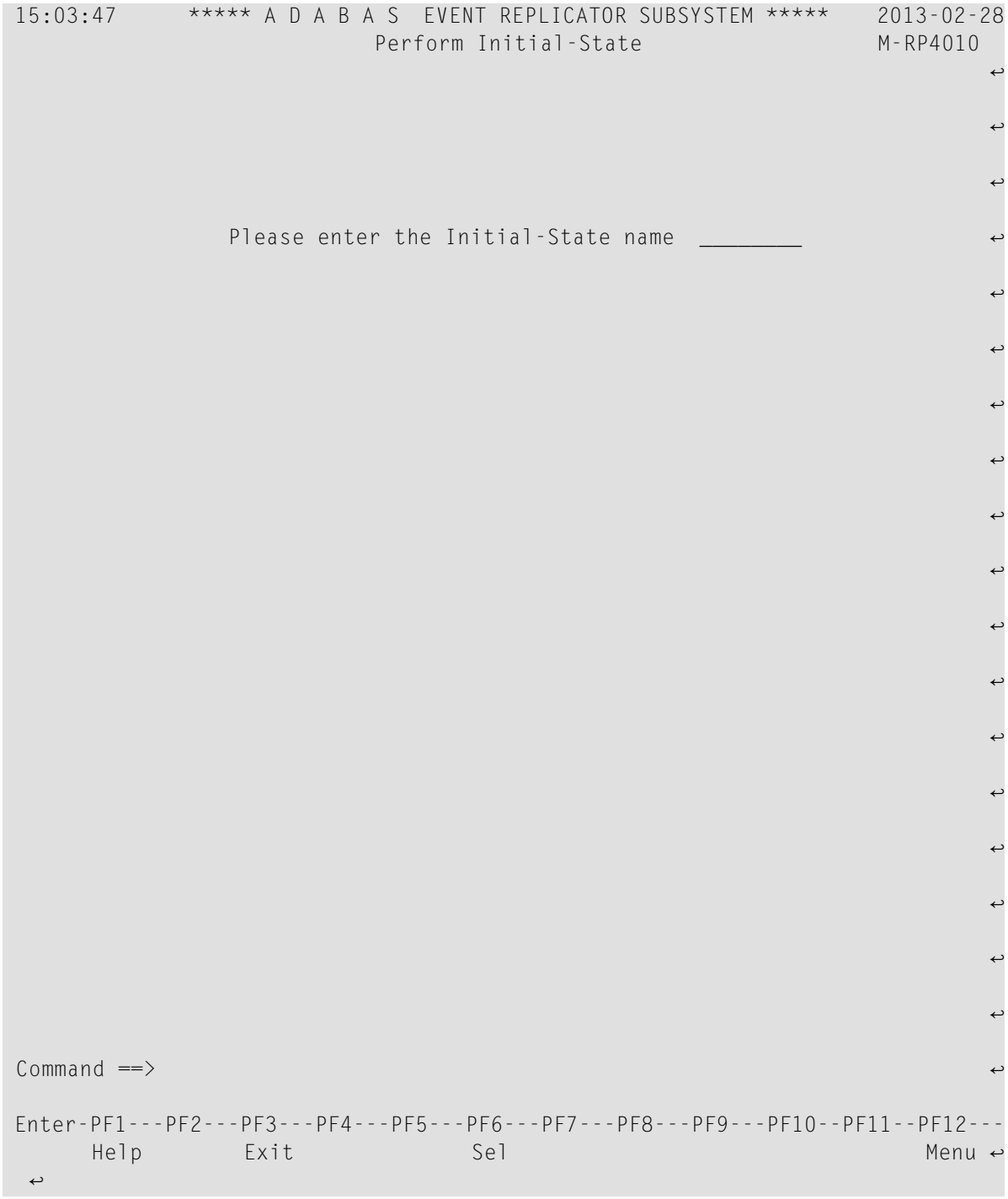
1 Select option **A** from the Adabas Event Replicator Subsystem Main Menu.

The Administration menu appears.

```
19:30:31      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2017-03-15
                                Administration                        M-RP1100 ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
                        Code    Function                                        ↩
                        ----    --------------------------------                ↩
                          D     Database ID                                     ↩
                          I     Perform Initial-State                           ↩
                          P     PLOG Information                                ↩
                          R     Initiate Replay                                 ↩
                          S     System Functions                                ↩
                          T     Target Adapter                                  ↩
                          V     Global Values                                   ↩
                          X     SYSAOS Replicator Management                    ↩
                          ?     Help                                            ↩
                          .     Exit                                            ↩
                        ----    --------------------------------                ↩
                  Code ...  _                                                   ↩
                                                                                ↩
                                                                                ↩
Command ==>                                                                     ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                    Menu
```

2 Select option **I** from the Adabas Event Replicator Subsystem Administration menu.

The **Perform Initial-State** screen appears.



3 Type the name of the initial-state definition you want used to populate the database and press Enter. You can also use the PF6 function key to select an initial-state definition from a list of the available definitions.

The **Perform Initial-State** screen appears.

```
15:04:41      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                            Perform Initial-State                   M-RP4012


                      Initial-State Name ....... I040155

Sel  IDBID IFile I        Selection Criteria   or   ISN List
--- ------ ----- - ------------------------------------------------------------
 _ + 40    155   I
 _
 _
 _
 _
 _
 _
 _
 _
 _


Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Sub                                     Menu  ↩
 ↩
```

4 The **Perform Initial-State** screen lists all of the files in the initial-state definition. All files that are ready to have the initial-state run on their data appear in the list with a plus sign (+) before their associated database ID (IDBID field). If they are not ready, a plus sign does *not* appear.

5 If any files in this list require an ISN filter (they appear in the list with an "I" in the **I** column), they will not be ready for the initial-state run until you specify an ISN or ISN list. To do this, complete the following steps:

1. Type an "E" in the **Sel** column corresponding to the file that requires an ISN filter.

   A **Specify ISNs** screen appears.

```
15:04:41     ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                               Specify ISNs                          M-RP4013
                                                                                ↩
                                                                                ↩
                                                                                ↩
                     Initial-State Name ....... I040155                         ↩
                                                                                ↩
                                                                                ↩
_______________________________________________________________________________ ↩
                                                                                ↩
_______________________________________________________________________________ ↩
                                                                                ↩
_______________________________________________________________________________ ↩
                                                                                ↩
_______________________________________________________________________________ ↩
                                                                                ↩
_______________________________________________________________________________ ↩
                                                                                ↩
_______________________________________________________________________________ ↩
                                                                                ↩
_______________________________________________________________________________ ↩
                                                                                ↩
_______________________________________________________________________________ ↩
                                                                                ↩
_______________________________________________________________________________ ↩
                                                                                ↩
_______________________________________________________________________________ ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
Command ==>                                                                     ↩

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Accpt                                     Menu ↩
  ↩
```

2. On the blank lines on this screen, type in the ISN or list of ISNs you want to use as a filter for the file. Separate multiple ISNs using commas (no blanks). If you want to specify a range of ISNs, specify them in ascending order in matched (open and closed) parentheses. For example, the following ISN list selects ISNs 111 and 222, all the ISNs between 333 and 666, and ISN 9999.

```
111,222,(333,666),9999
```

3. When all ISNs have been specified as you like, press the PF5 function key to accept the list and return to the **Perform Initial-State** screen.

6 If any files in this list require a value buffer (they appear in the list with an "S" in the **I** column), they will not be ready for the initial-state run until you specify value buffer. To do this, complete the following steps:

1. Type an "E" in the **Sel** column corresponding to the file that requires a value buffer.

   A **Specify Value Buffer** screen appears.

```
15:04:41     ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                           Specify Value Buffer                     M-RP4015
                                                                              ↩
                                                                              ↩
Initial-State Name ....... IO40155         Value Buffer Length....... _____ ↩
                                                                              ↩
_____________________________________________________________________________ ↩
_____________________________________________________________________________ ↩
_____________________________________________________________________________ ↩
_____________________________________________________________________________ ↩
_____________________________________________________________________________ ↩
_____________________________________________________________________________ ↩
_____________________________________________________________________________ ↩
_____________________________________________________________________________ ↩
_____________________________________________________________________________ ↩
_____________________________________________________________________________ ↩
_____________________________________________________________________________ ↩
_____________________________________________________________________________ ↩
                                                                              ↩
                                                                              ↩
Command ==>                                                                   ↩

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit  Mode      Accpt                                     ↩
Menu   ↩
```

2. On the blank lines on this screen, type in the value buffer, using all of the characters required by each field in the corresponding initial-state definition's search buffer. Use blanks to fill out the values in the value buffer for each field specified by the search buffer. For example, if a field in the search buffer can be ten bytes long, but the actual value is only three bytes long, use blanks to fill out the remaining seven bytes in the value buffer specification.

Note: You can use the PF4 function key to select character or hexadecimal mode.

3. In the **Value Buffer Length** field, specify the total length of the value buffer. Be sure to get this total from the total possible sizes of all the fields in the search buffer. This field is required.
4. When the value buffer and length have been specified as needed, press the PF5 function key to accept it and return to the **Perform Initial-State** screen.

7 When all files have a plus sign (+) in front of them, press the PF5 function key to submit the initial-state request. Messages will appear describing the success or failure of the request.

Setting Global Values

You can set the following global values in the Replicator system file using the Adabas Event Replicator Subsystem screens.

To set global values for the Adabas Event Replicator Subsystem:

1 Select option **A** from the Adabas Event Replicator Subsystem Main Menu.

The Administration menu appears.

```
19:30:31      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2017-03-15
                                 Administration                      M-RP1100 ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
                        Code    Function                                        ↩
                        ----    --------------------------------                ↩
                          D     Database ID                                     ↩
                          I     Perform Initial-State                           ↩
                          P     PLOG Information                                ↩
                          R     Initiate Replay                                 ↩
                          S     System Functions                                ↩
                          T     Target Adapter                                  ↩
                          V     Global Values                                   ↩
                          X     SYSAOS Replicator Management                    ↩
                          ?     Help                                            ↩
                          .     Exit                                            ↩
                        ----    --------------------------------                ↩
                 Code ...  _                                                    ↩
                                                                                ↩
                                                                                ↩
Command ==>                                                                     ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                   Menu
```

2 Select option **V** from the Adabas Event Replicator Subsystem Administration menu.

The Global Values screens appear.

```
17:03:12      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2017-10-23
                               Global Values                          M-RP1110 ↩
                                                                                ↩
                                                                                ↩
---------- General Values -----------   ----------- General Values -----------
Subtasks ................. ___3         EntireX Broker Stub Name .. BKIMBTSO  ↩
                                                                                ↩
Max Output Size .......... ____100000   Verify Mode ............... N         ↩
                                                                                ↩
Input Request Msg Limit .. _________10   Format Buffer Validation .. N         ↩
                                                                                ↩
Input Request Msg Intrvl . _________60   Record PLOG information ... N         ↩
                                                                                ↩
Max Record Size .......... _____32767   Log Input Transaction ..... 080       ↩
                                                                                ↩
Max Variable Record Size . _____32767   Subtask Activation Wait ... __10      ↩
                                                                                ↩
Retry Interval............ __________0   Open at Start ............. Y         ↩
                                                                                ↩
Retry Count............... _________10   Num Parallel Adabas Calls . _3        ↩
                                                                                ↩
SLOGCHECKINTERVAL ........ __________0   Queue Full Delay .......... _60       ↩
                                                                                ↩
 ASSOTHRESHOLD ........... _80          STATINTERVAL .............. __________0
 DATATHRESHOLD ........... _80                                                  ↩
                                                                                ↩
 SLOGACTHRESHOLD ......... _70                                                  ↩
                                                                                ↩
 SLOGDSTHRESHOLD ......... _70                                                  ↩
                                                                                ↩
 SLOGNITHRESHOLD ......... _70                                                  ↩
                                                                                ↩
 SLOGUITHRESHOLD ......... _70                                                  ↩
                                                                                ↩
                                                                                ↩
Command ==>                                                                     ↩
                                                                                ↩
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save              +                        Menu
```

```
17:03:22      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2017-10-23
                                Global Values                          M-RP1111 ↩
                                                                                ↩
                                                                                ↩
-------- TLOG Values ------------                                              ↩
                                                                                ↩
Maximum RPL Usage ........ _50 %                                                ↩
                                                                                ↩
Restart RPL Usage ......... 40 %                                                ↩
                                                                                ↩
Input Queue Level ............ 0                                               ↩
                                                                                ↩
No Match Level ............... 0                                               ↩
                                                                                ↩
Queue Completion Level ....... 0                                                ↩
                                                                                ↩
Completion Level ............. 0                                               ↩
                                                                                ↩
Request Received Level ....... 0                                                ↩
                                                                                ↩
Request Rejected Level ....... 0                                                ↩
                                                                                ↩
Request Error Level .......... 0                                               ↩
                                                                                ↩
Status Request Level ......... 0                                               ↩
                                                                                ↩
I-State Start Request Level .. 0                                               ↩
                                                                                ↩
I-State Completion Level ..... 0                                               ↩
                                                                                ↩
Retransmit Request Level ..... 0                                               ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
Command ==>                                                                     ↩
                                                                                ↩
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save        -                              Menu
```

3 Update the global parameters on this screen as described in the following table.

Parameter Name	Specify	Default
Completion Level	The level of transaction logging that should occur when a transaction has been fully processed and Adabas has been informed that the transaction was successfully replicated. Valid values are "0" (no logging) or "1" (log event and input transaction data). This is the equivalent of specifying the TLCOMP parameter directly in the Event Replicator Server startup job.	0
EntireX Broker Stub Name	The default EntireX Broker stub name to be used by the Event Replicator Server if no other name is specified. This is the same as specifying the ETBBROKERNAME parameter directly in the Event Replicator Server startup job. EntireX Broker is a component of webMethods EntireX.	BROKER
Format Buffer Validation	Indicates the level of format buffer validation that should occur for subscriptions. Valid values are a blank, "A", "D", "N", or "W". ■ If a blank or "N" are specified, format buffer validation is not performed at Event Replicator initialization, at the initial handshake of databases, or when an updated FDT is received. Format buffer validation still occurs during the subscription phase of transaction processing, with validation errors written to the URBRRSP field of the URBR record. ■ If "A" is specified, format buffer validation is performed. If validation errors are identified at Event Replicator Server initialization, the Event Replicator Server is terminated. If validation errors are identified after Event Replicator initialization, warning messages are issued for each format buffer in error. ■ If "D" is specified, format buffer validation is performed. If validation errors are identified, the subscription for which format buffer validation failed is deactivated and warning messages are issued for each format buffer in error. ■ If "W" is specified, format buffer validation is performed. If validation errors are identified, warning messages are issued for each format buffer in error. This is the same as specifying the FBVALIDATION parameter directly in the Event Replicator Server startup job.	N
I-State Completion Level	The level of transaction logging that should occur when an initial-state information request has completed. Valid values are "0" (no logging), "1" (log event and URBS), "2" (log event and URBI, if available), or "3" (log event, URBS, and URBI, if available). This is the equivalent of specifying the TLISTATECOMP parameter in the Event Replicator Server startup job.	0
I-State Start Request Level	The level of transaction logging that should occur when a user request for initial-state information for a file has started. Valid values are "0" (no logging), "1" (log event and URBS), "2" (log event and URBI), or "3" (log event, URBS, and URBI). This is the	0

Parameter Name	Specify	Default
	equivalent of specifying the TLISTATE parameter in the Event Replicator Server startup job.	
Input Queue Level	The level of transaction logging that should occur when a transaction is taken off the input queue and put on the transaction assignment queue. Valid values are "0" (no logging), "1" (log event and input transaction data), "2" (log event, input transaction, and file/ record information), or "3" (log event and all available input information for the event). This is the equivalent of specifying the TLINPUT parameter in the Event Replicator Server startup job.	0
Input Request Msg Interval	The interval during which the message limit specified by the **Input Request Message Limit** field applies. This is the equivalent of specifying the IRMSGINTERVAL parameter directly in the Event Replicator Server startup job.	60
Input Request Msg Limit	The maximum number of input request error messages issued by the Event Replicator Server during the interval set by the **Input Request Message Interval** field (also on this screen). This is the equivalent of specifying the IRMSGLIMIT parameter directly in the Event Replicator Server startup job.	10
Log Input Transaction	Whether or not the Event Replicator should use its SLOG system file as a temporary storage location for incoming compressed replication transactions, before they are queued for processing. Once transactions have been written to the SLOG system file, the Event Replicator Server processes them using a throttling mechanism so that only a limited amount of Event Replicator Server replication pool space is used at a time. Valid values are ALL (indicating that input transactions will always will be written to the SLOG system file), NO (the default, indicating that input transaction will not be written to the SLOG system file, or an integer in the range from 1 to 99. The integer setting specifies a threshold percentage of the LRPL (Event Replicator Server replication pool space) that can be used before triggering the writing of input transaction to the SLOG system file. This is the equivalent of specifying the LOGINPUTTRANSACTION parameter directly in the Event Replicator Server startup job.	NO
Max Output Size	The maximum output message size for the Event Replicator for Adabas. This is the equivalent of specifying the MAXOUTPUTSIZE parameter directly in the Event Replicator Server startup job. The minimum value you can specify for this field is 32768. This parameter may be specified in bytes or it may be specified with the suffix K to indicate kilobytes. The maximum value is 2,147,483,647 bytes. The practical maximum is limited by the region size of the Event Replicator Server. One output buffer is acquired for each output task.	100,000

Parameter Name	Specify	Default
Max Record Size	The maximum length (in bytes) of any decompressed record that can be processed by the Event Replicator Server. This is the equivalent of specifying the MAXRECORDSIZE parameter directly in the startup job. The minimum value you can specify for this field is 1. For an Event Replicator Server running with Adabas 7, the maximum value that can be specified for MAXRECORDSIZE is 32,767 bytes. However, in Adabas 8 systems, this limit has been lifted; the size of a decompressed record may be much larger than 32,767 bytes. Therefore, for an Event Replicator Server running with Adabas 8, the maximum value that can be specified for MAXRECORDSIZE is the larger of either 32,767 bytes or 50% of the setting of the ADARUN LRPL parameter.	32,767 bytes
Max Variable Record Size	The maximum length (in bytes) of variable decompressed records that can be processed by the Event Replicator Server. This is the equivalent of specifying the MAXVARRECORDSIZE parameter directly in the startup job. The value specified for MAXVARRECORDSIZE must be less than or equal to the setting of the MAXRECORDSIZE setting. The minimum value you can specify for this field is 1. For an Event Replicator Server running with Adabas 7, the maximum value that can be specified for MAXRECORDSIZE or MAXVARRECORDSIZE is 32,767 bytes. However, in Adabas 8 systems, this limit has been lifted; the size of a decompressed record may be much larger than 32,767 bytes. Therefore, for an Event Replicator Server running with Adabas 8, the maximum value that can be specified for MAXRECORDSIZE or MAXVARRECORDSIZE is the larger of either 32,767 bytes or 50% of the setting of the ADARUN LRPL parameter.	32,767 bytes
Maximum RPL Usage	The maximum percentage of the Event Replicator Server replication pool that can be used for transaction log (TLOG) processing. Valid values range from "0" through "100". This is the equivalent of specifying the TLMAX parameter in the Event Replicator Server startup job.	50
No Match Level	The level of transaction logging that should occur when a transaction is not queued to any subscription in the Event Replicator Server. Valid values are "0" (no logging), "1" (log event and input transaction data), "2" (log event, input transaction, and file/record information), or "3" (log event and all available input information for the event). This is the equivalent of specifying the TLNOSUB parameter in the Event Replicator Server startup job.	0
Num Parallel Adabas Calls	The maximum number of parallel Adabas calls that can be made. Valid values are integers in the range from 1 through 50. Use	3

Parameter Name	Specify	Default
	this parameter to improve throughput of data replication to your Adabas destinations. This is the equivalent of specifying the NPADACALLS parameter in the Event Replicator Server startup job.	
Open at Start	Indicates whether or not destinations with **Open at Start** (DOPEN) set to "G" or input queues with **Open Queue at Start** (IQOPEN) set to "GLOBAL" in their definitions should be opened at Event Replicator Server startup. This parameter specifies the global policy for determining whether destinations with **Open at Start** set to "G" or input queues with **Open Queue at Start** set to "GLOBAL" are opened at Event Replicator Server startup. Valid values are "Y" or "N", with "Y" as the default. When this parameter is set to "Y", any destinations with **Open at Start** set to "G" or input queues with **Open Queue at Start** set to "GLOBAL" are opened at Event Replicator Server startup. When this parameter is set to "N", any destinations with **Open at Start** set to "G" or input queues with **Open Queue at Start** set to "GLOBAL" are *not* opened at Event Replicator Server startup. This is the equivalent of specifying the GOPEN parameter in the Event Replicator Server startup job.	Y
Queue Completion Level	The level of transaction logging that should occur prior to a transaction being assigned to the completion queue. Valid values are "0" (no logging), "1" (log event and input transaction data, "2" (log event, input transaction, and file/record information), or "3" (log event and all input information available for the event). This is the same as specifying the TLQCOMP parameter directly in the Event Replicator Server startup job.	0
Queue Full Delay	The number of seconds between retry attempts when resending output transactions to a previously-full webMethods EntireX or MQSeries destination. Valid values are integers in the range from "5" through "300". This is the same as specifying the GQFULLDELAY parameter directly in the Event Replicator Server startup job.	60
Record PLOG Information	Indicates whether or not PLOG information is saved in the Replicator system file. Valid values are "Y" (store the information) or "N" (do not store the information). This is the same as specifying the RECORDPLOGINFO parameter directly in the Event Replicator Server startup job. If you plan on using the automated replay facility provided with Event Replicator, you must set this parameter to "Y". The related information is collected when the source Adabas nucleus processes the SYNP checkpoint created by ADARES PLCOPY.	N

Parameter Name	Specify	Default
	Note: If this parameter is set to "Y", the PLOG information screens in the Adabas Event Replicator Subsystem will either not display any PLOG data set information or will display outdated information on the screens. For more information about the PLOG information screens, read *Reviewing and Managing the PLOG Data Set List*, elsewhere in this guide.	
Request Error Level	The level of transaction logging that should occur when a user request is rejected due to an error in carrying out the request. Valid values are "0" (no logging), "1" (log event and URBS), "2" (log event and URBI), or "3" (log event, URBS, and URBI). This is the equivalent of specifying the TLREQERR parameter in the Event Replicator Server startup job.	0
Request Received Level	The level of transaction logging that should occur when a user request has been received. Valid values are "0" (no logging), "1" (log event but no data), or "2" (log event and the entire input buffer before and after translation if appropriate). This is the equivalent of specifying the TLREQRECV parameter in the Event Replicator Server startup job.	0
Request Rejected Level	The level of transaction logging that should occur when a user request is rejected due to an error in interpreting the request. Valid values are "0" (no logging), "1" (log event but no data), or "2" (log event, error code, and entire input buffer). This is the equivalent of specifying the TLREQREJECT parameter in the Event Replicator Server startup job.	0
Restart RPL Usage	The amount of available Event Replicator Server replication pool storage that must be available before transaction logging (TLOG logging) can restart. Valid values range from "0" through "99". This is the equivalent of specifying the TLRESTART parameter in the Event Replicator Server startup job.	40
Retransmit Request Level	The level of transaction logging that should occur when a user request to retransmit a specific transaction has been processed. Valid values are "0" (no logging), "1" (log event and URBS), "2" (log event and URBI), or "3" (log event, URBS, and URBI). This is the equivalent of specifying the TLRETRANS parameter in the Event Replicator Server startup job.	0
Retry Count	The default number of times that an attempt to open a destination or input queue will be retried at the interval specified by the Retry Interval parameter. This is the equivalent of specifying the RETRYCOUNT parameter in the Event Replicator Server startup job. Valid values range from 0 through 2,147,483,647. A value of zero indicates that no retry processing should occur for any affected destinations.	10
Retry Interval	The default number of seconds between retry attempts that will be performed for any destination on input queue for which no	0

Parameter Name	Specify	Default
	specific retry interval has been specified. This is the equivalent of specifying the RETRYINTERVAL parameter in the Event Replicator Server startup job. Valid values range from 0 through 2,147,483,647. A value of zero indicates that no retry processing should occur for any affected destinations.	
SLOGCHECKINTERVAL	The interval (in minutes) between checks of the Associator and Data file space usage. The SLOGCHECKINTERVAL is an optional parameter. If it is not specified, a value of zero (0) is assumed, indicating that no threshold monitoring should occur. If it is specified, valid values are zero (0) or any numbers in the range from 5 to 2880 minutes (48 hours). Once a value is specified, the settings of the thresholds specified by the ASSOTHRESHOLD, DATATHRESHOLD, SLOGACTHRESHOLD, SLOGNITHRESHOLD, SLOGUITHRESHOLD, and SLOGDSTHRESHOLD initialization parameters are used to evaluate your system's Adabas and SLOG Associator and Data file space usage. If the thresholds are met or reached, warning messages are issued to the console and DDPRINT	0
ASSOTHRESHOLD	The percentage of the Event Replicator database Associator space that can be used before threshold warning messages are issued. Associator space usage must meet or exceed this threshold value to trigger the warning messages. Valid values are 0-100. The ASSOTHRESHOLD is an optional parameter. If the percentage of space used meets or exceeds the setting of this parameter, warning messages are issued to the console. If zero (0) is specified for this parameter, no threshold monitoring will occur for the Event Replicator database Associator space. **Note:** If the SLOGCHECKINTERVAL initialization parameter is not specified, no Event Replicator database Associator space threshold monitoring will occur, regardless of the setting of this parameter.	80
DATATHRESHOLD	The percentage of the Event Replicator database Data space that can be used before threshold warning messages are issued. Data space usage must meet or exceed this threshold value to trigger the warning messages. Valid values are 0-100. The DATATHRESHOLD is an optional parameter. If the percentage of space used meets or exceeds the setting of this parameter, warning messages are issued to the console. If zero (0) is specified for this parameter, no threshold monitoring will occur for the Event Replicator database Data space.	80

Parameter Name	Specify	Default
	Note: If the SLOGCHECKINTERVAL initialization parameter is not specified, no Event Replicator database Data space threshold monitoring will occur, regardless of the setting of this parameter.	
SLOGACTHRESHOLD	The percentage of the SLOG address converter space that can be used before threshold warning messages are issued. Address converter space usage must meet or exceed this threshold value to trigger the warning messages. Valid values are 0-100. The SLOGACTHRESHOLD is an optional parameter. If the percentage of space used meets or exceeds the setting of this parameter, warning messages are issued to the console. To determine if the address converter threshold has been reached, the percentage used is calculated using the following formula: `(TOPISN-MINISN+1)/(MAXISN-MINISN+1)` If zero (0) is specified for this parameter, no threshold monitoring will occur for the SLOG address converter space. **Note:** If the SLOGCHECKINTERVAL initialization parameter is not specified, no SLOG address converter space threshold monitoring will occur, regardless of the setting of this parameter.	70
SLOGDSTHRESHOLD	The percentage of the SLOG Data Storage space that can be used before threshold warning messages are issued. Data Storage space usage must meet or exceed this threshold value to trigger the warning messages. Valid values are 0-100. The SLOGDSTHRESHOLD is an optional parameter. If the percentage of space used meets or exceeds the setting of this parameter, warning messages are issued to the console. To determine if the Data Storage threshold has been reached, the percentage used is calculated using the following formula, where acyl represents the number of allocated cylinders, uncyl represents the number of unused cylinders, and unbytes represents the number of unused bytes (per the DSST): `(acyl - uncyl - unbytes) / acyl` If zero (0) is specified for this parameter, no threshold monitoring will occur for the SLOG Data space. **Note:** If the SLOGCHECKINTERVAL initialization parameter is not specified, no SLOG Data Storage space threshold monitoring will occur, regardless of the setting of this parameter.	70
SLOGNITHRESHOLD	The percentage of the SLOG normal index space that can be used before threshold warning messages are issued. Normal index	70

Parameter Name	Specify	Default
	space usage must meet or exceed this threshold value to trigger the warning messages. Valid values are 0-100. The SLOGNITHRESHOLD is an optional parameter. If the percentage of space used meets or exceeds the setting of this parameter, warning messages are issued to the console. To determine if the normal index threshold has been reached, the percentage used is calculated using the following formula, where acyl represents the number of allocated cylinders and ucyl represents the number of used cylinders: `(ucyl / acyl)` If zero (0) is specified for this parameter, no threshold monitoring will occur for the SLOG normal index space. **Note:** If the SLOGCHECKINTERVAL initialization parameter is not specified, no SLOG normal index space threshold monitoring will occur, regardless of the setting of this parameter.	
SLOGUITHRESHOLD	The percentage of the SLOG upper index space that can be used before threshold warning messages are issued. Upper index space usage must meet or exceed this threshold value to trigger the warning messages. Valid values are 0-100. The SLOGUITHRESHOLD is an optional parameter. If the percentage of space used meets or exceeds the setting of this parameter, warning messages are issued to the console. To determine if the upper index threshold has been reached, the percentage used is calculated using the following formula, where acyl represents the number of allocated cylinders and ucyl represents the number of used cylinders: `(ucyl / acyl)` If zero (0) is specified for this parameter, no threshold monitoring will occur for the SLOG upper index space. **Note:** If the SLOGCHECKINTERVAL initialization parameter is not specified, no SLOG upper index space threshold monitoring will occur, regardless of the setting of this parameter.	70
STATINTERVAL	The interval (in seconds) at which Event Replicator statistics should be published by the Event Replicator Server. Valid values are 0, or any numbers in the range of 60 to 86400 (24 hours). The STATINTERVAL initialization parameter can be used to activate Replication Monitoring. Once set to a non zero value, interval statistics will be generated by the Event Replicator	0

Parameter Name	Specify	Default
	Server. See *Replication Monitoring* for more details. A value of zero (0) indicates that interval statistics should not be published.	
Status Request Level	The level of transaction logging that should occur when a user request for status on an Event Replicator Server resource has been processed. Valid values are "0" (no logging), "1" (log event and URBS), "2" (log event and URBI), or "3" (log event, URBS, and URBI). This is the equivalent of specifying the `TLSTATUS` parameter in the Event Replicator Server startup job.	0
Subtask Activation Wait	The number of seconds that can be used to override the default time to wait for a subtask to finish initialization and activate. Valid values are from 1 to 3600 seconds. This is the equivalent of specifying the `SUBTASKWAIT` parameter in the Event Replicator Server startup job.	10
Subtasks	The number of subtasks in the Event Replicator Server. This is the equivalent of specifying the `SUBTASKS` parameter in the Event Replicator Server startup job.	3
Verify Mode	Whether the Event Replicator for Adabas should run in verify (test) mode or not. Valid values are "Y" (run in verify mode) or "N" (do not run in verify mode). This is the equivalent of specifying the `VERIFYMODE` parameter in the Event Replicator Server startup job.	N

4 Press PF5 to save your settings.

SYSAOS Replicator Management

You can directly access SYSAOS Replicator Management, using the Replicator DBID as the Database ID.

To access SYSAOS Replicator Management:

1 Select option **A** from the Adabas Event Replicator Subsystem Main Menu.

The Administration menu appears.

```
19:30:31     ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2017-03-15
                                Administration                        M-RP1100


                       Code    Function
                       ----    -------------------------------
                         D     Database ID
                         I     Perform Initial-State
                         P     PLOG Information
                         R     Initiate Replay
                         S     System Functions
                         T     Target Adapter
                         V     Global Values
                         X     SYSAOS Replicator Management
                         ?     Help
                         .     Exit
                       ----    -------------------------------
               Code ...  _


Command ==>



Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                  Menu
```

2 Select option **X** from the Adabas Event Replicator Subsystem Administration menu.

The Replicator Management screen appears.

```
19:52:15          ***** A D A B A S  BASIC  SERVICES *****          2017-03-15
Replicator 1954          - Replicator Management -                   PRPT002   ↩
                                                                                ↩
                                                                                ↩
                       Code   Service                                           ↩
                       ----   -------------------------                        ↩
                        A     Activate/deact/open/close                        ↩
                        D     Display Reptor definitions                       ↩
                        F     Display Reptor statistics                        ↩
                        H     Perform RPLCheck                                 ↩
                        L     Perform RPLCleanup                               ↩
                        P     Perform RPLRefresh                               ↩
                        R     Parameter subsystem                              ↩
                        ?     Help                                             ↩
                        .     Exit                                             ↩
                       ----   -------------------------                        ↩
                                                                                ↩
          Code ......... _                                                      ↩
          Database ID .. 1954  (REP1954)                                        ↩
                                                                                ↩
                                                                                ↩
Command ==>                                                                     ↩
PF1----- PF2------ PF3------ PF4------ PF6----- PF7----- PF8----- PF12-----     ↩
Help     File Serv Exit                                          Menu
```

3 View or update the parameters on this screen as described in section *Managing Replication from AOS,* in *Event Replicator for Adabas Administration and Operations Guide*.

4 If you enter option **R** in this screen, you will be taken to the Parameter subsystem main menu of SYSRPTR.

Getting Help

Online help is provided for every Adabas Event Replicator Subsystem screen and message that appears in the Adabas Event Replicator Subsystem. This section covers the following topics:

Getting Screen-Level Help

To get screen-level help for any Adabas Event Replicator Subsystem screen:

1 Be sure the screen you want help for is displayed. Navigate to it if you need to.

2 Press the PF1 key.

Help for the screen displays.

Getting Help for Messages

To get help for Adabas Event Replicator Subsystem messages:

- At the Command prompt, enter:

```
msg nnn
```

where *nn* is the valid two- or three-digit message number (omit the leading zeros). For example, to get help on message ARF00013, you would enter:

```
msg 13
```

Using Function Keys

The following table describes the function keys available while using the Adabas Event Replicator Subsystem screens. Note that not all function keys are available on all screens and some PF keys have meanings only to specific screens (these PF keys are not described here, but are described with the screen to which they apply).

Function Key	Display Title	Description
PF1	Help	Provides help on the current screen.
PF2	Repos	Displays a pop-up screen allowing you to specify the definition name to which you want a list of definitions repositioned. This is useful if you have many definitions listed on one of the Adabas Event Replicator Subsystem list screens.
PF3	Exit	Exits the current screen without saving any changes you might have made. If the current screen is the Main Menu, this function key has no effect.
PF4	Add	Displays a screen that allows you to add a definition.
PF5	Save or Exec	Saves the changes you have made or allows you to execute the program corresponding to the Adabas Event Replicator Subsystem screen displayed.
PF7	-	Scrolls backwards through the data on a screen.
PF8	+	Scrolls forward through the data on a screen.
PF12	Menu	Leaves the Adabas Event Replicator Subsystem screens.

Leaving the Adabas Event Replicator Subsystem Screens

To leave the Adabas Event Replicator Subsystem screens, press PF12 from any screen. If you are on a menu screen in the Adabas Event Replicator Subsystem, you can select the dot (.) option to leave.

Note: When you leave the Adabas Event Replicator Subsystem screens, any modifications to the subsystem since the last save will not be stored.

3 Maintaining Initial-State Definitions

An initial-state definition defines an initial-state request for data from the target application. Initial-state definitions identify the subscription, destination, and specific Adabas files to use in an Event Replicator for Adabas run; at least one subscription, destination, or file must be specified.

Initial-state data can contain any subset of the data on the Adabas database, based on the specifications in the initial-state definition and parameters supplied in the initial-state request. Records can be selected for initial-state processing in one of the following manners:

- The complete file can be selected.
- Records are selected from the file based on an ISN list.
- Records are selected from the file based on specified selection criteria.

Note: Each replicated initial-state record contains the related data storage after image. No before image is replicated for an initial-state record.

You can populate a database with initial-state data using the Adabas Event Replicator Subsystem. For more information, read ***Populating a Database With Initial-State Data***, elsewhere in this guide.

In addition, you can populate a database with initial-state data from a client application. For more information, read *Event Replicator Client Requests*, in *Event Replicator for Adabas Application Programmer's Reference Guide*.

Listing Initial-State Definitions

To use the Adabas Event Replicator Subsystem to list the initial-state definitions stored in the Replicator system file:

- Select option **I** from the Adabas Event Replicator Subsystem Main Menu.

  The List of Initial-State Definitions screen appears showing all of the initial-state definitions in the Adabas Event Replicator Subsystem.

```
15:12:32      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                      List of Initial-State Definitions              M-RP1500

 Sel   Name          Sel   Name          Sel   Name          Sel   Name
 ---------------     ---------------     ---------------     -------------
  _  I040155           _                   _                   _
  _  I042060           _                   _                   _
  _  I042200           _                   _                   _
  _  I046088           _                   _                   _
  _  I062026           _                   _                   _
  _  I062029           _                   _                   _
  _  I062035           _                   _                   _
  _  I062055           _                   _                   _
  _  I062079           _                   _                   _
  _  I062143           _                   _                   _
  _  I064121           _                   _                   _
  _  I120248           _                   _                   _
  _  I215168           _                   _                   _
  _                    _                   _                   _

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Repos Exit  Add                -     +                     Menu  ↩
 ↩
```

The function keys on this screen perform the following functions:

Function Key	Description
PF1/F1 (Help)	Provides you with help for this screen.
PF2/F2 (Repos)	Provides you with a pop-up panel that allows you to specify the name of the definition you want to locate in the list. Once you have specified a name on the pop-up panel and pressed Enter, the list is repositioned so the name you selected appears first. You can use an asterisk as a wild card character at the end of the definition name or partial definition name you specify on the pop-up panel. Or, you can simply enter the first few characters of the name to reposition the list to the first occurrence in the list of a name starting with those characters.
PF3/F3 (Exit)	Returns you to the previous screen.
PF4/F4 (Add)	Allows you to add a new definition. A new screen appears.
PF7/F7 (-)	Allows you to scroll backwards through the list of definitions.
PF8/F8 (+)	Allows you to scroll forwards through the list of definitions.
PF12/F12 (Menu)	Returns you to the main menu.

Adding Initial-State Definitions

This section describes the steps you must complete to use the Adabas Event Replicator Subsystem to add an initial-state definition to the Replicator system file:

Step 1. Access the Initial-State Definition Area of the Adabas Event Replicator Subsystem

To access the initial-state definition area of the Adabas Event Replicator Subsystem:

1 Select option **I** from the Adabas Event Replicator Subsystem Main Menu.

The List of Initial-State Definitions screen appears.

2 Press PF4 on the List of Initial-State Definitions.

The Initial-State Definition screen appears.

```
15:13:07      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                          Initial-State Definition                   M-RP1510



       Initial-State Name ............................. ________
          Initial-State Destinations ................... _
          Initial-State Subscriptions .................. _
          Num of Concurrent Initial-State Requests ..... _________1

Sel IDBID IFile I  Selection Criteria
--------------------------------------------------------------------------------
 _  _____ _____  _ ____________________________________________________________
 _  _____ _____  _ ____________________________________________________________
 _  _____ _____  _ ____________________________________________________________
 _  _____ _____  _ ____________________________________________________________
 _  _____ _____  _ ____________________________________________________________



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save        -     +                     Menu  ↩
 ↩
```

Step 2. Supply a Name for the Initial-State Definition

On the **Initial-State Definition** screen, tab to the **Initial-State Name** field and specify a unique name for the initial-state definition. The name must use alphanumeric characters and be between one and 8 characters long.

Step 3. Specify the Number of Concurrent Initial-State Requests Possible for the Initial-State Definition

On the **Initial-State Definition** screen, tab to the **Num of Concurrent Initial State Requests** field and specify the number of concurrent initial-state requests that can occur in a given instance of this initial-state definition. Valid values range from 1 to 2,147,483,647. The default is 1. This is the same as specifying the `IMAXREQ` parameter directly in the Event Replicator Server startup job.

Step 4. Select Destinations, Subscriptions, and Files for the Initial-State Definition

At least one subscription, destination, or file must be specified for an initial-state definition; all three may be specified, but at least one *must* be.

When one or more DBID/file combinations are specified in an initial-state definition, and one or more destinations or subscriptions are also defined, the initial-state data for the specified DBID/files are sent only to the specified destinations or as described by the specified subscriptions. In other words, delivery of the initial-state data in the files is restricted by the destination and subscription definitions. If no DBID/file combinations are specified in the initial-state definition, the specified destination and subscription definitions are used to construct a list of related DBID/files that should be used for the run.

This step describes how to:

- Select Destinations for the Initial-State Definition
- Select Subscriptions for the Initial-State Definition
- Select Files for the Initial-State Definition

Select Destinations for the Initial-State Definition

To select destination definitions for the initial-state definition:

The destination definitions must be previously defined.

1 On the **Initial-State Definition** screen, enter an S in the **Initial-State Destinations** field.

The Initial-State Destination List screen appears allowing you to select destinations for the replicated data that will be generated by this initial-state request.

```
15:14:23      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                        Initial-State Destination List                M-RP1520
                                     TEST

 Name             Name             Name             Name             Name
-------------    -------------    -------------    -------------    -------------
_________        _________        _________        _________        _________
_________        _________        _________        _________        _________
_________        _________        _________        _________        _________
_________        _________        _________        _________        _________
_________        _________        _________        _________        _________
_________        _________        _________        _________        _________
_________        _________        _________        _________        _________
_________        _________        _________        _________        _________
_________        _________        _________        _________        _________
_________        _________        _________        _________        _________
_________        _________        _________        _________        _________



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save  Sel                                 Menu  ↩
 ↩
```

2 Tab to the **Name** field and specify the name of a destination definition to use for this initial-state definition. The destination definition must be previously defined.

Or:

Press the PF6 key to bring up a list of destination definitions already defined. Using this list you can select definitions for use by the initial-state definition. Simply place an S next to the destinations you want to use in the list. When all destinations are selected, press the PF5 key to accept your selections and close the list. The selected destinations will appear on the Initial-State Destination List screen.

For more information about creating destination definitions, read *Maintaining Destination Definitions*, elsewhere in this guide.

3 Repeat Steps 1-2 until all destinations for this initial-state definition have been specified. Use PF7 and PF8 to scroll through the destinations you have added.

4 Press PF5 to save the destination list. Then press PF3 to return to the Initial-State Definition screen.

Select Subscriptions for the Initial-State Definition

To select subscription definitions for the initial-state definition:

The subscription definitions must be previously defined.

1 On the **Initial-State Definition** screen, enter an S in the **Initial-State Subscriptions** field.

The Initial-State Subscription List appears allowing you to select subscriptions for the replicated data that will be generated by this initial-state request.

```
15:15:29      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                       Initial-State Subscription List                M-RP1530
                                    TEST

 Name             Name             Name             Name             Name
--------------   --------------   --------------   --------------   --------------
________         ________         ________         ________         ________
________         ________         ________         ________         ________
________         ________         ________         ________         ________
________         ________         ________         ________         ________
________         ________         ________         ________         ________
________         ________         ________         ________         ________
________         ________         ________         ________         ________
________         ________         ________         ________         ________
________         ________         ________         ________         ________
________         ________         ________         ________         ________
________         ________         ________         ________         ________



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save  Sel                                 Menu  ↩
 ↩
```

2 Tab to the **Name** field and specify the name of the subscription definition to use for this initial-state definition. The subscription must be previously defined.

Or:

Press the PF6 key to bring up a list of subscription definitions already defined. Using this list you can select definitions for use by the initial-state definition. Simply place an S next to the subscriptions you want to use in the list. When all subscriptions are selected, press the PF5 key to accept your selections and close the list. The selected subscriptions will appear on the Initial-State Subscription List screen.

For more information about creating subscription definitions, read *Adding Subscription Definitions*, elsewhere in this guide.

3 Repeat Step 2 until all subscriptions for this initial-state definition have been specified. Use PF7 and PF8 to scroll through the subscriptions you have added.

4 Press PF5 to save the subscription list. Then press PF3 to return to the Initial-State Definition screen.

Select Files for the Initial-State Definition

To select files for the initial-state definition:

1 On the **Initial-State Definition** screen, tab to the **IDBID** field and specify the database ID associated with an input file you want used for this initial-state definition.

The database ID is numeric and can range from one to 65535. There is no default.

2 Remaining on the same line, tab to the **IFile** field and specify the file number of an input file you want used for this initial-state definition. There is no default.

3 Tab to the **I** field and specify the filter method that should be used to process data from this input file. Valid values are:

Value	Description
A	No filter method is used. All records from the input file are processed and replicated.
I	Only records with ISNs that match those specified in the initial-state request will be replicated. If you elect to use an ISN list, be sure to supply an ISN list when you submit the initial-state request. The ISN list can be specified using the Adabas Event Replicator Subsystem (read ***Populating a Database With Initial-State Data***, elsewhere in this guide) or in the client request (read *ISN List Format* in the section entitled *Initial-State Requests* in *Event Replicator for Adabas Application Programmer's Reference Guide*).
<blank>	If this field is left blank, the records are filtered based on Adabas search criteria you specify in the **Selection Criteria** field. Only records that meet the criteria are processed and replicated.

4 If you left the **I** field blank in the previous step, use the **Selection Criteria** field to specify any Adabas search criteria you want to use to select input data for processing. Only the records in the input file that satisfy the search criteria will be processed. A maximum of a 60-byte search buffer is provided. The search criteria you specify should be in the same format used to specify an Adabas S1 command. For more information about the S1 command, read the Adabas command reference documentation found in *Adabas Command Reference*.

5 Repeat Steps 1 through 4 until all the file specifications you want have been added to the initial-state definition. Use the PF7 and PF8 to scroll through the input database/file specifications for this initial-state definition.

Use the **Sel** column to delete or copy file specifications in the initial-state definition. To copy a file specification, type a "C" in the **Sel** column. To delete a file specification, type a "D" in the **Sel** column.

6 Press PF5 to save the initial-state definition.

Step 5. Save the Initial-State Definition

Press PF5 to save the initial-state definition.

Modifying Initial-State Definitions

To use the Adabas Event Replicator Subsystem to modify an initial-state definition in the Replicator system file:

1 List the initial-state definitions in the Adabas Event Replicator Subsystem, as described in *Listing Initial-State Definitions*, elsewhere in this guide.

The initial state definitions are listed on the List of Initial-State Definitions screen.

2 Locate the definition you want to modify on the screen and enter an **M** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

The Initial-State Definition screen appears with the settings for the initial-state definition you selected. For information on modifying this screen, read the description of *Adding Initial-State Definitions*, elsewhere in this section.

3 When all modifications have been made, press PF5 to save the changes.

Copying Initial-State Definitions

To use the Adabas Event Replicator Subsystem to copy an initial-state definition in the Replicator system file:

1 List the initial-state definitions in the Adabas Event Replicator Subsystem, as described in *Listing Initial-State Definitions*, elsewhere in this guide.

The initial state definitions are listed on the List of Initial-State Definitions screen.

2 Locate the definition you want to copy on the screen and enter a **C** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

A dialog appears requesting a name for the copy of the initial-state definition.

```
Enter new name: ________
or press PF3 to cancel
                              ↩
```

3 Specify a new, unique name for the copy of the initial-state definition and press Enter.

The initial-state definition is copied and the copy appears on the List of Initial-State Definitions screen.

Deleting Initial-State Definitions

To use the Adabas Event Replicator Subsystem to delete an initial-state definition in the Replicator system file:

1 List the initial-state definitions in the Adabas Event Replicator Subsystem, as described in *Listing Initial-State Definitions*, elsewhere in this guide.

The initial state definitions are listed on the List of Initial-State Definitions screen.

2 Locate the definition you want to delete on the screen and enter a **D** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

The initial-state definition is deleted.

4 Maintaining Destination Definitions

A destination definition defines the destination of the replicated data. Destination definitions can be created for Adabas, webMethods EntireX (or, EntireX Broker), WebSphere MQ, File, and Null destinations. At least one definition is required for every Event Replicator for Adabas destination you intend to use.

Destination Type	Description
Adabas	Data is replicated to one or more Adabas files.
EntireX Broker	Replicated data is written to an output queue via webMethods EntireX Broker.
WebSphere MQ	Replicated data is written to an output queue via IBM WebSphere MQ.
Null	Data replication is tested without actually sending the data to a destination.
File	Replicated data is written to the CLOG, using TLOG URBLTDOD records.

Listing Destination Definitions

To use the Adabas Event Replicator Subsystem to list the destination definitions stored in the Replicator system file:

- Select option **D** from the Adabas Event Replicator Subsystem Main Menu.

  The List of Destinations screen appears showing all of the destination definitions in the Adabas Event Replicator Subsystem.

```
15:33:16      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                           List of Destinations                      M-RP1200

 Sel   Name   Typ   Sel   Name   Typ   Sel   Name   Typ   Sel   Name   Typ
 ----------------  ----------------  ----------------  ----------------
  _  ADA1     ADA    _  52200202 ADA    _  52201616 ADA    _  52203111 ADA
  _  ADA2     ADA    _  52200303 ADA    _  52201717 ADA    _  52203212 ADA
  _  D314315  NULL   _  52200404 ADA    _  52201818 ADA    _  52203313 ADA
  _  D314315A ADA    _  52200505 ADA    _  52201919 ADA    _  52203414 ADA
  _  D4       ADA    _  52200606 ADA    _  52202101 ADA    _  52203515 ADA
  _  EXXDEST  ETB    _  52200707 ADA    _  52202202 ADA    _  52203616 ADA
  _  FILE1    FILE   _  52200808 ADA    _  52202303 ADA    _  52203717 ADA
  _  IS1-DEST ADA    _  52200909 ADA    _  52202404 ADA    _  52203818 ADA
  _  MYDEST   ADA    _  52201010 ADA    _  52202505 ADA    _  52203919 ADA
  _  NULLX1   NULL   _  52201111 ADA    _  52202606 ADA    _  52204101 ADA
  _  NULLX2   NULL   _  52201212 ADA    _  52202707 ADA    _  52204202 ADA
  _  XYZ123   ADA    _  52201313 ADA    _  52202808 ADA    _  52204303 ADA
  _  52005019 ADA    _  52201414 ADA    _  52202909 ADA    _  52204404 ADA
  _  52200101 ADA    _  52201515 ADA    _  52203010 ADA    _  52204505 ADA

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Repos Exit  Add               -     +                     Menu  ↩
 ↩
```

The function keys on this screen perform the following functions:

Function Key	Description
PF1/F1 (Help)	Provides you with help for this screen.
PF2/F2 (Repos)	Provides you with a pop-up panel that allows you to specify the name of the definition you want to locate in the list. Once you have specified a name on the pop-up panel and pressed Enter, the list is repositioned so the name you selected appears first. You can use an asterisk as a wild card character at the end of the definition name or partial definition name you specify on the pop-up panel. Or, you can simply enter the first few characters of the name to reposition the list to the first occurrence in the list of a name starting with those characters.
PF3/F3 (Exit)	Returns you to the previous screen.
PF4/F4 (Add)	Allows you to add a new definition. A new screen appears.
PF7/F7 (-)	Allows you to scroll backwards through the list of definitions.
PF8/F8 (+)	Allows you to scroll forwards through the list of definitions.
PF12/F12 (Menu)	Returns you to the main menu.

Creating an Adabas Destination Definition

Using Adabas destination definitions, data can be replicated to one or more Adabas files. This section describes how to create an Adabas destination definition using the Adabas Event Replicator Subsystem.

Note: An Adabas destination can be referenced by no more than one subscription.

To create an Adabas destination definition in the Adabas Event Replicator Subsystem, complete the following steps:

- Step 1. Access the Adabas Destination Definition Creation Area
- Step 2. Specify General and TLOG Adabas Destination Parameters
- Step 3. Specify Input and Target Adabas Destination Databases and Files
- Step 4. (Optional) Specify File-Related Parameters for the Adabas Destination
- Step 5. Save the Adabas Destination Definition

Step 1. Access the Adabas Destination Definition Creation Area

To access the Adabas destination definition creation area of the Adabas Event Replicator Subsystem:

1 Select option **D** from the Adabas Event Replicator Subsystem Main Menu.

The List of Destinations screen appears.

2 Press PF4 on the List of Destinations screen.

The Create New Destination screen appears.

```
15:34:11      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                          Create New Destination                    M-RP1290


                        Code    Function
                        ----    -------------------------
                          A     Create Adabas Destination
                          E     Create Broker Destination
                          F     Create File Destination
                          N     Create Null Destination
                          M     Create MQ Destination
                          ?     Help
                          .     Exit
                         ----   -------------------------

                 Code ...  _




Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu  ↩
 ↩
```

3 Select option **A** on the Create New Destination screen.

The Adabas Destination Definition screen appears.

```
15:35:03      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                        Adabas Destination Definition                M-RP1235


 Destination Name .... ________                Input   Input   Target  Target
 Destination Active .. Y                 Sel   DBID    File     DBID    File
 Allow Logging ....... N                 ---  ------  ------   ------  ------
 Retry Interval ...... GLOBAL____         _   ______  ____     ______  ____
 Retry Count ......... GLOBAL____         _   ______  ____     ______  ____
 Error Action ........ ALTACTION          _   ______  ____     ______  ____
 Open at Startup ..... G                  _   ______  ____     ______  ____
                                          _   ______  ____     ______  ____
 TLOG Parameters                          _   ______  ____     ______  ____
 ----------------------                   _   ______  ____     ______  ____
 Assign Level ........ 0                  _   ______  ____     ______  ____
 Completion Level .... 0                  _   ______  ____     ______  ____
 SLOG Write Level .... 0                  _   ______  ____     ______  ____
 SLOG Read Level ..... 0
 Adabas Level ........ 0

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save         -     +                     Menu  ↩
 ↩
```

Step 2. Specify General and TLOG Adabas Destination Parameters

To use the Adabas Event Replicator Subsystem to supply general and TLOG specifications for an Adabas destination definition, complete the following steps:

1 Update the following general fields on the Adabas Destination Definition screen as described in the following table.

Parameter Name	Specify	Default
Destination Name	The unique name for the Adabas destination definition. This is the equivalent of specifying the `DESTINATION NAME` parameter directly in the Event Replicator Server startup job. The specified name must be alphanumeric and be between one and eight characters long.	---
Destination Active	Whether or not this destination definition should be activated for use once it is loaded by the Event Replicator Server. Valid values are "Y" (load and activate the definition) or "N" (load, but do not activate the definition). This is the equivalent of specifying the `DACTIVE` parameter in the Event Replicator Server startup job.	Y
Allow Logging	Whether or not subscription logging should be activated for this destination definition. Valid values are "Y" (activate subscription logging) or "N" (do	N

Parameter Name	Specify	Default
	not activate subscription logging). This is the equivalent of specifying the DLOG parameter in the Event Replicator Server startup job.	
Retry Interval	The default number of seconds between retry attempts to open the destination. This is the equivalent of specifying the DRETRYINTERVAL parameter directly in the Event Replicator Server startup job. Valid values are 0, 5 through 2,147,483,647, or the literal "GLOBAL". If the value "GLOBAL" is specified for this parameter, the specification for the **Retry Interval global variable** will be used. A value of zero indicates that no retry attempt to open this destination should occur. Except for a specification of zero, the minimum value that can be specified for this parameter is 5 seconds.	The value of the **Retry Interval global variable**.
Retry Count	The number of times that an attempt to open the destination will be retried at the interval specified by the Retry Interval parameter. This is the equivalent of specifying the DRETRYCOUNT parameter directly in the Event Replicator Server startup job. Valid values range from 0 through 2,147,483,647 or the literal "GLOBAL". If the value "GLOBAL" is specified for this parameter, the specification for the **Retry Count global variable** will be used. Any retry attempts will occur at the interval specified by the Retry Interval parameter. A value of zero indicates that no retry attempt to open this destination should occur.	The value of the **Retry Count global variable**
Error Action	The action to be taken when an error occurs during replication to an Adabas destination. This is the equivalent of specifying the DAERROR parameter directly in the Event Replicator Server startup job. Valid values are ALTACTION, BACKOUT, or CLOSE. In all cases (ALTACTION, BACKOUT, and CLOSE), if response 148 is returned and the SLOG system file is available, the destination is closed. If an insert, update, or delete operation fails because a replicated record already exists or does not exist, an appropriate message is issued. If the transaction fails because of an error, a message containing the two-character Adabas command, the database ID, the file number, the response code and the subcode is written. If the DATMETHOD parameter is set to ISN, the text "ISN" will be appended to this message as well as the ISN value. If the DATMETHOD parameter is set to "KEY", the record key will be written in both hexadecimal and readable formats in a separate message. Additional actions are taken, based on the value of this parameter. These actions are: ■ ALTACTION: Processing continues with the next update that is part of the same transaction. Some special processing occurs when DAERROR=ALTACTION:	ALTACTION

Parameter Name	Specify	Default
	■ If an insert is processed and the record already exists, the record is updated. ■ If an update is processed and the record does not exist, the record is inserted. ■ If a delete is processed and the record does not exist, processing continues with the next record. For other errors, the record is skipped. ■ BACKOUT: A message is issued indicating that the transaction will be backed out and then ignored. The current transaction is backed out and processing continues with the next transaction to be replicated. ■ CLOSE: A message is issued indicating that the transaction will be backed out and the destination will be closed. The current transaction is backed out and transaction logging (to the SLOG file) will begin, if defined for the destination.	
Open at Startup	Whether or not the destination should be opened at Event Replicator Server startup. Valid values are "Y", "N", or "G", with "G" as the default. When this parameter is set to "Y", the destination is opened at Event Replicator Server startup. When this parameter is set to "N", the destination is *not* opened at Event Replicator Server startup. When this parameter is set to "G", the decision to open the destination at Event Replicator Server startup depends on the setting of the **Open Destinations at start** (GOPEN) global parameter. If GOPEN=YES, the destination is opened at Event Replicator Server startup; if GOPEN=NO, it is not opened. This is the equivalent of specifying the `DOPEN` parameter in the Event Replicator Server startup job.	G

2 Update the following TLOG fields on the Adabas Destination Definition screen as described in the following table.

Parameter Name	Specify	Default
Assign Level	The level of transaction logging that should occur when a transaction is assigned to a destination for output processing. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the `DTLASSIGN` parameter in the Event Replicator Server startup job.	0
Completion Level	The level of transaction logging that should occur when a transaction has been successfully output to the messaging system. Valid values are "0" (no logging),	0

Parameter Name	Specify	Default
	"1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the DTLCOMP parameter in the Event Replicator Server startup job.	
SLOG Write Level	The level of transaction logging that should occur when a transaction has been successfully written to the SLOG file. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the DTLSLOGWRITE parameter in the Event Replicator Server startup job.	0
SLOG Read Level	The level of transaction logging that should occur when a transaction has been successfully read from the SLOG and is about to be queued for output to the destination. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the DTLSLOGREAD parameter in the Event Replicator Server startup job.	0
Adabas Level	The level of transaction logging that should occur when a transaction for an Adabas destination incurred an error. This is the equivalent of specifying the DTLADABAS parameter in the Event Replicator Server startup job. Valid values range from 0 through 3, as described below: ■ 0: No transaction logging should occur. ■ 1: Log event and Adabas error information. ■ 2: Log event, Adabas error information, and file and record data. ■ 3: Log event, Adabas error information, file and record data, and the actual payload.	0

Step 3. Specify Input and Target Adabas Destination Databases and Files

To specify the input and target Adabas destination databases and file:

- Update the following fields on the Adabas Destination Definition screen as described in the following table.

Parameter Name	Specify	Default
Input DBID	The database ID associated with an input file (see the Input File field) for this Adabas destination. This is the equivalent of specifying the DAIDBID parameter directly in the Event Replicator Server startup job. The database ID is numeric and can range from one to 65535. The input database ID and file listed in this destination definition must also be included in an SFILE definition in the subscription. Multiple input database IDs can be specified in a single destination definition, as needed. Only unique combinations of Input DBID and Input File parameters can be specified in a single Adabas destination definition. This implies that the input from a database and file combination can only have a single Adabas target within a specific Adabas destination definition. If you want the input from a database and file combination to go to more than one Adabas target, define multiple Adabas destination definitions.	---
Input File	The number of the input file for this Adabas destination definition. This is the equivalent of specifying the DAIFILE parameter directly in the Event Replicator Server startup job. The input database ID and file listed in this destination definition must also be included in an SFILE definition in the subscription. At least one file must be listed for an Adabas destination definition. Multiple input files can be specified in a single destination definition, as needed. Only unique combinations of Input DBID and Input File parameters can be specified in a single Adabas destination definition. This implies that the input from a database and file combination can only have a single Adabas target within a specific Adabas destination definition. If you want the input from a database and file combination to go to more than one Adabas target, define multiple Adabas destination definitions.	---
Target DBID	The database ID associated with the target file for the replicated data. This is the equivalent of specifying the DATDBID parameter directly in the Event Replicator Server startup job. The database ID is numeric and can range from one to 65535. Multiple target database IDs and files can be specified in a single destination definition, as needed.	---
Target File	The number of the target (output) file for the replicated data associated with the input file on the same line in this destination definition. There is no default. This is the equivalent of specifying the DATFILE parameter directly in the Event Replicator Server startup job. Multiple target database IDs and files can be specified in a single destination definition, as needed.	---

Note: The Sel field on this screen is used to modify parameters for replication specific to a given database and file combination. Skip to the instructions for *Step 3. (Optional) Specify File-Related Parameters for the Adabas Destination* in this series of steps for information about the fields on the **File-Related Parameters** screen.

Step 4. (Optional) Specify File-Related Parameters for the Adabas Destination

To specify file-related parameters for the Adabas destination:

1 On the **Adabas Destination Definition** screen, enter "e", "m", or "s" in the **Sel** field corresponding to the Adabas file for which you want to specify file-related parameters.

The **File-Related Parameters** screen appears.

```
15:37:29      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                        Adabas Destination Definition                M-RP1237
                           File-Related Parameters

 Input               Input                   Target            Target
 DBID ....    42     File ...    2           DBID ...    63    File ...    3


Replicate Utility .......... N
Replication Method ......... ISN
After Image Offset ......... _________0
After Image Key Length ..... ____0
Before Image Offset ........ _________0
Before Image Key Length .... ____0
Key Offset ................. ____0
Key Length ................. ____0
Search Buffer.. ____________________________________________________________



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu  ↩
 ↩
```

2 Update the fields on this screen as described in the following table.

Note: This `Input DBID` and `Input File` fields display the database ID and file number of the input file you selected on the previous Adabas destination screen. In addition, the `Target DBID` and `Target File` display the database ID and file number of the target file associated with the input file you selected on the previous Adabas destination screen.

Parameter Name	Description	Default
After Image Key Length	When the Replication Method parameter is set to "KEY", this parameter is required. If the Replication Method parameter is set to ISN, this parameter may not be specified. Specify the length (in bytes) of the key to be used in the after image of the data buffer. Valid values range from 1 through 32,767 bytes. This is the equivalent of specifying the `DATKEYAIL` parameter directly in the Event Replicator Server startup job.	none
After Image Offset	This optional parameter can only be set if the Replication Method parameter is set to "KEY"; if the Replication Method parameter is set to ISN, this parameter may not be specified. Specify the offset of the key to be used in the after image of the data buffer. Valid values range from 0 through 2,147,483,646 bytes. The sum of this parameter value and the length of the key to be used in the after image (After Image Key Length parameter), must be less than or equal to 2,147,483,647 bytes. This is the equivalent of specifying the `DATKEYAIO` parameter directly in the Event Replicator Server startup job.	0
Before Image Key Length	This optional parameter can only be set if the Replication Method parameter is set to "KEY"; if the Replication Method parameter is set to ISN, this parameter may not be specified. If the Before Image Offset parameter is specified, this parameter must also be specified with a value greater than zero. Specify the length (in bytes) of the key to be used in the before image of the data buffer. Valid values range from 1 through 32767 bytes. This is the equivalent of specifying the `DATKEYBIL` parameter directly in the Event Replicator Server startup job.	none
Before Image Offset	This optional parameter can only be set if the Replication Method parameter is set to "KEY"; if the Replication Method parameter is set to ISN, this parameter may not be specified. Specify the offset of the key to be used in the before image of the data buffer. Valid values range from 0 through 2,147,483,646 bytes. The sum of this parameter value and the length of the key to be used in the before image (Before Image Key Length parameter), must be less than or equal to 2,147,483,647 bytes. If this parameter is specified, the Before Image Key Length parameter must also be specified with a value greater than zero. This is the equivalent of specifying the `DATKEYBIO` parameter directly in the Event Replicator Server startup job.	0
Key Length	This optional parameter can only be set if the Replication Method parameter is set to "KEY"; if the Replication Method parameter is set to ISN, this parameter	none

Parameter Name	Description	Default
	may not be specified. If the Key Offset parameter is specified, this parameter must also be specified with a value greater than zero. Specify the length (in bytes) of the key to be used in the before image of the primary key. Valid values range from 1 through 32767 bytes. This is the equivalent of specifying the DATKEYKYL parameter directly in the Event Replicator Server startup job.	
Key Offset	This optional parameter can only be set if the Replication Method parameter is set to "KEY"; if the Replication Method parameter is set to ISN, this parameter may not be specified. Specify the offset of the key to be used in the before image of the primary key. Valid values range from 0 through 32767 bytes. The sum of this parameter value and the length of the key to be used in the before image of the primary key (Key Length parameter), must be less than or equal to 32767 bytes. If this parameter is specified, the Key Length parameter must also be specified with a value greater than zero. This is the equivalent of specifying the DATKEYKYO parameter directly in the Event Replicator Server startup job.	0
Replicate Utility	This parameter can be specified regardless of the Replication Method selected. Specify whether Adabas utility change replication should be activated for a specific target file at Event Replicator Server startup. This is the equivalent of specifying the DAREPLICATEUTI parameter directly in the Event Replicator Server startup job. Valid values are "Y" and "N". If "Y" is specified, utility replication is activated for the target file at Event Replicator Server startup; if "N" is specified, utility replication is not activated for the target file. For more information about replicating utility functions, read *Replicating Utility Functions*, in *Event Replicator for Adabas Concepts*.	N
Replication Method	Specify the method to be used when searching for a record on the target database. Valid values are "ISN" and "KEY". This is the equivalent of specifying the DATMETHOD parameter directly in the Event Replicator Server startup job. When this parameter is set to "KEY", the parameters Search Buffer, After Image Offset, After Image Key Length, Before Image Offset, Before Image Key Length, Key Offset, and Key Length may also be supplied to indicate where to find the key in the replicated data. All of these parameters are optional except Search Buffer and After Image Length, which are required.	ISN

Parameter Name	Description	Default
Search Buffer	When the Replication Method parameter is set to "KEY", this parameter is required. If the Replication Method parameter is set to ISN, this parameter may not be specified. Specify a search buffer to be used for keyed replication. Up to 60 alphanumeric characters can be specified. This is the equivalent of specifying the DATKEYSB parameter directly in the Event Replicator Server startup job.	none

3 When all file-related parameters have been specified for the file, press PF3 to return to the first Adabas Destination Definition screen.

Step 5. Save the Adabas Destination Definition

To save the Adabas destination definition in the Replicator system file:

- Press PF5 on the **Adabas Destination Definition** screen.

  The Adabas destination definition is saved in the Replicator system file.

Creating an EntireX Broker Destination Definition

Using an EntireX Broker destination definition, replicated data is written to an output queue via webMethods EntireX. Be sure to read *Using webMethods EntireX as the Messaging System*, in *Event Replicator for Adabas Administration and Operations Guide*, prior to using webMethods EntireX as the messaging subsystem.

To create an EntireX Broker destination definition in the Adabas Event Replicator Subsystem, complete the following steps:

- Step 1. Access the EntireX Broker Destination Definition Creation Area
- Step 2. Specify General and TLOG EntireX Broker Destination Parameters
- Step 3. (Optional) Specify Destination Class Information, If Applicable

Step 1. Access the EntireX Broker Destination Definition Creation Area

To access the EntireX Broker destination definition creation area of the Adabas Event Replicator Subsystem:

1 Select option **D** from the Adabas Event Replicator Subsystem Main Menu.

The List of Destinations screen appears.

2 Press PF4 on the List of Destinations screen.

The Create New Destination screen appears.

```
15:34:11      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                          Create New Destination                    M-RP1290


                       Code    Function
                       ----    -------------------------
                         A     Create Adabas Destination
                         E     Create Broker Destination
                         F     Create File Destination
                         N     Create Null Destination
                         M     Create MQ Destination
                         ?     Help
                         .     Exit
                        ----   -------------------------

                 Code ...  _




Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu  ↩
 ↩
```

3 Select option **E** on the Create New Destination screen.

The EntireX Broker Destination Definition screen appears.

```
13:43:20      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2017-10-23
                                EntireX Broker                        M-RP1220 ↩

                             Destination Definition                              ↩

                                                                                 ↩

Dest Name ..... CMAP____  EntireX Broker ..... UHO2135:SVC229:NET_______________
Architecture ........ _2  Broker Service ..... SEXX____________________________
Threshold ... _________5  Service Name ....... ETBSERV_________________________
Destination Active ... Y  Service Class ...... ARF2____________________________
Allow Logging ........ N   Retry Interval ..... GLOBAL____                     ↩

Replicate Util Chgs .. N  Retry Count ........ GLOBAL____                     ↩

Event Logging ........ N  Queue Full Delay ... GLOBAL____                     ↩

Open at Startup ...... G  Single Conversation mode .. NO_
Max Out Size. __________  EntireX Token ...... _______________________________
DSTATLOG ............. N  EntireX Userid ..... _______________________________
TLOG Parms                Heartbeat Interval . _________0                     ↩

------------------------              Destination Class ..... _________
Assign Level ......... 0               Destination Class Parameter Data ..... ↩

Completion Level ..... 0             ___________________________________________
SLOG Write Level ..... 0             ___________________________________________
SLOG Read Level ...... 0             ___________________________________________
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save                                      Menu
```

Step 2. Specify General and TLOG EntireX Broker Destination Parameters

To use the Adabas Event Replicator Subsystem to supply general and TLOG specifications for an EntireX Broker destination definition, complete the following steps:

1 Update the following general fields on the EntireX Broker Destination Definition screen as described in the following table.

Parameter Name	Specify	Default
Allow Logging	Whether or not subscription logging should be activated for this destination definition. Valid values are "Y" (activate subscription logging) or "N" (do not activate subscription logging). This is the equivalent of specifying the `DLOG` parameter in the Event Replicator Server startup job.	N

Parameter Name	Specify	Default
Architecture	The data architecture for fields in the URB* control structures sent to the EntireX Broker destination. For complete information on calculating a value for this parameter, read `DARC`, in *Event Replicator for Adabas Reference Guide*..	2
Broker Service	The EntireX Broker service for which this destination definition applies. The service identification can be up to 32 characters long. This is the equivalent of specifying the `DETBSERVICE` parameter directly in the Event Replicator Server startup job.	---
Dest Name	The unique name for the EntireX Broker destination definition. This is the equivalent of specifying the `DESTINATION NAME` parameter directly in the Event Replicator Server startup job. The specified name must be alphanumeric and be between one and eight characters long.	---
Destination Active	Whether or not this destination definition should be activated for use once it is loaded by the Event Replicator Server. Valid values are "Y" (load and activate the definition) or "N" (load, but do not activate the definition). This is the equivalent of specifying the `DACTIVE` parameter in the Event Replicator Server startup job.	Y
Destination Class	Leave this field blank for now. It is described later in this section.	---
Destination Class Parameter Data	Leave this field blank for now. It is described later in this section.	---
DSTATLOG	Decides whether or not the associated destination should receive interval statistics published by the Event Replicator Server. This parameter is used with *Replication Monitoring*. Valid values are Y or N (YES or NO). A value of Y indicates that interval statistics should be published to the destination; a value of N indicates they should not. A destination defined with DSTATLOG=Y cannot be specified in a subscription.	N

Parameter Name	Specify	Default

EntireX Broker ID	The EntireX Broker ID for which this destination definition applies. The name can be up to 32 characters long.
	
	Broker IDs come in two formats: one for TCP/IP communications and one for Adabas SVC communications. For TCP/IP communications, the format is:
	
	`addr:port-number:TCP`
	
	In this case, the *addr* setting is either the TCP/IP IP address or the host name. The *port-number* setting should match the EntireX Broker PORT parameter.
	
	For Adabas SVC communications, the format is:
	
	`'broker-id:SVCnnn:NET'`
	
	In this case, the *broker-id* setting should match the EntireX Broker BROKER-ID parameter in the Broker ETBFILE DD. The *nnn* setting should match either the EntireX Broker ADASVC or ADA5SVC parameters in the Broker PARMS DD statement.
	
	If no name is specified, the default EntireX Broker ID specified by the ETBBROKERID parameter is used. This is the equivalent of specifying the DETBBROKERID in the Event Replicator Server startup job.
		--- 	

EntireX Token	This is the equivalent of the DETBTOKEN subparameter for the DESTINATION parameter, defined in Event Replicator Initialization parameters. It specifies the webMethods EntireX token. This optional parameter can only be specified when DETBSCONV=YES. Refer to *TOKEN* in the *EntireX Broker ACI for Assembler* section of the *webMethods EntireX* documentation for more information on usage and syntax.	REPTOR-DEST-*destination name*
EntireX Userid	This is the equivalent of the DETBUSERID subparameter subparameter for the DESTINATION parameter, defined in Event Replicator Initialization parameters. It specifies the webMethods user ID. This optional parameter can only be specified when DETBSCONV=YES.	REPTOR--*node*-----*jobname*

Parameter Name	Specify	Default
	Refer to *TOKEN* in the *EntireX Broker ACI for Assembler* section of the *webMethods EntireX* documentation for more information on usage and syntax.	
Event Logging	Whether or not events should be logged by the Event Replicator Server and sent to this destination. This is the equivalent of specifying the `DEVENTLOG` parameter directly in the Event Replicator Server startup job. Valid values are "Y" or "N". When this optional parameter is set to "Y", Event Replicator Server events are logged to the destination. When this parameter is set to "N" (the default), they are not. Event Replicator Server events are logged in URBS elements. These URBS elements are sent to destinations related to the event itself. The URBS elements are also sent to any other destinations you have defined "Event Logging =Y". If a related destination also is defined with "Event Logging =Y", it will only receive one instance of the URBS element. To access this log of Event Replicator Server events in the destination queue, you must supply your own application that reads the event URBS elements in the destination queue. If such an application does not exist, the logged events simply sit in the queue.	N
Heartbeat Interval	This is the equivalent of the `DHBINTERVAL` subparameter for the `DESTINATION` parameter, defined in Event Replicator Initialization parameters. It specifies the interval (in seconds) at which the heartbeat should be published to a destination. Valid values are 0 or 60 to 86400 (24 hours). Use this subparameter to set the interval at which heartbeat events are sent to a particular destination. Information for a heartbeat event is collected if at least Heartbeat Interval (DHBINTERVAL) seconds have passed since a heartbeat event was last sent to the destination. This heartbeat information is inserted into the replication data stream at the exit from the Event Replicator Server (destination processing). A value of zero (0) indicates that a heartbeat event should not be published for a particular destination. The Heartbeat Interval (DHBINTERVAL) subparameter is only allowed for destinations defined with	0

Parameter Name	Specify	Default
	DTYPE=ETBROKER, DTYPE=MQSERIES, or DTYPE=NULL. The DHBINTERVAL parameter is only allowed for destinations defined with DCLASS=SAGTARG. **Note:** The interval between creating heartbeat events may be slightly longer (e.g. 10 seconds longer) than the interval specified for this parameter.	
Max Output Size	The maximum output size (in bytes) for the destination. This is the equivalent of specify the DMAXOUTPUTSIZE parameter directly in the Event Replicator Server startup job. Valid values are 0 or any integer ranging from 4096 through 2,147,483,647. You can specify the value for this parameter in a purely numeric form or use K at the end of the number to specify kilobytes. For example, DMAXOUTPUTSIZE=4K is the same as DMAXOUTPUTSIZE=4096. The value for this parameter will be used if it is less than or equal to the maximum output size for the Event Replicator Server (specified using the MAXOUTPUTSIZE global parameter) and less than or equal to the maximum output allowed for the messaging system queue being defined. If this value is larger than the MAXOUTPUTSIZE specification or the maximum output size allowed by the messaging system, the smaller value will be used. A value of 0 indicates that no specific limit is set for this destination. Instead, the smaller of the MAXOUTPUTSIZE specification or the maximum output size allowed by the messaging system will be used.	0
Open at Startup	Whether or not the destination should be opened at Event Replicator Server startup. Valid values are "Y", "N", or "G", with "G" as the default. When this parameter is set to "Y", the destination is opened at Event Replicator Server startup. When this parameter is set to "N", the destination is *not* opened at Event Replicator Server startup. When this parameter is set to "G", the decision to open the destination at Event Replicator Server startup depends on the setting of the **Open Destinations at start** (GOPEN) global parameter. If GOPEN=YES, the	G

Parameter Name	Specify	Default
	destination is opened at Event Replicator Server startup; if GOPEN=NO, it is not opened. This is the equivalent of specifying the DOPEN parameter in the Event Replicator Server startup job.	
Queue Full Delay	The number of seconds between retry attempts when resending output transactions to a specific and previously-full webMethods EntireX destination. Valid values are integers in the range from "5" through "300" or the word "GLOBAL". If the value "GLOBAL" is specified, the number seconds between retry attempts is set to the value of the GQFULLDELAY initialization parameter. This is the equivalent of specifying the DQFULLDELAY parameter directly in the Event Replicator Server startup job.	GLOBAL
Replicate Util Chgs	Whether Adabas utility change replication should be activated for a destination at Event Replicator Server startup. This is the equivalent of specify the DREPLICATEUTI parameter directly in the Event Replicator Server startup job. Valid values are "Y" and "N". If "Y" is specified, utility replication is activated for the destination at Event Replicator Server startup; if "N" is specified, utility replication is not activated for the destination. For more information about replicating utility functions, read *Replicating Utility Functions*, in *Event Replicator for Adabas Concepts*.	N
Retry Count	The number of times that an attempt to open the destination will be retried at the interval specified by the Retry Interval parameter. This is the equivalent of specifying the DRETRYCOUNT parameter directly in the Event Replicator Server startup job. Valid values range from 0 through 2,147,483,647 or the literal "GLOBAL". If the value "GLOBAL" is specified for this parameter, the specification for the **Retry Count global variable** will be used. Any retry attempts will occur at the interval specified by the Retry Interval parameter. A value of zero indicates that no retry attempt to open this destination should occur.	The value of the **Retry Count global variable**

Parameter Name	Specify	Default
Retry Interval	The default number of seconds between retry attempts to open the destination. This is the equivalent of specifying the DRETRYINTERVAL parameter directly in the Event Replicator Server startup job. Valid values are 0, 5 through 2,147,483,647, or the literal "GLOBAL". If the value "GLOBAL" is specified for this parameter, the specification for the **Retry Interval global variable** will be used. A value of zero indicates that no retry attempt to open this destination should occur. Except for a specification of zero, the minimum value that can be specified for this parameter is 5 seconds.	The value of the **Retry Interval global variable**.
Service Class	The EntireX Broker service class name for which this destination definition applies. The name can be up to 32 characters long. This is the equivalent of specifying the DETBSERVICECLASS parameter in the Event Replicator Server startup job.	---
Service Name	The EntireX Broker service name for which the EntireX Broker destination definition applies. This is the equivalent of specifying the DETBSERVICENAME parameter directly in the Event Replicator Server startup job. The name can be up to 32 characters long.	---
Single Conversation Mode	This is the equivalent of the DETBSCONV subparameter for the DESTINATION parameter, defined in Event Replicator Initialization parameters. It specifies whether or not the destination is to run in single conversation mode for Node to Node support. Valid values are YES or NO. This parameter should be used for high volume destinations. When this optional parameter is set to YES for a destination, all messages are sent using the same EntireX Broker conversation ID. In the event of a failure, or if the destination is closed, the same conversation ID will be used the next time the destination connection to the EntireX Broker is opened. If there is no active conversation, then a new conversation ID will be established.	N

Parameter Name	Specify	Default
Threshold	The number of messages that will be sent to the EntireX Broker destination before a commit is performed for those messages. For complete information on specifying a value for this parameter, read *DCOMMITTHRESHOLD*, in *Event Replicator for Adabas Reference Guide*.	5

2 Update the following TLOG fields on the EntireX Broker Destination Definition screen as described in the following table.

Parameter Name	Specify	Default
Assign Level	The level of transaction logging that should occur when a transaction is assigned to a destination for output processing. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the `DTLASSIGN` parameter in the Event Replicator Server startup job.	0
Completion Level	The level of transaction logging that should occur when a transaction has been successfully output to the messaging system. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the `DTLCOMP` parameter in the Event Replicator Server startup job.	0
SLOG Write Level	The level of transaction logging that should occur when a transaction has been successfully written to the SLOG file. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the `DTLSLOGWRITE` parameter in the Event Replicator Server startup job.	0
SLOG Read Level	The level of transaction logging that should occur when a transaction has been successfully read from the SLOG and is about to be queued for output to the destination. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the `DTLSLOGREAD` parameter in the Event Replicator Server startup job.	0

Step 3. (Optional) Specify Destination Class Information, If Applicable

The Destination Class and Destination Class Parameter Data fields can be used to invoke and pass parameters to the Event Replicator Target Adapter for this destination. The fields are located at the bottom of the EntireX Broker destination screen. Do not use these fields unless you want to invoke and pass parameters to the Event Replicator Target Adapter for the destination or unless otherwise requested by a Software AG support representative.

If applicable, use the Destination Class (`DCLASS`) field to specify the destination class for this destination definition. Valid values are blank or "SAGTARG" (if Event Replicator Target Adapter processing should be invoked). There is no default.

Note: DCLASS=SAGTARG cannot be specified for destinations with DTYPE=ADABAS or FILE. It is only valid for webMethods EntireX, WebSphere MQ, or NULL destinations. When DCLASS=SAGTARG is specified, the ADARUN RPLPARMS parameter must be set to "FILE" or "BOTH" to provide access to any field table (GFFT) definitions.

If you specify a value for the Destination Class field, you can optionally use the Destination Class Parameter Data (`DCLASSPARM`) field to specify up to 120 bytes of character data to be passed to the optional destination output user exit.

If DCLASS=SAGTARG is specified (if the Destination Class field is set to "SAGTARG") to invoke the Event Replicator Target Adapter for a destination, you may want to specify one or more of the following keyword parameters:

Note: These parameter keywords must be specified in uppercase.

NOSPRE

Specify the "NOSPRE" keyword in the DCLASSPARM parameter if you do not want the subscription name to prefix the names of the tables produced by the Event Replicator Target Adapter. When "NOSPRE" is specified, the schema file name (Predict view name) alone is used for the table names; when "NOSPRE" is *not* specified, the subscription name prefixes the schema file name in the table names.

Note: Oracle identifiers are limited to 30 characters. If NOSPRE is *not* specified and an Oracle RDBMS is used by the Event Replicator Target Adapter, the identifier names may exceed 30 characters and errors may occur. We recommend using NOSPRE if an Oracle RDBMS is also used.

SPRE

Specify the SPRE keyword in the DCLASSPARM parameter if you do want the subscription name to prefix the names of the tables produced by the Event Replicator Target Adapter.

The notation is `SPRE=`*`xxxxxxxx`*. Where *`xxxxxxxx`* can be a subscription prefix to be used instead of the subscription name used in the job execution. The prefix can be 1 to 8 characters in size.

The SPRE keyword is available for use with SAGTARG exit.

OPTIONS

The OPTIONS keyword parameter can be used to specify options for the destination. Specify the OPTIONS keyword parameter, using the syntax OPTIONS=*nnnn*. Possible values of the OPTIONS (*nnnn*) are listed in the following table. However, if you want to combine options, add their values together and enter the total value.

For example, if you want to combine option 32 (to send the full image on an update) and option 64 (to set the XML transaction committed time value to local time instead of GMT/UTC), specify OPTIONS=96 (with 96 being the sum of 32 and 64).

Option Value	Description
1	This option is no longer supported. If specified, it will be ignored.
2	Specify OPTIONS=2 to indicate that long names should be used. This option will cause long names to be sent in place of the default short names used for various elements and attributes. Short names are the default and save on the amount of data being transferred. Long names make for better readability. For example, the short name <F> would appear as <Field> using long names.
4	Specify OPTIONS=4 to ensure that invalid XML characters found in alphanumeric fields are not translated to spaces.
8	Specify OPTIONS=8 to ensure that trailing blanks in alphanumeric fields are not removed.
16	Specify OPTIONS=16 to ensure that characters used by XML are not replaced automatically with predefined entity references. For example, if OPTIONS=16 is set, the ampersand (&) character would not be replaced with the literal "&".
32	Specify OPTIONS=32 to send the full image on an update and insert. The full before image (if available) and after image of all fields are sent for an update and insert, even if the field values were not changed or are null. **Note:** This option is mutually exclusive with option 128. If both options are specified, then OPTIONS=32 will be set and OPTIONS=128 ignored.
64	Specify OPTIONS=64 to set the XML transaction committed time value to local time instead of GMT/UTC.
128	Specify OPTIONS=128 if the before image is not to be included when a field is updated/changed if the before image data storage records are included. Primary field before images will still be included because the key is needed to find the record that is to be changed. **Note:** This option is mutually exclusive with option 32. If both options are specified, then OPTIONS=32 will be set and OPTIONS=128 ignored.

TRACE

The TRACE keyword parameter can be used to specify the contents of the trace. Specify the TRACE keyword parameter using the syntax TRACE=*nnnn*. Possible values of the TRACE (*nnnn*) are listed in the following table. However, if you want to trace multiple control blocks, add their trace values together and enter the total value. For example, to trace the before and after

images of the URBD control blocks, you would specify `TRACE=24` because the sum of 8 (URBD control block before image) and 16 (URBD control block after image) is 24.

Note: If tracing is enabled using this keyword parameter, be sure to include the following JCL statement in the startup JCL of the Event Replicator Server:

```
//DDTRACE1 DD SYSOUT=X
```

Trace Value	Description
1	Trace the URBS control block.
2	Trace the URBT control block.
4	Trace the URBR control block.
8	Trace the URBD control block before image.
16	Trace the URBD control block after image.
32	Trace the URBF/URBG control block before image.
64	Trace the URBF/URBG control block after image.
128	Trace the send buffer.
256	Trace the URBY control block.
512	Trace the URBO control block.
1024	Trace the output parameters.
2048	Trace the subscription table.

Step 4. Save the EntireX Broker Destination Definition

To save the EntireX Broker destination definition in the Replicator system file:

- Press PF5 on the webMethods EntireX Destination Definition screen.

  The EntireX Broker destination definition is saved in the Replicator system file.

Creating a WebSphere MQ Destination Definition

Using a WebSphere MQ destination definition, replicated data is written to an output queue via IBM WebSphere MQ. Be sure to read *Using WebSphere MQ as the Messaging System*, in *Event Replicator for Adabas Administration and Operations Guide*, prior to using WebSphere MQ as the messaging subsystem.

Note: If you are running on z/OS using IBM WebSphere MQ Series definitions for your Event Replicator DESTINATION or IQUEUE definitions, a S0D3 abend can occur if you run it as a started task and specify the parameter REUSASID=YES. This is a documented IBM WebSphere MQ Series issue.

To create a WebSphere MQ destination definition in the Adabas Event Replicator Subsystem, complete the following steps:

Step 1. Access the WebSphere MQ Destination Definition Creation Area

To access the WebSphere MQ destination definition creation area of the Adabas Event Replicator Subsystem:

1 Select option **D** from the Adabas Event Replicator Subsystem Main Menu.

The List of Destinations screen appears.

2 Press PF4 on the List of Destinations screen.

The Create New Destination screen appears.

```
15:34:11      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                          Create New Destination                    M-RP1290


                       Code    Function
                       ----    -------------------------
                         A     Create Adabas Destination
                         E     Create Broker Destination
                         F     Create File Destination
                         N     Create Null Destination
                         M     Create MQ Destination
                         ?     Help
                         .     Exit
                        ----   -------------------------

                Code ...  _




Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu  ↩
 ↩
```

3 Select option **M** on the Create New Destination screen.

The MQSeries Destination Definition screen appears.

```
17:11:20      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2017-10-23
                     MQSeries Destination Definition                 M-RP1210

Destination Name .... _________  Architecture ... 2_
Queue Manager Name... ____________________________________________________
Queue Name .......... ____________________________________________________
Dynamic Queue Name... ____________________________________________________
Destination Active... Y       Retry Interval ................... GLOBAL____
Allow Logging ....... Y       Retry Count ...................... GLOBAL____
Threshold .. __________5      Max Output Size .................. _______8192
Replicate Util Chgs.. N       Destination Class ................. _________
Event Logging ....... N       MQ Format ......................... _________
Open at Startup ..... G       MQSeries Coded Character Set ID .. ___________
DSTATLOG ............ N       Queue Full Delay ................. GLOBAL____
TLOG Parms                    Heartbeat Interval ............... __________0
----------------------        Destination Class Parameter Data .............
Assign Level ........ 0       ____________________________________________
Completion Level .... 0       ____________________________________________
SLOG Write Level .... 0       _________________________________
SLOG Read Level ..... 0
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save                                    Menu
```

Step 2. Specify General and TLOG WebSphere MQ Destination Parameters

To use the Adabas Event Replicator Subsystem to supply general and TLOG specifications for a WebSphere MQ destination definition, complete the following steps:

1 Update the following general fields on the MQSeries Destination Definition screen as described in the following table.

Parameter Name	Specify	Default
Allow Logging	Whether or not subscription logging should be activated for this destination definition. Valid values are "Y" (activate subscription logging) or "N" (do not activate subscription logging). This is the equivalent of specifying the `DLOG` parameter in the Event Replicator Server startup job.	N
Architecture	The data architecture for fields in the URB* control structures sent to the WebSphere MQ destination. For complete information on calculating a value for this parameter, read *`DARC`*, in *Event Replicator for Adabas Reference Guide*..	2
Destination Active	Whether or not this destination definition should be activated for use once it is loaded by the Event Replicator Server. Valid values are "Y" (load and activate the definition) or "N" (load, but do not activate the definition). This	Y

Parameter Name	Specify	Default
	is the equivalent of specifying the DACTIVE parameter in the Event Replicator Server startup job.	
Destination Class	Leave this field blank for now. It is described later in this section.	---
Destination Class Parameter Data	Leave this field blank for now. It is described later in this section.	---
Destination Name	The unique name for the WebSphere MQ destination definition. This is the equivalent of specifying the DESTINATION NAME parameter directly in the Event Replicator Server startup job. The specified name must be alphanumeric and be between one and eight characters long.	---
DSTATLOG	Decides whether or not the associated destination should receive interval statistics published by the Event Replicator Server. This parameter is used with *Replication Monitoring*. Valid values are Y or N (YES or NO). A value of Y indicates that interval statistics should be published to the destination; a value of N indicates they should not. A destination defined with DSTATLOG=Y cannot be specified in a subscription.	N
Dynamic Queue Name	The WebSphere MQ dynamic queue name. This is the equivalent of specifying the DMQDYNQNAME parameter directly in the Event Replicator Server startup job. The name can be up to 48 characters long.	blanks
Event Logging	Whether or not events should be logged by the Event Replicator Server and sent to this destination. This is the equivalent of specifying the DEVENTLOG parameter directly in the Event Replicator Server startup job. Valid values are "Y" or "N". When this optional parameter is set to "Y", Event Replicator Server events are logged to the destination. When this parameter is set to " N" (the default), they are not. Event Replicator Server events are logged in URBS elements. These URBS elements wre sent to destinations related to the event itself. The URBS elements are also sent to any other destinations you have defined "Event Logging =Y". If a related destination also is defined with "Event Logging =Y", it will only receive one instance of the URBS element. To access this log of Event Replicator Server events in the destination queue, you must supply your own application that reads the event URBS elements in the destination queue. If such an application does not exist, the logged events simply sit in the queue.	N
Heartbeat Interval	This is the equivalent of the DHBINTERVAL subparameter for the DESTINATION parameter, defined in Event Replicator Initialization parameters. It specifies the interval (in seconds) at which the heartbeat should be published to a destination. Valid values are 0 or 60 to 86400 (24 hours).	0

Parameter Name	Specify	Default
	Use this subparameter to set the interval at which heartbeat events are sent to a particular destination. Information for a heartbeat event is collected if at least Heartbeat Interval (DHBINTERVAL) seconds have passed since a heartbeat event was last sent to the destination. This heartbeat information is inserted into the replication data stream at the exit from the Event Replicator Server (destination processing). A value of zero (0) indicates that a heartbeat event should not be published for a particular destination. The Heartbeat Interval (DHBINTERVAL) subparameter is only allowed for destinations defined with DTYPE=ETBROKER, DTYPE=MQSERIES, or DTYPE=NULL. The `DHBINTERVAL` parameter is only allowed for destinations defined with DCLASS=SAGTARG. **Note:** The interval between creating heartbeat events may be slightly longer (e.g. 10 seconds longer) than the interval specified for this parameter.	
Max Output Size	The maximum output size (in bytes) for the destination. This is the equivalent of specify the `DMAXOUTPUTSIZE` parameter directly in the Event Replicator Server startup job. Valid values are 0 or any integer ranging from 4096 through 2,147,483,647. You can specify the value for this parameter in a purely numeric form or use K at the end of the number to specify kilobytes. For example, DMAXOUTPUTSIZE=4K is the same as DMAXOUTPUTSIZE=4096. The value for this parameter will be used if it is less than or equal to the maximum output size for the Event Replicator Server (specified using the `MAXOUTPUTSIZE` global parameter) and less than or equal to the maximum output allowed for the messaging system queue being defined. If this value is larger than the `MAXOUTPUTSIZE` specification or the maximum output size allowed by the messaging system, the smaller value will be used. A value of 0 indicates that no specific limit is set for this destination. Instead, the smaller of the `MAXOUTPUTSIZE` specification or the maximum output size allowed by the messaging system will be used.	0
MQ Format	The optional MQ format name. The format name can be up to eight characters long. This is the equivalent of specifying the `DMQFORMAT` parameter in the Event Replicator Server startup job. **Note:** You cannot specify a value for this parameter if a value has not also been specified for the Destination Class parameter.	blanks
MQSeries Coded	The destination-specific coded character set ID (CCSID) for the WebSphere MQ destination. This is the equivalent of specifying the `DMQCCSID` parameter in the Event Replicator Server startup job. Valid values range from 0 through 2,147,483,647.	0

Parameter Name	Specify	Default
Character Set ID	This optional parameter can only be specified when the DCLASS or DEXIT parameters are specified. The Event Replicator Server does not attempt to verify the value of this parameter as the character codes may be changed or added to as time goes on. The value for this parameter is simply passed in the appropriate WebSphere MQ request as the CCSID.	
Open at Startup	Whether or not the destination should be opened at Event Replicator Server startup. Valid values are "Y", "N", or "G", with "G" as the default. When this parameter is set to "Y", the destination is opened at Event Replicator Server startup. When this parameter is set to "N", the destination is *not* opened at Event Replicator Server startup. When this parameter is set to "G", the decision to open the destination at Event Replicator Server startup depends on the setting of the **Open Destinations at start** (GOPEN) global parameter. If GOPEN=YES, the destination is opened at Event Replicator Server startup; if GOPEN=NO, it is not opened. This is the equivalent of specifying the DOPEN parameter in the Event Replicator Server startup job.	G
Queue Full Delay	The number of seconds between retry attempts when resending output transactions to a specific and previously-full WebSphere MQ destination. Valid values are integers in the range from "5" through "300" or the word "GLOBAL". If the value "GLOBAL" is specified, the number seconds between retry attempts is set to the value of the GQFULLDELAY initialization parameter. This is the equivalent of specifying the DQFULLDELAY parameter directly in the Event Replicator Server startup job.	GLOBAL
Queue Manager Name	The WebSphere MQ queue manager name. The name can be up to 48 characters long. This is the equivalent of specifying the DMQQMGRNAME parameter in the Event Replicator Server startup job.	---
Queue Name	The WebSphere MQ queue name. The name can be up to 48 characters long. There is no default. This is the equivalent of specifying the DMQQNAME parameter in the Event Replicator Server startup job.	---
Replicate Util Chgs	Whether Adabas utility change replication should be activated for a destination at Event Replicator Server startup. This is the equivalent of specify the DREPLICATEUTI parameter directly in the Event Replicator Server startup job. Valid values are "Y" and "N". If "Y" is specified, utility replication is activated for the destination at Event Replicator Server startup; if "N" is specified, utility replication is not activated for the destination.	N

Parameter Name	Specify	Default
	For more information about replicating utility functions, read *Replicating Utility Functions*, in *Event Replicator for Adabas Concepts*.	
Retry Count	The number of times that an attempt to open the destination will be retried at the interval specified by the Retry Interval parameter. This is the equivalent of specifying the DRETRYCOUNT parameter directly in the Event Replicator Server startup job. Valid values range from 0 through 2,147,483,647 or the literal "GLOBAL". If the value "GLOBAL" is specified for this parameter, the specification for the **Retry Count global variable** will be used. Any retry attempts will occur at the interval specified by the Retry Interval parameter. A value of zero indicates that no retry attempt to open this destination should occur.	The value of the **Retry Count global variable**
Retry Interval	The default number of seconds between retry attempts to open the destination. This is the equivalent of specifying the DRETRYINTERVAL parameter directly in the Event Replicator Server startup job. Valid values are 0, 5 through 2,147,483,647, or the literal "GLOBAL". If the value "GLOBAL" is specified for this parameter, the specification for the **Retry Interval global variable** will be used. A value of zero indicates that no retry attempt to open this destination should occur. Except for a specification of zero, the minimum value that can be specified for this parameter is 5 seconds.	The value of the **Retry Interval global variable**
Threshold	The number of messages that will be sent to the WebSphere MQ destination before a commit is performed for those messages. This is the equivalent of specifying the DCOMMITTHRESHOLD parameter directly in the Event Replicator Server startup job. The term "commit" in this context means that the Event Replicator Server informs the messaging system that all messages sent (since the last commit) should be made permanent. In the case of WebSphere MQ, commit means that the Event Replicator Server will issue an MQCMIT call for the queue.	5

2 Update the following TLOG fields on the MQSeries Destination Definition screen as described in the following table.

Parameter Name	Specify	Default
Assign Level	The level of transaction logging that should occur when a transaction is assigned to a destination for output processing. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the DTLASSIGN parameter in the Event Replicator Server startup job.	0

Parameter Name	Specify	Default
Completion Level	The level of transaction logging that should occur when a transaction has been successfully output to the messaging system. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the DTLCOMP parameter in the Event Replicator Server startup job.	0
SLOG Read Level	The level of transaction logging that should occur when a transaction has been successfully read from the SLOG and is about to be queued for output to the destination. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the DTLSLOGREAD parameter in the Event Replicator Server startup job.	0
SLOG Write Level	The level of transaction logging that should occur when a transaction has been successfully written to the SLOG file. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the DTLSLOGWRITE parameter in the Event Replicator Server startup job.	0

Step 3. (Optional) Specify Destination Class Information, If Applicable

The Destination Class and Destination Class Parameter Data fields can be used to invoke and pass parameters to the Event Replicator Target Adapter for this destination. The fields are located at the right of the WebSphere MQ destination screen. Do not use these fields unless you want to invoke and pass parameters to the Event Replicator Target Adapter for the destination or unless otherwise requested by a Software AG support representative.

If applicable, use the Destination Class (DCLASS) field to specify the destination class for this destination definition. Valid values are blank or "SAGTARG" (if Event Replicator Target Adapter processing should be invoked). There is no default.

Note: DCLASS=SAGTARG cannot be specified for destinations with DTYPE=ADABAS or FILE. It is only valid for webMethods EntireX, WebSphere MQ, or NULL destinations. When DCLASS=SAGTARG is specified, the ADARUN RPLPARMS parameter must be set to "FILE" or "BOTH" to provide access to any field table (GFFT) definitions.

If you specify a value for the Destination Class field, you can optionally use the Destination Class Parameter Data (DCLASSPARM) field to specify up to 120 bytes of character data to be passed to the optional destination output user exit.

If DCLASS=SAGTARG is specified (if the Destination Class field is set to "SAGTARG") to invoke the Event Replicator Target Adapter for a destination, you may want to specify one or more of the following keyword parameters:

Note: These parameter keywords must be specified in uppercase.

NOSPRE

Specify the "NOSPRE" keyword in the DCLASSPARM parameter if you do not want the subscription name to prefix the names of the tables produced by the Event Replicator Target Adapter. When "NOSPRE" is specified, the schema file name (Predict view name) alone is used for the table names; when "NOSPRE" is *not* specified, the subscription name prefixes the schema file name in the table names.

Note: Oracle identifiers are limited to 30 characters. If NOSPRE is *not* specified and an Oracle RDBMS is used by the Event Replicator Target Adapter, the identifier names may exceed 30 characters and errors may occur. We recommend using NOSPRE if an Oracle RDBMS is also used.

SPRE

Specify the SPRE keyword in the DCLASSPARM parameter if you do want the subscription name to prefix the names of the tables produced by the Event Replicator Target Adapter.

The notation is `SPRE=`*`xxxxxxxx`*. Where *xxxxxxxx* can be a subscription prefix to be used instead of the subscription name used in the job execution. The prefix can be 1 to 8 characters in size.

The SPRE keyword is available for use with `SAGTARG` exit.

OPTIONS

The OPTIONS keyword parameter can be used to specify options for the destination. Specify the OPTIONS keyword parameter, using the syntax `OPTIONS=`*nnnn*. Possible values of the OPTIONS (*nnnn*) are listed in the following table. However, if you want to combine options, add their values together and enter the total value.

For example, if you want to combine option 32 (to send the full image on an update) and option 64 (to set the XML transaction committed time value to local time instead of GMT/UTC), specify `OPTIONS=96` (with 96 being the sum of 32 and 64).

Option Value	Description
1	This option is no longer supported. If specified, it will be ignored.
2	Specify `OPTIONS=2` to indicate that long names should be used. This option will cause long names to be sent in place of the default short names used for various elements and attributes. Short names are the default and save on the amount of data being transferred. Long names make for better readability. For example, the short name <F> would appear as <Field> using long names.
4	Specify `OPTIONS=4` to ensure that invalid XML characters found in alphanumeric fields are not translated to spaces.
8	Specify `OPTIONS=8` to ensure that trailing blanks in alphanumeric fields are not removed.

Option Value	Description
16	Specify `OPTIONS=16` to ensure that characters used by XML are not replaced automatically with predefined entity references. For example, if `OPTIONS=16` is set, the ampersand (&) character would not be replaced with the literal "&".
32	Specify `OPTIONS=32` to send the full image on an update and insert. The full before image (if available) and after image of all fields are sent for an update and insert, even if the field values were not changed or are null. **Note:** This option is mutually exclusive with option 128. If both options are specified, then `OPTIONS=32` will be set and `OPTIONS=128` ignored.
64	Specify `OPTIONS=64` to set the XML transaction committed time value to local time instead of GMT/UTC.
128	Specify `OPTIONS=128` if the before image is not to be included when a field is updated/changed if the before image data storage records are included. Primary field before images will still be included because the key is needed to find the record that is to be changed. **Note:** This option is mutually exclusive with option 32. If both options are specified, then `OPTIONS=32` will be set and `OPTIONS=128` ignored.

TRACE

The TRACE keyword parameter can be used to specify the contents of the trace. Specify the TRACE keyword parameter using the syntax `TRACE=`*nnnn*. Possible values of the TRACE (*nnnn*) are listed in the following table. However, if you want to trace multiple control blocks, add their trace values together and enter the total value. For example, to trace the before and after images of the URBD control blocks, you would specify `TRACE=24` because the sum of 8 (URBD control block before image) and 16 (URBD control block after image) is 24.

Note: If tracing is enabled using this keyword parameter, be sure to include the following JCL statement in the startup JCL of the Event Replicator Server:

```
//DDTRACE1 DD SYSOUT=X
```

Trace Value	Description
1	Trace the URBS control block.
2	Trace the URBT control block.
4	Trace the URBR control block.
8	Trace the URBD control block before image.
16	Trace the URBD control block after image.
32	Trace the URBF/URBG control block before image.
64	Trace the URBF/URBG control block after image.
128	Trace the send buffer.
256	Trace the URBY control block.
512	Trace the URBO control block.

Trace Value	Description
1024	Trace the output parameters.
2048	Trace the subscription table.

Step 4. Save the WebSphere MQ Destination Definition

To save the WebSphere MQ destination definition in the Replicator system file:

- Press PF5 on the MQSeries Destination Definition screen.

  The WebSphere MQ destination definition is saved in the Replicator system file.

Creating a File Destination Definition

Using a File destination definition, replicated data is written to the CLOG, using TLOG URBLTDOD records. You can use these records in the CLOG file to create a sequential output file of the replicated data. For more information, read *Creating a Sequential Output File* in *Event Replicator for Adabas Administration and Operations Guide*.

Caution: Be sure that the CLOG is defined in the Event Replicator Server startup JCL (via one or more DDCLOGR*n* DD statements) if you will be using a File destination definition during Event Replicator for Adabas processing. If you do not, a warning message will be issued and the File destination will be set to "Unavailable". For more information about the CLOG, read your Adabas documentation.

To create a File destination definition in the Adabas Event Replicator Subsystem, complete the following steps:

- Step 1. Access the File Destination Definition Creation Area
- Step 2. Specify General and TLOG File Destination Parameters
- Step 3. Save the File Destination Definition

Step 1. Access the File Destination Definition Creation Area

To access the File destination definition creation area of the Adabas Event Replicator Subsystem:

1 Select option **D** from the Adabas Event Replicator Subsystem Main Menu.

  The List of Destinations screen appears.

2 Press PF4 on the List of Destinations screen.

  The Create New Destination screen appears.

```
15:34:11      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                          Create New Destination                    M-RP1290


                        Code    Function
                        ----    -------------------------
                          A     Create Adabas Destination
                          E     Create Broker Destination
                          F     Create File Destination
                          N     Create Null Destination
                          M     Create MQ Destination
                          ?     Help
                          .     Exit
                         ----   -------------------------

                 Code ...  _




Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu  ↩
 ↩
```

3 Select option **F** on the Create New Destination screen.

The File Destination Definition screen appears.

```
15:42:54      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                       File Destination Definition                  M-RP1250

                  Destination Name ............. ________
                  Commit Threshold ........... _________0
                  Destination Active ................. Y
                  Allow Logging ...................... N
                  Event Logging ...................... N
                  Replicate Utility Changes .......... N
                  Open at Startup .................... G

                  TLOG Parms
                  ---------------------------------------
                  Assign Level ....................... 0
                  Completion Level ................... 0
                  SLOG Write Level ................... 0
                  SLOG Read Level .................... 0



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save                                  Menu  ↩
 ↩
```

Step 2. Specify General and TLOG File Destination Parameters

To use the Adabas Event Replicator Subsystem to supply general and TLOG specifications for a File destination definition, complete the following steps:

1 Update the following general fields on the File Destination Definition screen as described in the following table.

Parameter Name	Specify	Default
Destination Name	The unique name for the File destination definition. This is the equivalent of specifying the `DESTINATION NAME` parameter directly in the Event Replicator Server startup job. The specified name must be alphanumeric and be between one and eight characters long.	---
Commit Threshold	The number of URBLTDOD TLOG record bytes that will be written to the CLOG sequential file before the buffers are flushed. For complete information on specifying a value for this parameter, read *DCOMMITTHRESHOLD*, in *Event Replicator for Adabas Reference Guide*.	0
Destination Active	Whether or not this destination definition should be activated for use once it is loaded by the Event Replicator Server. Valid values are "Y" (load and activate the definition) or "N" (load, but do not activate the definition). This is the	Y

Parameter Name	Specify	Default
	equivalent of specifying the DACTIVE parameter in the Event Replicator Server startup job.	
Allow Logging	Whether or not subscription logging should be activated for this destination definition. Valid values are "Y" (activate subscription logging) or "N" (do not activate subscription logging). This is the equivalent of specifying the DLOG parameter in the Event Replicator Server startup job.	N
Event Logging	Whether or not events should be logged by the Event Replicator Server and sent to this destination. This is the equivalent of specifying the DEVENTLOG parameter directly in the Event Replicator Server startup job. Valid values are "Y" or "N". When this optional parameter is set to "Y", Event Replicator Server events are logged to the destination. When this parameter is set to "N" (the default), they are not. Event Replicator Server events are logged in URBS elements. These URBS elements are sent to destinations related to the event itself. The URBS elements are also sent to any other destinations you have defined "Event Logging =Y". If a related destination also is defined with "Event Logging =Y", it will only receive one instance of the URBS element. To access this log of Event Replicator Server events in the destination queue, you must supply your own application that reads the event URBS elements in the destination queue. If such an application does not exist, the logged events simply sit in the queue.	N
Replicate Utility Changes	Whether Adabas utility change replication should be activated for a destination at Event Replicator Server startup. This is the equivalent of specify the DREPLICATEUTI parameter directly in the Event Replicator Server startup job. Valid values are "Y" and "N". If "Y" is specified, utility replication is activated for the destination at Event Replicator Server startup; if "N" is specified, utility replication is not activated for the destination. For more information about replicating utility functions, read *Replicating Utility Functions*, in *Event Replicator for Adabas Concepts*.	N
Open at Startup	Whether or not the destination should be opened at Event Replicator Server startup. Valid values are "Y", "N", or "G", with "G" as the default. When this parameter is set to "Y", the destination is opened at Event Replicator Server startup. When this parameter is set to "N", the destination is *not* opened at Event Replicator Server startup. When this parameter is set to "G", the decision to open the destination at Event Replicator Server startup depends on the setting of the **Open Destinations at start** (GOPEN) global parameter. If GOPEN=YES, the destination is opened at Event Replicator Server startup; if GOPEN=NO, it is not opened. This is the equivalent of specifying the DOPEN parameter in the Event Replicator Server startup job.	G

2 Update the following TLOG fields on the File Destination Definition screen as described in the following table.

Parameter Name	Specify	Default
Assign Level	The level of transaction logging that should occur when a transaction is assigned to a destination for output processing. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the DTLASSIGN parameter in the Event Replicator Server startup job.	0
Completion Level	The level of transaction logging that should occur when a transaction has been successfully output to the messaging system. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the DTLCOMP parameter in the Event Replicator Server startup job.	0
SLOG Write Level	The level of transaction logging that should occur when a transaction has been successfully written to the SLOG file. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the DTLSLOGWRITE parameter in the Event Replicator Server startup job.	0
SLOG Read Level	The level of transaction logging that should occur when a transaction has been successfully read from the SLOG and is about to be queued for output to the destination. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the DTLSLOGREAD parameter in the Event Replicator Server startup job.	0

Step 3. Save the File Destination Definition

To save the File destination definition in the Replicator system file:

- Press PF5 on the File Destination Definition screen.

  The File destination definition is saved in the Replicator system file.

Creating a Null Destination Definition

Using null destinations, data replication is tested without actually sending the data to any destination.

To create a null destination definition in the Adabas Event Replicator Subsystem, complete the following steps:

Step 1. Access the Null Destination Definition Creation Area

To access the null destination definition creation area of the Adabas Event Replicator Subsystem:

1 Select option **D** from the Adabas Event Replicator Subsystem Main Menu.

The List of Destinations screen appears.

2 Press PF4 on the List of Destinations screen.

The Create New Destination screen appears.

```
15:34:11      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                          Create New Destination                    M-RP1290


                        Code    Function
                        ----    -------------------------
                          A     Create Adabas Destination
                          E     Create Broker Destination
                          F     Create File Destination
                          N     Create Null Destination
                          M     Create MQ Destination
                          ?     Help
                          .     Exit
                         ----   -------------------------

                 Code ...  _




Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                              Menu  ↩
 ↩
```

3 Select option **N** on the Create New Destination screen.

The Null Destination Definition screen appears.

```
17:12:32      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2017-10-23
                        Null Destination Definition                   M-RP1240 ↩

                                                                                ↩

Destination Name ......... ________    Destination Class ..... ________
Architecture ................... _2     Destination Class Parameter Data ..... ↩

Commit Threshold ....... _________5    ________________________________________
Destination Active ..... Y             ________________________________________
Allow Logging .......... N             ________________________________________
Replicate Util Chgs .... N             Heartbeat Interval .........          0
Max Output Size ........ __________                                           ↩

Event Logging .......... Y                                                    ↩

Open at Startup ........ G                                                    ↩

DSTATLOG ............... N                                                    ↩

                                                                              ↩

TLOG Parms                                                                    ↩

----------------------                                                        ↩

Assign Level ........ 0                                                       ↩

Completion Level .... 0                                                       ↩

SLOG Write Level .... 0                                                       ↩

SLOG Read Level ..... 0                                                       ↩

Command ==>                                                                   ↩

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save                                      Menu
```

Step 2. Specify General and TLOG Null Destination Parameters

To use the Adabas Event Replicator Subsystem to supply general and TLOG specifications for a null destination definition, complete the following steps:

1 Update the following general fields on the Null Destination Definition screen as described in the following table.

Parameter Name	Specify	Default
Destination Name	The unique name for the null destination definition. This is the equivalent of specifying the DESTINATION NAME parameter directly in the Event Replicator Server startup job. The specified name must be alphanumeric and be between one and eight characters long.	---
Destination Class	Leave this field blank for now. It is described later in this section.	---
Architecture	The data architecture for fields in the URB* control structures sent to the destination. For complete information on calculating a value for this parameter, read *DARC*, in *Event Replicator for Adabas Reference Guide*..	2
Destination Class Parameter Data	Leave this field blank for now. It is described later in this section.	---
Commit Threshold	The number of messages that will be sent to the null destination before a commit is performed for those messages. For complete information on specifying a value for this parameter, read *DCOMMITTHRESHOLD*, in *Event Replicator for Adabas Reference Guide*.	5
Destination Active	Whether or not this destination definition should be activated for use once it is loaded by the Event Replicator Server. Valid values are "Y" (load and activate the definition) or "N" (load, but do not activate the definition). This is the equivalent of specifying the DACTIVE parameter in the Event Replicator Server startup job.	Y
Allow Logging	Whether or not subscription logging should be activated for this destination definition. Valid values are "Y" (activate subscription logging) or "N" (do not activate subscription logging). This is the equivalent of specifying the DLOG parameter in the Event Replicator Server startup job.	N
Replicate Util Chgs	Whether Adabas utility change replication should be activated for a destination at Event Replicator Server startup. This is the equivalent of specify the DREPLICATEUTI parameter directly in the Event Replicator Server startup job. Valid values are "Y" and "N". If "Y" is specified, utility replication is activated for the destination at Event Replicator Server startup; if "N" is specified, utility replication is not activated for the destination. For more information about replicating utility functions, read *Replicating Utility Functions*, in *Event Replicator for Adabas Concepts*.	N
Heartbeat Interval	This is the equivalent of the DHBINTERVAL subparameter for the DESTINATION parameter, defined in Event Replicator Initialization parameters. It specifies the interval (in seconds) at which the heartbeat should be published to a destination. Valid values are 0 or 60 to 86400 (24 hours).	0

Parameter Name	Specify	Default
	Use this subparameter to set the interval at which heartbeat events are sent to a particular destination. Information for a heartbeat event is collected if at least Heartbeat Interval (DHBINTERVAL) seconds have passed since a heartbeat event was last sent to the destination. This heartbeat information is inserted into the replication data stream at the exit from the Event Replicator Server (destination processing). A value of zero (0) indicates that a heartbeat event should not be published for a particular destination. The Heartbeat Interval (DHBINTERVAL) subparameter is only allowed for destinations defined with DTYPE=ETBROKER, DTYPE=MQSERIES, or DTYPE=NULL. The `DHBINTERVAL` parameter is only allowed for destinations defined with DCLASS=SAGTARG. **Note:** The interval between creating heartbeat events may be slightly longer (e.g. 10 seconds longer) than the interval specified for this parameter.	
Max Output Size	The maximum output size (in bytes) for the destination. This is the equivalent of specify the `DMAXOUTPUTSIZE` parameter directly in the Event Replicator Server startup job. Valid values are 0 or any integer ranging from 4096 through 2,147,483,647. You can specify the value for this parameter in a purely numeric form or use K at the end of the number to specify kilobytes. For example, DMAXOUTPUTSIZE=4K is the same as DMAXOUTPUTSIZE=4096. The value for this parameter will be used if it is less than or equal to the maximum output size for the Event Replicator Server (specified using the `MAXOUTPUTSIZE` global parameter) and less than or equal to the maximum output allowed for the messaging system queue being defined. If this value is larger than the `MAXOUTPUTSIZE` specification or the maximum output size allowed by the messaging system, the smaller value will be used. A value of 0 indicates that no specific limit is set for this destination. Instead, the smaller of the `MAXOUTPUTSIZE` specification or the maximum output size allowed by the messaging system will be used.	0
Event Logging	Whether or not events should be logged by the Event Replicator Server and sent to this destination. This is the equivalent of specifying the `DEVENTLOG` parameter directly in the Event Replicator Server startup job. Valid values are "Y" or "N". When this optional parameter is set to "Y", Event Replicator Server events are logged to the destination. When this parameter is set to "N" (the default), they are not. Event Replicator Server events are logged in URBS elements. These URBS elements are sent to destinations related to the event itself. The URBS elements are also sent to any other destinations you have defined "Event Logging =Y". If a related destination also is defined with "Event Logging =Y", it will only receive one instance of the URBS element.	N

Parameter Name	Specify	Default
	To access this log of Event Replicator Server events in the destination queue, you must supply your own application that reads the event URBS elements in the destination queue. If such an application does not exist, the logged events simply sit in the queue.	
Open at Startup	Whether or not the destination should be opened at Event Replicator Server startup. Valid values are "Y", "N", or "G", with "G" as the default. When this parameter is set to "Y", the destination is opened at Event Replicator Server startup. When this parameter is set to "N", the destination is *not* opened at Event Replicator Server startup. When this parameter is set to "G", the decision to open the destination at Event Replicator Server startup depends on the setting of the **Open Destinations at start** (GOPEN) global parameter. If GOPEN=YES, the destination is opened at Event Replicator Server startup; if GOPEN=NO, it is not opened. This is the equivalent of specifying the `DOPEN` parameter in the Event Replicator Server startup job.	G
DSTATLOG	Decides whether or not the associated destination should receive interval statistics published by the Event Replicator Server. This parameter is used with *Replication Monitoring*. Valid values are Y or N (YES or NO). A value of Y indicates that interval statistics should be published to the destination; a value of N indicates they should not. A destination defined with DSTATLOG=Y cannot be specified in a subscription.	N

2 Update the following TLOG fields on the Null Destination Definition screen as described in the following table.

Parameter Name	Specify	Default
Assign Level	The level of transaction logging that should occur when a transaction is assigned to a destination for output processing. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the `DTLASSIGN` parameter in the Event Replicator Server startup job.	0
Completion Level	The level of transaction logging that should occur when a transaction has been successfully output to the messaging system. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the `DTLCOMP` parameter in the Event Replicator Server startup job.	0
SLOG Write Level	The level of transaction logging that should occur when a transaction has been successfully written to the SLOG file. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and	0

Parameter Name	Specify	Default
	file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the DTLSLOGWRITE parameter in the Event Replicator Server startup job.	
SLOG Read Level	The level of transaction logging that should occur when a transaction has been successfully read from the SLOG and is about to be queued for output to the destination. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the DTLSLOGREAD parameter in the Event Replicator Server startup job.	0

Step 3. (Optional) Specify Destination Class Information, If Applicable

The Destination Class and Destination Class Parameter Data fields can be used to invoke and pass parameters to the Event Replicator Target Adapter for this destination. The fields are located at the right of the null destination screen. Do not use these fields unless you want to invoke and pass parameters to the Event Replicator Target Adapter for the destination or unless otherwise requested by a Software AG support representative.

If applicable, use the Destination Class (DCLASS) field to specify the destination class for this destination definition. Valid values are blank or "SAGTARG" (if Event Replicator Target Adapter processing should be invoked). There is no default.

Note: DCLASS=SAGTARG cannot be specified for destinations with DTYPE=ADABAS or FILE. It is only valid for webMethods EntireX, WebSphere MQ, or NULL destinations. When DCLASS=SAGTARG is specified, the ADARUN RPLPARMS parameter must be set to "FILE" or "BOTH" to provide access to any field table (GFFT) definitions.

If you specify a value for the Destination Class field, you can optionally use the Destination Class Parameter Data (DCLASSPARM) field to specify up to 120 bytes of character data to be passed to the optional destination output user exit.

If DCLASS=SAGTARG is specified (if the Destination Class field is set to "SAGTARG") to invoke the Event Replicator Target Adapter for a destination, you may want to specify one or more of the following keyword parameters:

Note: These parameter keywords must be specified in uppercase.

NOSPRE

Specify the "NOSPRE" keyword in the DCLASSPARM parameter if you do not want the subscription name to prefix the names of the tables produced by the Event Replicator Target Adapter. When "NOSPRE" is specified, the schema file name (Predict view name) alone is used for the table names; when "NOSPRE" is *not* specified, the subscription name prefixes the schema file name in the table names.

Note: Oracle identifiers are limited to 30 characters. If NOSPRE is *not* specified and an Oracle RDBMS is used by the Event Replicator Target Adapter, the identifier names may exceed 30 characters and errors may occur. We recommend using NOSPRE if an Oracle RDBMS is also used.

SPRE

Specify the SPRE keyword in the DCLASSPARM parameter if you do want the subscription name to prefix the names of the tables produced by the Event Replicator Target Adapter.

The notation is `SPRE=xxxxxxxx`. Where *xxxxxxxx* can be a subscription prefix to be used instead of the subscription name used in the job execution. The prefix can be 1 to 8 characters in size.

The SPRE keyword is available for use with `SAGTARG` exit.

OPTIONS

The OPTIONS keyword parameter can be used to specify options for the destination. Specify the OPTIONS keyword parameter, using the syntax `OPTIONS=nnnn`. Possible values of the OPTIONS (*nnnn*) are listed in the following table. However, if you want to combine options, add their values together and enter the total value.

For example, if you want to combine option 32 (to send the full image on an update) and option 64 (to set the XML transaction committed time value to local time instead of GMT/UTC), specify `OPTIONS=96` (with 96 being the sum of 32 and 64).

Option Value	Description
1	This option is no longer supported. If specified, it will be ignored.
2	Specify `OPTIONS=2` to indicate that long names should be used. This option will cause long names to be sent in place of the default short names used for various elements and attributes. Short names are the default and save on the amount of data being transferred. Long names make for better readability. For example, the short name <F> would appear as <Field> using long names.
4	Specify `OPTIONS=4` to ensure that invalid XML characters found in alphanumeric fields are not translated to spaces.
8	Specify `OPTIONS=8` to ensure that trailing blanks in alphanumeric fields are not removed.
16	Specify `OPTIONS=16` to ensure that characters used by XML are not replaced automatically with predefined entity references. For example, if `OPTIONS=16` is set, the ampersand (&) character would not be replaced with the literal "&".
32	Specify `OPTIONS=32` to send the full image on an update and insert. The full before image (if available) and after image of all fields are sent for an update and insert, even if the field values were not changed or are null. **Note:** This option is mutually exclusive with option 128. If both options are specified, then `OPTIONS=32` will be set and `OPTIONS=128` ignored.
64	Specify `OPTIONS=64` to set the XML transaction committed time value to local time instead of GMT/UTC.

Option Value	Description
128	Specify `OPTIONS=128` if the before image is not to be included when a field is updated/changed if the before image data storage records are included. Primary field before images will still be included because the key is needed to find the record that is to be changed. **Note:** This option is mutually exclusive with option 32. If both options are specified, then `OPTIONS=32` will be set and `OPTIONS=128` ignored.

TRACE

The TRACE keyword parameter can be used to specify the contents of the trace. Specify the TRACE keyword parameter using the syntax `TRACE=`*nnnn*. Possible values of the TRACE (*nnnn*) are listed in the following table. However, if you want to trace multiple control blocks, add their trace values together and enter the total value. For example, to trace the before and after images of the URBD control blocks, you would specify `TRACE=24` because the sum of 8 (URBD control block before image) and 16 (URBD control block after image) is 24.

Note: If tracing is enabled using this keyword parameter, be sure to include the following JCL statement in the startup JCL of the Event Replicator Server:

```
//DDTRACE1 DD SYSOUT=X
```

Trace Value	Description
1	Trace the URBS control block.
2	Trace the URBT control block.
4	Trace the URBR control block.
8	Trace the URBD control block before image.
16	Trace the URBD control block after image.
32	Trace the URBF/URBG control block before image.
64	Trace the URBF/URBG control block after image.
128	Trace the send buffer.
256	Trace the URBY control block.
512	Trace the URBO control block.
1024	Trace the output parameters.
2048	Trace the subscription table.

Step 4. Save the Null Destination Definition

To save the null destination definition in the Replicator system file:

- Press PF5 on the Null Destination Definition screen.

  The null destination definition is saved in the Replicator system file.

Modifying Destination Definitions

To use the Adabas Event Replicator Subsystem to modify a destination definition in the Replicator system file:

1. List the destination definition in the Adabas Event Replicator Subsystem, as described in *Listing Destination Definitions*, elsewhere in this guide.

   The destinations are listed on the List of Destinations screen.
2. Locate the definition you want to modify on the screen and enter an **M** in the **Sel** column for that definition.

   You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

   An appropriate destination definition screen appears for the destination type you selected. For information on modifying this screen, read the description of adding that type of destination definition, elsewhere in this section.
3. When all modifications have been made, press PF5 to save the changes.

Copying Destination Definitions

To use the Adabas Event Replicator Subsystem to copy a destination definition in the Replicator system file:

1. List the destination definition in the Adabas Event Replicator Subsystem, as described in *Listing Destination Definitions*, elsewhere in this guide.

   The destinations are listed on the List of Destinations screen.
2. Locate the definition you want to copy on the screen and enter a **C** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

A dialog appears requesting a name for the copy of the destination definition.

```
Enter new name: ________
or press PF3 to cancel
                        ↩
```

3 Specify a new, unique name for the copy of the destination definition and press Enter.

The destination definition is copied and the copy appears on the List of Destinations screen.

Activating and Deactivating Destination Definitions

You can use Adabas Online System (AOS) to activate and deactivate destination definitions. For more information, read *Activating and Deactivating Replication Definitions and Databases*, in the *Event Replicator for Adabas Administration and Operations Guide*

Caution: Be careful when you activate and deactivate replication definitions and databases, especially if replication is ongoing at the time. Whenever you activate or deactivate definitions or databases, you run the risk of altering what data is replicated and how that replication occurs. If the Event Replicator Server receives data from an Adabas database for which it has no active definitions, replication simply does not occur.

Deleting Destination Definitions

To use the Adabas Event Replicator Subsystem to delete a destination definition in the Replicator system file:

1 List the destination definition in the Adabas Event Replicator Subsystem, as described in *Listing Destination Definitions*, elsewhere in this guide.

The destinations are listed on the List of Destinations screen.

2 Locate the definition you want to delete on the screen and enter a **D** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

The destination definition is deleted.

5 Maintaining Input Queue (IQUEUE) Definitions

An IQUEUE definition defines the input queue on which Event Replicator for Adabas should listen for requests from webMethods EntireX (or, EntireX Broker) and WebSphere MQ targets. At least one definition is required for every webMethods EntireX or WebSphere MQ target you intend to use.

Listing IQUEUE Definitions

To use the Adabas Event Replicator Subsystem to list the IQUEUE definitions stored in the Replicator system file:

- Select option **Q** from the Adabas Event Replicator Subsystem Main Menu.

  The List of IQUEUE Definitions screen appears showing all of the IQUEUE definitions in the Adabas Event Replicator Subsystem.

```
19:57:51      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-03-12
                        List of Input Queue Definitions              M-RP1300

 Sel   Name   Typ   Sel   Name   Typ   Sel   Name   Typ   Sel   Name   Typ
 ----------------  ----------------  ----------------  ----------------
  _  ABCDEF1  MQ     _                 _                 _
  _  ETBIN1   ETB    _                 _                 _
  _  ETBQUEUE ETB    _                 _                 _
  _  E2       ETB    _                 _                 _
  _                  _                 _                 _
  _                  _                 _                 _
  _                  _                 _                 _
  _                  _                 _                 _
  _                  _                 _                 _
  _                  _                 _                 _
  _                  _                 _                 _
  _                  _                 _                 _
  _                  _                 _                 _
  _                  _                 _                 _
  _                  _                 _                 _

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Repos Exit  Add               -     +                     Menu  ↩
 ↩
```

The function keys on this screen perform the following functions:

Function Key	Description
PF1/F1 (Help)	Provides you with help for this screen.
PF2/F2 (Repos)	Provides you with a pop-up panel that allows you to specify the name of the definition you want to locate in the list. Once you have specified a name on the pop-up panel and pressed Enter, the list is repositioned so the name you selected appears first. You can use an asterisk as a wild card character at the end of the definition name or partial definition name you specify on the pop-up panel. Or, you can simply enter the first few characters of the name to reposition the list to the first occurrence in the list of a name starting with those characters.
PF3/F3 (Exit)	Returns you to the previous screen.
PF4/F4 (Add)	Allows you to add a new definition. A new screen appears.
PF7/F7 (-)	Allows you to scroll backwards through the list of definitions.
PF8/F8 (+)	Allows you to scroll forwards through the list of definitions.
PF12/F12 (Menu)	Returns you to the main menu.

Creating an EntireX Broker IQUEUE Definition

Be sure to read *Using webMethods EntireX as the Messaging System*, in *Event Replicator for Adabas Administration and Operations Guide*, prior to webMethods EntireX as the messaging subsystem.

To use the Adabas Event Replicator Subsystem to add an EntireX Broker IQUEUE definition to the Replicator system file:

1 Select option **Q** from the Adabas Event Replicator Subsystem Main Menu.

The List of IQUEUE Definitions screen appears.

2 Press PF4 on the List of IQUEUE Definitions screen.

The Create New IQUEUE Definition screen appears.

```
18:26:36      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                        Create New IQUEUE Definition                M-RP1390


                   Code    Function
                   ----    -------------------------------------
                     M     Create MQ Input Queue Definition
                     E     Create Broker Input Queue Definition
                     ?     Help
                     .     Exit
                    ----   -------------------------------------


             Code ... _









Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu  ↩
 ↩
```

3 Select option **E** on the Create New IQUEUE Definition screen.

The EntireX Broker Input Queue Definition screen appears.

```
18:27:14      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                                EntireX Broker                       M-RP1320
                             Input Queue Definition



        Input Queue Name ..... ________
        Broker ID ............ ________________________________________
        Broker Service ....... ________________________________________
        Service Name ......... ________________________________________
        Service Class ........ ________________________________________
        Input Queue Buffer Length ... ______2048
        Retry Interval ..... GLOBAL____
        Retry Count ........ GLOBAL____
        Open Queue at start ..... GLOBAL
        Single Conversation mode .... NO_




Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save                                  Menu  ↩
 ↩
```

4 Update the fields on this screen as described in the following table.

Parameter Name	Specify	Default

Broker ID	The EntireX Broker ID you want used for this IQUEUE definition. For the valid values of this parameter, check with your webMethods EntireX Broker development staff and the webMethods EntireX Broker documentation. A maximum of 32 characters can be specified.
	
	Broker IDs come in two formats: one for TCP/IP communications and one for Adabas SVC communication. For TCP/IP communications, the format is:
	
	`ip-address:port-number:TCP`
	
	In this case, the *ip-address* setting is the TCP/IP IP address and the *port-number* setting should match the EntireX Broker `PORT` parameter.
	
	For Adabas SVC communications, the format is:
	
	`'broker-id:SVCnnn:NET'`
	
	In this case, the *broker-id* setting should match the EntireX Broker `BROKER-ID` parameter in the Broker ETBFILE DD. The *nnn* setting should
		`ETBBROKERID` value

Parameter Name	Specify	Default
	match either the EntireX Broker ADASVC or ADA5SVC parameters in the Broker PARMS DD statement. This is the equivalent of specifying the IQETBBROKERID parameter in the Event Replicator Server startup job.	
Broker Service	The EntireX Broker service you want used for this IQUEUE definition. This should be the same as the value specified for the SERVICE parameter in EntireX Broker. The ID can be up to 32 characters long. For the valid values of this parameter, check with your webMethods EntireX Broker development staff and the webMethods EntireX Broker documentation. This is the equivalent of specifying the IQETBSERVICE parameter in the Event Replicator Server startup job.	---
Input Queue Buffer Length	The length, in bytes, of the input buffer associated with this input queue. Valid values range from "2048" through "2,147,483,647". However, the practical maximum value is restricted by the amount of virtual storage available in the Event Replicator Server address space. If you enter a value less than "2048", "2048" is used. This is the equivalent of specifying the IQBUFLEN parameter in the Event Replicator Server startup job. This value should be set to a value greater than or equal to the largest message that will be received by the input queue. When the input queue will receive data as a part of node-to-node replication, the largest message will be limited by the minimum of the following Event Replicator Server settings: MAXOUTPUTSIZE parameter, DMAXOUTPUTSIZE parameter (if specified for the destination), or the message limit imposed by the messaging system.	2048
Input Queue Name	A unique name for the IQUEUE definition. The name must use alphanumeric characters and be between one and 8 characters long. This is the equivalent of specifying the IQUEUE NAME parameter in the Event Replicator Server startup job.	---
Open Queue at Start	Whether the input queue should be opened automatically when the Event Replicator Server starts up or when an RPLREFRESH command is run. Valid values are GLOBAL, NO, or YES: ■ GLOBAL (the default): This setting indicates that the decision about automatically opening the input queue should be based on the setting of the GOPEN global parameter. If the GOPEN global parameter is set to "YES", the input queue is opened; if GOPEN is set to NO, the input queue is not opened. ■ NO: This setting indicates that the input queue should never be opened automatically when the Event Replicator Server starts up or when an RPLREFRESH command is run.	GLOBAL

Parameter Name	Specify	Default
	■ YES: This setting indicates that the input queue should always be opened automatically when the Event Replicator Server starts up or when an RPLREFRESH command is run. This is the equivalent of specifying the `IQOPEN` parameter in the Event Replicator Server startup job.	
Retry Count	The number of times that an attempt to open the input queue will be retried at the interval specified by the Retry Interval parameter. This is the equivalent of specifying the `IQRETRYCOUNT` parameter directly in the Event Replicator Server startup job. Valid values range from 0 through 2,147,483,647 or the literal "GLOBAL". If the value "GLOBAL" is specified for this parameter, the specification for the **Retry Count global variable** will be used. Any retry attempts will occur at the interval specified by the Retry Interval parameter. A value of zero indicates that no retry attempt to open this input queue should occur.	The value of the **Retry Count global variable**
Retry Interval	The default number of seconds between retry attempts to open the input queue identified by this definition. This is the equivalent of specifying the `IQRETRYINTERVAL` parameter directly in the Event Replicator Server startup job. Valid values are 0, 5 through 2,147,483,647, or the literal "GLOBAL". If the value "GLOBAL" is specified for this parameter, the specification for the **Retry Interval global variable** will be used. A value of zero indicates that no retry attempt to open this input queue should occur. Except for a specification of zero, the minimum value that can be specified for this parameter is 5 seconds.	The value of the **Retry Interval global variable**
Service Class	The EntireX Broker service class you want used for this IQUEUE definition. This should be the same as the value specified for the CLASS parameter in EntireX Broker. The name can be up to 32 characters long. For the valid values of this parameter, check with your webMethods EntireX Broker development staff and the webMethods EntireX Broker documentation. This is the equivalent of specifying the `IQETBSERVICECLASS` parameter in the Event Replicator Server startup job.	---
Service Name	The name of the EntireX Broker service you want used for this IQUEUE definition. This should be the same as the value specified for the SERVER parameter in EntireX Broker. The name can be up to 32 characters long. For the valid values of this parameter, check with your webMethods EntireX Broker development staff and the webMethods EntireX Broker documentation. This is the equivalent of specifying the `IQETBSERVICENAME` parameter in the Event Replicator Server startup job.	---

5 Press PF5 to save the IQUEUE definition.

Creating a WebSphere MQ IQUEUE Definition

Be sure to read *Using WebSphere MQ as the Messaging System*, in *Event Replicator for Adabas Administration and Operations Guide*, prior to using WebSphere MQ as the messaging subsystem.

Note: If you are running on z/OS using IBM WebSphere MQ Series definitions for your Event Replicator DESTINATION or IQUEUE definitions, a S0D3 abend can occur if you run it as a started task and specify the parameter REUSASID=YES. This is a documented IBM WebSphere MQ Series issue.

To use the Adabas Event Replicator Subsystem to add a WebSphere MQ IQUEUE definition to the Replicator system file:

1 Select option **Q** from the Adabas Event Replicator Subsystem Main Menu.

The List of IQUEUE Definitions screen appears.

2 Press PF4 on the List of IQUEUE Definitions screen.

The Create New IQUEUE Definition screen appears.

```
18:26:36      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                      Create New IQUEUE Definition                  M-RP1390


                  Code    Function
                  ----    -------------------------------------
                    M     Create MQ Input Queue Definition
                    E     Create Broker Input Queue Definition
                    ?     Help
                    .     Exit
                   ----   -------------------------------------


           Code ... _




Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                Menu  ↩
 ↩
```

3 Select option **M** on the Create New IQUEUE Definition screen.

The MQSeries Input Queue Definition screen appears.

```
18:27:53      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                                  MQSeries                           M-RP1310
                             Input Queue Definition



Input Queue Name ........... ________
Queue Manager Name.......... ________________________________________________
MQ Queue Name .............. ________________________________________________
Input Queue Buffer Length .. _______2048
Retry Interval ............. GLOBAL____
Retry Count ................ GLOBAL____
Open Queue at start ........ GLOBAL









Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save                                    Menu  ↩
 ↩
```

4 Update the fields on this screen as described in the following table.

Parameter Name	Specify	Default
Input Queue Buffer Length	The length, in bytes, of the input buffer associated with this input queue. Valid values range from "2048" through "2,147,483,647". However, the practical maximum value is restricted by the amount of virtual storage available in the Event Replicator Server address space. If you enter a value less than "2048", "2048" is used. This is the equivalent of specifying the `IQBUFLEN` parameter in the Event Replicator Server startup job. This value should be set to a value greater than or equal to the largest message that will be received by the input queue. When the input queue will receive data as a part of node-to-node replication, the largest message will be limited by the minimum of the following Event Replicator Server settings: `MAXOUTPUTSIZE` parameter, `DMAXOUTPUTSIZE` parameter (if specified for the destination), or the message limit imposed by the messaging system.	2048

Parameter Name	Specify	Default
Input Queue Name	A unique name for the IQUEUE definition. The name must use alphanumeric characters and be between one and 8 characters long. This is the equivalent of specifying the `IQUEUE NAME` parameter in the Event Replicator Server startup job.	---
MQ Queue name	The name of the WebSphere MQ queue you want used for this IQUEUE definition. The name can be up to 48 characters long. For the valid values of this parameter, check with your WebSphere MQ development staff and the WebSphere MQ documentation. This is the equivalent of specifying the `IQMQQNAME` parameter in the Event Replicator Server startup job.	---
Open Queue at Start	Whether the input queue should be opened automatically when the Event Replicator Server starts up or when an RPLREFRESH command is run. Valid values are GLOBAL, NO, or YES: ■ GLOBAL (the default): This setting indicates that the decision about automatically opening the input queue should be based on the setting of the GOPEN global parameter. If the GOPEN global parameter is set to "YES", the input queue is opened; if GOPEN is set to NO, the input queue is not opened. ■ NO: This setting indicates that the input queue should never be opened automatically when the Event Replicator Server starts up or when an RPLREFRESH command is run. ■ YES: This setting indicates that the input queue should always be opened automatically when the Event Replicator Server starts up or when an RPLREFRESH command is run. This is the equivalent of specifying the `IQOPEN` parameter in the Event Replicator Server startup job.	GLOBAL
Queue Manager Name	The name of the WebSphere MQ queue manager you want used for this IQUEUE definition. The name can be up to 48 characters long. For the valid values of this parameter, check with your WebSphere MQ development staff and the WebSphere MQ documentation. This is the equivalent of specifying the `IQMQQMGRNAME` parameter in the Event Replicator Server startup job.	---
Retry Count	The number of times that an attempt to open the input queue will be retried at the interval specified by the Retry Interval parameter. This is the equivalent of specifying the `IQRETRYCOUNT` parameter directly in the Event Replicator Server startup job. Valid values range from 0 through 2,147,483,647 or the literal "GLOBAL". If the value "GLOBAL" is specified for this parameter, the specification for the **Retry Count global variable** will be used. Any retry attempts will occur	The value of the **Retry Count global variable**

Parameter Name	Specify	Default
	at the interval specified by the Retry Interval parameter. A value of zero indicates that no retry attempt to open this input queue should occur.	
Retry Interval	The default number of seconds between retry attempts to open the input queue identified by this definition. This is the equivalent of specifying the `IQRETRYINTERVAL` parameter directly in the Event Replicator Server startup job. Valid values are 0, 5 through 2,147,483,647, or the literal "GLOBAL". If the value "GLOBAL" is specified for this parameter, the specification for the **Retry Interval global variable** will be used. A value of zero indicates that no retry attempt to open this input queue should occur. Except for a specification of zero, the minimum value that can be specified for this parameter is 5 seconds.	The value of the **Retry Interval global variable**

5 Press PF5 to save the IQUEUE definition.

Modifying IQUEUE Definitions

To use the Adabas Event Replicator Subsystem to modify an IQUEUE definition in the Replicator system file:

1 List the IQUEUE definitions in the Adabas Event Replicator Subsystem, as described in *Listing IQUEUE Definitions*, elsewhere in this guide.

The IQUEUE definitions are listed on the List of IQUEUE Definitions screen.

2 Locate the definition you want to modify on the screen and enter an **M** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

An appropriate IQUEUE definition screen appears for the IQUEUE type you selected. For information on modifying this screen, read the description of adding that type of IQUEUE definition, elsewhere in this section.

3 When all modifications have been made, press PF5 to save the changes.

Copying IQUEUE Definitions

To use the Adabas Event Replicator Subsystem to copy an IQUEUE definition in the Replicator system file:

1 List the IQUEUE definitions in the Adabas Event Replicator Subsystem, as described in *Listing IQUEUE Definitions*, elsewhere in this guide.

The IQUEUE definitions are listed on the List of IQUEUE Definitions screen.

2 Locate the definition you want to copy on the screen and enter a **C** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

A dialog appears requesting a name for the copy of the IQUEUE definition.

```
Enter new name: ________
or press PF3 to cancel
                          ↩
```

3 Specify a new, unique name for the copy of the IQUEUE definition and press Enter.

The IQUEUE definition is copied and the copy appears on the List of IQUEUE Definitions screen.

Deleting IQUEUE Definitions

To use the Adabas Event Replicator Subsystem to delete an IQUEUE definition in the Replicator system file:

1 List the IQUEUE definitions in the Adabas Event Replicator Subsystem, as described in *Listing IQUEUE Definitions*, elsewhere in this guide.

The IQUEUE definitions are listed on the List of IQUEUE Definitions screen.

2 Locate the definition you want to delete on the screen and enter a **D** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

The IQUEUE definition is deleted.

6 Maintaining Resend Buffer Definitions

A resend buffer definition defines a resend buffer that can be used by any subscription to expedite the retransmission of a transaction. No resend buffer definitions are required. Resend buffers are defined using the RESENDBUFFER initialization parameter, but should be maintained using the Adabas Event Replicator Subsystem.

Listing Resend Buffer Definitions

To use the Adabas Event Replicator Subsystem to list the resend buffer definitions stored in the Replicator system file:

- Select option **R** from the Adabas Event Replicator Subsystem Main Menu.

  The List of Resend Buffer Definitions screen appears showing all of the resend buffer definitions in the Adabas Event Replicator Subsystem.

```
18:28:30      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                     List of Resend Buffer Definitions              M-RP1600

 Sel   Name         Sel   Name         Sel   Name         Sel   Name
 --------------     --------------     --------------     -------------

  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _
  _                  _                  _                  _

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Repos Exit  Add                -     +                     Menu  ↩
 ↩
```

The function keys on this screen perform the following functions:

Function Key	Description
PF1/F1 (Help)	Provides you with help for this screen.
PF2/F2 (Repos)	Provides you with a pop-up panel that allows you to specify the name of the definition you want to locate in the list. Once you have specified a name on the pop-up panel and pressed Enter, the list is repositioned so the name you selected appears first. You can use an asterisk as a wild card character at the end of the definition name or partial definition name you specify on the pop-up panel. Or, you can simply enter the first few characters of the name to reposition the list to the first occurrence in the list of a name starting with those characters.
PF3/F3 (Exit)	Returns you to the previous screen.
PF4/F4 (Add)	Allows you to add a new definition. A new screen appears.
PF7/F7 (-)	Allows you to scroll backwards through the list of definitions.
PF8/F8 (+)	Allows you to scroll forwards through the list of definitions.
PF12/F12 (Menu)	Returns you to the main menu.

Adding Resend Buffer Definitions

To use the Adabas Event Replicator Subsystem to add a resend buffer definition to the Replicator system file:

1 Select option **R** from the Adabas Event Replicator Subsystem Main Menu.

The List of Resend Buffer Definitions screen appears.

2 Press PF4 on the List of Resend Buffer Definitions.

The Resend Buffer Definition screen appears.

```
18:28:56      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                         Resend Buffer Definition                   M-RP1610



                    Resend Buffer Name ............... _________
                    Resend Buffer Size ............... _____32768

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save                                    Menu  ↩
 ↩
```

3 Fill in the values for the fields on this screen as described below:

Parameter Name	Description	Default
Resend Buffer Name (RESENDBUFFERNAME)	Specify a unique name for the resend buffer definition. The name must use alphanumeric characters and be between one and 8 characters long. There are some constraints on the name. It must: ■ Be comprised of one to eight uppercase, alphanumeric characters and can include the special characters "@", "$", or "#". If the name is less than eight characters, it is automatically padded on the right with blanks. ■ Not begin with a numeric character or a blank. ■ Have no embedded blanks. ■ Not begin with the letters "SYS".	---
Resend Buffer Size(RSIZE)	Specify the amount of storage allocated to the buffer. The value must be numeric and specified in units of KB. The default is 32 (32K), and the maximum is 2,097,151 (2,097,151K). If a resend buffer is defined for a subscription that delivers data to multiple destinations, multiple copies of the sent data may be saved, one copy for each destination. The specification of **Resend Buffer Size**	32

Parameter Name	Description	Default
	must be large enough to accommodate these multiple copies of the data.	

4 Press PF5 to save the resend buffer definition.

Modifying Resend Buffer Definitions

To use the Adabas Event Replicator Subsystem to modify a resend buffer definition in the Replicator system file:

1 List the resend buffer definitions in the Adabas Event Replicator Subsystem, as described in *Listing Resend Buffer Definitions*, elsewhere in this guide.

The resend buffer definitions are listed on the List of Resend Buffer Definitions screen.

2 Locate the definition you want to modify on the screen and enter an **M** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

An appropriate resend buffer definition screen appears for the resend buffer you selected. For information on modifying this screen, read the description of adding resend buffer definitions in *Adding Resend Buffer Definitions*, elsewhere in this section.

3 When all modifications have been made, press PF5 to save the changes.

Copying Resend Buffer Definitions

To use the Adabas Event Replicator Subsystem to copy a resend buffer definition in the Replicator system file:

1 List the resend buffer definitions in the Adabas Event Replicator Subsystem, as described in *Listing Resend Buffer Definitions*, elsewhere in this guide.

The resend buffer definitions are listed on the List of Resend Buffer Definitions screen.

2 Locate the definition you want to copy on the screen and enter a **C** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

A dialog appears requesting a name for the copy of the resend buffer definition.

```
Enter new name: ________
or press PF3 to cancel
                          ↩
```

3 Specify a new, unique name for the copy of the resend buffer definition and press Enter.

The resend buffer definition is copied and the copy appears on the List of Resend Buffer Definitions screen.

Deleting Resend Buffer Definitions

To use the Adabas Event Replicator Subsystem to delete a resend buffer definition in the Replicator system file:

1 List the resend buffer definitions in the Adabas Event Replicator Subsystem, as described in *Listing Resend Buffer Definitions*, elsewhere in this guide.

The resend buffer definitions are listed on the List of Resend Buffer Definitions screen.

2 Locate the definition you want to delete on the screen and enter a **D** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

The resend buffer definition is deleted.

7 Maintaining Transaction Filter Definitions

A transaction filter definition specifies filter conditions for replication, based on the values of fields in the database records. No transaction filter definitions are required. Transaction filter definitions are defined using the `FILTER` initialization parameter, but should be maintained using the Adabas Event Replicator Subsystem.

Listing Transaction Filter Definitions

To use the Adabas Event Replicator Subsystem to list the transaction filter definitions stored in the Replicator system file:

- Select option **F** from the Adabas Event Replicator Subsystem Main Menu.

  The List of Transaction Filters screen appears showing all of the transaction filter definitions in the Adabas Event Replicator Subsystem.

```
18:28:56      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                        List of Transaction Filters                  M-RP1140

 Sel   Name          Sel   Name          Sel   Name          Sel   Name
 ------------        ------------        ------------        ------------
  _  ABCDEF01         _  52ZA05           _  52ZA19           _  52207313
  _  ABCDEF02         _  52ZA06           _  52005019         _  52207414
  _  AFD              _  52ZA07           _  52206101         _  52207515
  _  DNXABF           _  52ZA08           _  52206202         _  52207616
  _  NXNAF            _  52ZA09           _  52206303         _  52207717
  _  SMXTF            _  52ZA10           _  52206404         _  52207818
  _  UYU              _  52ZA11           _  52206505         _  52207919
  _  WIS1             _  52ZA12           _  52206606         _  52208101
  _  WIS2             _  52ZA13           _  52206707         _  52208202
  _  XYZ              _  52ZA14           _  52206808         _  52208303
  _  52ZA01           _  52ZA15           _  52206909         _  52208404
  _  52ZA02           _  52ZA16           _  52207010         _  52208505
  _  52ZA03           _  52ZA17           _  52207111         _  52208606
  _  52ZA04           _  52ZA18           _  52207212         _  52208707

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Repos Exit  Add               -     +                      Menu  ↩
 ↩
```

The function keys on this screen perform the following functions:

Function Key	Description
PF1/F1 (Help)	Provides you with help for this screen.
PF2/F2 (Repos)	Provides you with a pop-up panel that allows you to specify the name of the definition you want to locate in the list. Once you have specified a name on the pop-up panel and pressed Enter, the list is repositioned so the name you selected appears first. You can use an asterisk as a wild card character at the end of the definition name or partial definition name you specify on the pop-up panel. Or, you can simply enter the first few characters of the name to reposition the list to the first occurrence in the list of a name starting with those characters.
PF3/F3 (Exit)	Returns you to the previous screen.
PF4/F4 (Add)	Allows you to add a new definition. A new screen appears.
PF7/F7 (-)	Allows you to scroll backwards through the list of definitions.
PF8/F8 (+)	Allows you to scroll forwards through the list of definitions.
PF12/F12 (Menu)	Returns you to the main menu.

Adding Transaction Filter Definitions

To use the Adabas Event Replicator Subsystem to add a transaction filter definition to the Replicator system file, complete the following steps:

Step 1. Access the Transaction Filter Definition Area of the Adabas Event Replicator Subsystem

To use the Adabas Event Replicator Subsystem to add a transaction filter definition to the Replicator system file:

1 Select option **F** from the Administration menu.

The List of Transaction Filters screen appears showing all of the transaction filter definitions in the Adabas Event Replicator Subsystem.

```
18:28:56      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                        List of Transaction Filters                  M-RP1140

 Sel   Name          Sel   Name          Sel   Name          Sel   Name
 ------------        ------------        ------------        ------------
  _  ABCDEF01         _  52ZA05           _  52ZA19           _  52207313
  _  ABCDEF02         _  52ZA06           _  52005019         _  52207414
  _  AFD              _  52ZA07           _  52206101         _  52207515
  _  DNXABF           _  52ZA08           _  52206202         _  52207616
  _  NXNAF            _  52ZA09           _  52206303         _  52207717
  _  SMXTF            _  52ZA10           _  52206404         _  52207818
  _  UYU              _  52ZA11           _  52206505         _  52207919
  _  WIS1             _  52ZA12           _  52206606         _  52208101
  _  WIS2             _  52ZA13           _  52206707         _  52208202
  _  XYZ              _  52ZA14           _  52206808         _  52208303
  _  52ZA01           _  52ZA15           _  52206909         _  52208404
  _  52ZA02           _  52ZA16           _  52207010         _  52208505
  _  52ZA03           _  52ZA17           _  52207111         _  52208606
  _  52ZA04           _  52ZA18           _  52207212         _  52208707

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Repos Exit  Add                 -     +                      Menu  ↩
 ↩
```

2 Press PF4 on the List of Transaction Filters.

The Transaction Filter screen appears.

```
18:30:29      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                              Transaction Filter                     M-RP1150

Transaction Filter Name ..... ________                               1 of 1
Exclude or Include Records .. I

          ------  Source  --------------------            -----  Target  -----
Sel Group Field   PE    MU Image  Begin Length    Cond    Field Value
--- ----- ------------------------------------    ----    --------------------









Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit  Add   Save        -     +                      Menu  ↩
 ↩
```

Step 2. Specify a Transaction Filter Definition Name and Type

To specify a transaction filter definition name and type:

1 Tab to the **Transaction Filter Name** field and specify a unique name for the transaction filter definition. The name must use alphanumeric characters and be between one and 8 characters long. This is the equivalent of specifying the `FILTER NAME` parameter in the Event Replicator Server startup job.

2 Tab to the **Exclude or Include Records** field and specify an "I" to include (replicate) the records selected by the filter definition or an "E" to exclude (do not replicate) records selected by the filter definition. This is the equivalent of specifying the `FRECORDS` parameter in the Event Replicator Server startup job. The default is "I".

3 Although no field filter conditions have yet been specified for the transaction filter definition, press PF5 to save it.

Step 3. Add Filter Conditions to the Transaction Filter Definition

For more information about rules of filter conditions, read *Rules for Writing Filter Conditions* elsewhere in this manual.

To add filter conditions to the transaction filter definition:

1 Press PF4 to define a field filter conditions for the transaction filter definition. Up to 2500 filter conditions can be specified.

The Filter Condition screen appears with the transaction filter name listed at the top of the screen.

```
18:31:16      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                              Filter Condition                       M-RP1155

Transaction Filter Name ..... MYFILT
        -------  Source  ------           -------  Target  ------
  Group Field   PE    MU  Image Condition Field   PE    MU  Image
   2nd line:  Begin Length                    Begin Length
  ----- ---------------------- --------- ----------------------
   ___   __   _____ _____  __       __     __   _____ _____  __    or value(s)
              _____ _____                    _____ _____
Target Value 1   ..
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
Target Value 2   ..
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save        -     +                      Menu  ↩
 ↩
```

Note that you can specify a target field, part of a target field, or multiple target values on this screen:

- The target field is specified under the **Target** heading in the **Field** column.
- The part of a target field is specified under the **Target** heading using a combination of the **Field** column and the **Begin** and **Length** columns (which are on the second line).
- Multiple target values can be specified in the **Target Value** *n* fields at the bottom part of the screen.

2 Update the fields on this screen as described in the following table.

Note: The transaction filter definition name cannot be changed.

Parameter Name	Specify	Default
Group	A group number you can use to group field filters together within a transaction filter definition. All of the field filters with the same group number are blocked and logically ANDed together when the filters are examined during subscription processing. In other words, a field in the database must meet all of the criteria of the group before it is selected. Likewise, different groups of field filters are logically ORed together. In other words, a field need only meet the criteria specified by one of the groups to be selected. Valid values are numbers ranging from "1" through "999". The equivalent processing in the Event Replicator Server startup job is available through the use of the OR keyword within a series of `FFIELD` parameters. For more information about rules of filter conditions, read *Rules for Writing Filter Conditions* elsewhere in this manual.	---
Source Field	The two-byte Adabas field code for the field to be compared. This field must be in the format buffer specified for the file in the SFILE definition of the subscription. This is the equivalent of specifying the `FFIELD` parameters in the Event Replicator Server startup job. If you want to specify part of a source field for the comparison, specify: ■ a starting byte number in the Source Begin field on the second line. ■ optionally, the length of the part to be compared in the Source Length field on the second line. **Note:** The format of the complete field is used for partial field comparisons. If the formats of a source field or partial source field and a target field or partial target field do not match, the comparisons may always result in an unequal condition. For example, comparing an alphanumeric field to a packed field will always result in an unequal condition. For more information about rules of filter conditions, read *Rules for Writing Filter Conditions* elsewhere in this manual.	---
Source PE	The index number (occurrence) of the periodic group (PE) to which the condition relates if the source field in this field filter is a PE field. Valid values range from 0 through 191. This is the equivalent of specifying the `FSPE` subparameter parameter in the Event Replicator Server startup job.	0, indicating the source field is not a PE field.

Parameter Name	Specify	Default
Source MU	The index number of the multiple-value field (MU) to which the condition relates if the source field in this field filter is an MU field. Valid values range from 0 through 191. This is the equivalent of specifying the FSMU subparameter in the Event Replicator Server startup job.	0, indicating the source field is not an MU field.
Source Image	Whether the source field is in the after image (AI), before image (BI), or the default image of the record. Valid values are "AI" and "BI". This is the equivalent for specifying the FSIMAGE subparameter in the Event Replicator Server startup job.	AI for adds and updates; BI for deletes
Source Begin	The starting byte number of the partial Adabas source field to be compared. This field is only used if you want to specify a partial field for comparison and only if the field is of alphanumeric or binary format. This is the equivalent of specifying the FSBEGIN subparameter in the Event Replicator Server startup job. **Note:** The format of the complete field is used for partial field comparisons. Valid comparisons of different field types are listed in *Field Type Considerations*, elsewhere in this guide. For fixed length fields, valid values range from "1" (the start of the field) through the maximum length of the field (the last byte of the field). For variable length fields, valid values range from "1" (the start of the field) to the maximum length allowed for that field type. Counting occurs from left to right beginning with 1 for fields defined with alphanumeric format, and from right to left beginning with 1 for fields defined with binary format.	1
Source Length	The numeric length of the partial Adabas source field that should be compared. This field is only used if you want to specify a partial field for comparison and only if the field is of alphanumeric or binary format. This is the equivalent of specifying the FSLENGTH subparameter in the Event Replicator Server startup job. **Note:** The format of the complete field is used for partial field comparisons. Valid comparisons of different field types are listed in *Field Type Considerations*, elsewhere in this guide. For fixed length fields, errors will occur if the sum of the values of the Source Begin and Source Length parameters exceeds the fixed length of the field. For variable length fields, the sum of the values of the Source Begin and Source Length parameters must not exceed the maximum length of the field plus 1. For example, if a variable length field has format "A" with a maximum length of 253 bytes, settings of Source Begin=1 and Source Length=253 are valid, but settings of Source Begin=2 and Source Length=254 are not.	If Source Begin is not specified, the default value is the entire field. If Source Begin is specified, the default value is the maximum length of the field minus the value of the Source Begin parameter plus 1.
Condition	A condition operator code for the filter. Valid values are "EQ" (equal to), "NE" (not equal to), "LT" (less than), "LE" (less than or equal to), "GT" (greater than), or "GE" (greater than or equal to). This is the equivalent	EQ

Parameter Name	Specify	Default
	of specifying the FCOND subparameter in the Event Replicator Server startup job. When EQ or NE are specified, multiple target values and target values using wildcards can be tested. For all other condition codes, only single target values without wildcards can be tested. For more information about rules of filter conditions, read ***Rules for Writing Filter Conditions*** elsewhere in this manual.	
Target Field	The two-byte Adabas field code for the field with which the source field will be compared. Use single quotation marks around the field code. This field must be in the same record as the source field. This is the equivalent of specifying the FTARGET parameter in the Event Replicator Server startup job. If you want to specify part of a target field for the comparison, specify: ■ a starting byte number in the Target Begin field on the second line. ■ optionally, the length of the part to be compared in the Target Length field on the second line. **Note:** The format of the complete field is used for partial field comparisons. Valid comparisons of different field types are listed in ***Field Type Considerations***, elsewhere in this guide. This field is mutually exclusive with the Target Value *n* fields. If you specify the target field code, you cannot specify values in the Target Value *n* fields. For more information about rules of filter conditions, read ***Rules for Writing Filter Conditions*** elsewhere in this manual.	---
Target PE	The index number of the periodic group (PE) to which the condition relates if the target field in this field filter is a PE field. Valid values range from 0 through 191. This is the equivalent of specifying the FTPE subparameter in the Event Replicator Server startup job. This field is mutually exclusive with the Target Value *n* fields. If you specify the target field code, you cannot specify values in the Target Value *n* fields.	0, indicating the target field is not a PE field.
Target MU	The index number (occurrence) of the multiple-value field (MU) to which the condition relates if the target field in this field filter is an MU field. Valid values range from 0 through 191. This is the equivalent of specifying the FTMU subparameter in the Event Replicator Server startup job. This field is mutually exclusive with the Target Value *n* fields. If you specify the target field code, you cannot specify values in the Target Value *n* fields.	0, indicating the target field is not an MU field.

Parameter Name	Specify	Default
Target Image	Whether the target field is in the after image (AI), before image (BI), or the default image of the record. Valid values are "AI" and "BI". This is the equivalent of specifying the FTIMAGE subparameter in the Event Replicator Server startup job. This field is mutually exclusive with the Target Value *n* fields. If you specify the target field code, you cannot specify values in the Target Value *n* fields.	AI for adds and updates; BI for deletes
Target Begin	The starting byte number of the partial Adabas target field at which the comparison should begin. This field should only be specified if you want to specify a partial field for comparison, if the field is of alphanumeric or binary format, and only if an Adabas target field (Target Field) is specified. This is the equivalent of specifying the FTBEGIN subparameter in the Event Replicator Server startup job. **Note:** The format of the complete field is used for partial field comparisons. Valid comparisons of different field types are listed in ***Field Type Considerations***, elsewhere in this guide. For fixed length fields, valid values range from "1" (the start of the field) through the maximum length of the field (the last byte of the field). For variable length fields, valid values range from "1" (the start of the field) to the maximum length allowed for that field type. Counting occurs from left to right beginning with 1 for fields defined with alphanumeric format, and from right to left beginning with 1 for fields defined with binary format.	1
Target Length	The numeric length of the partial Adabas target field that should be used for the comparison. This field should only be specified if you want to specify a partial field for comparison, if the field is of alphanumeric or binary format, and only if an Adabas target field (Target Field) is specified. This is the equivalent of specifying the FTLENGTH subparameter in the Event Replicator Server startup job. **Note:** The format of the complete field is used for partial field comparisons. Valid comparisons of different field types are listed in ***Field Type Considerations***, elsewhere in this guide. For fixed length fields, errors will occur if the sum of the values of the Target Begin and Target Length parameters exceeds the fixed length of the field. For variable length fields, the sum of the values of the Target Begin and Target Length parameters must not exceed the maximum length of the field plus 1. For example, if a variable length field has format "A" with a maximum length of 253 bytes, settings of Target Begin=1 and Target Length=253 are valid, but settings of Target Begin=2 and Target Length=254 are not.	If Target Begin is not specified, the default value is the entire field. If Target Begin is specified, the default value is the maximum length of the field minus the value of the Target Begin parameter plus 1.

Parameter Name	Specify	Default
Target Value *n*	A value against which the source field will be compared. Only one value can be specified in each Target Value field. Up to 128 Target Value fields are available in which you can specify values; use the PF7 and PF8 keys to scroll through them. Strings that include blanks should be enclosed in single quotes. Apostrophes in strings must be doubled (for example: 'six o''clock'). A maximum of 254 characters can be specified for each value. Each value may consist of either free-format characters or a mix of elements specified using the A() or X() notation. ■ If free-format data consists entirely of numeric data (including an optional leading "+" or "-" character) it is treated as a numeric value. ■ If a value (or part of a value) is specified using A() notation, it will be treated as alphabetic data. ■ Hexadecimal values may be specified using X() notation. A value must be specified entirely as free-format data, or composed of one or more A() or X() subelements. If a value begins with an A() or X() subelement all remaining subelements of the value must be so specified. For more information about rules of filter conditions, read ***Rules for Writing Filter Conditions*** elsewhere in this manual. This field is mutually exclusive with the Target Field, Target Image, Target MU, and Target PE fields. You cannot specify values for the Target Field, Target Image, Target MU, or Target PE fields if you have specified a value for this field. This is the equivalent of specifying the `FLIST` subparameter in the Event Replicator Server startup job.	---

Step 4. Save the Transaction Filter Definition

To save the transaction filter definition:

■ Press PF5 to save the transaction filter definition in the Replicator system file.

Modifying Transaction Filter Definitions

To use the Adabas Event Replicator Subsystem to modify a transaction filter definition in the Replicator system file:

1 List the transaction filter definitions in the Adabas Event Replicator Subsystem, as described in *Listing Transaction Filter Definitions*, elsewhere in this guide.

The transaction filter definitions are listed on the List of Transaction Filters screen.

2 Locate the definition you want to modify on the screen and enter an **M** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

An appropriate transaction filter definition screen appears for the transaction filter you selected.

3 Modify the **Exclude or Include Records** field, as necessary. If you want to modify a filter condition specification, enter an "M" next to it in the list to display and update the Filter Condition screen for that condition. If you want to delete a filter condition from the transaction filter definition, enter a "D" next to the condition in the list.

Note: You cannot alter the name of the transaction filter definition. If you want to rename a transaction filter definition, first **copy** it using the name you want and then **delete** the original.

For information on modifying this screen, read the description of adding transaction filter definitions in *Adding Transaction Filter Definitions*, elsewhere in this section.

4 When all modifications have been made, press PF5 to save the changes.

Copying Transaction Filter Definitions

To use the Adabas Event Replicator Subsystem to copy a transaction filter definition in the Replicator system file:

1 List the transaction filter definitions in the Adabas Event Replicator Subsystem, as described in *Listing Transaction Filter Definitions*, elsewhere in this guide.

The transaction filter definitions are listed on the List of Transaction Filters screen.

2 Locate the definition you want to copy on the screen and enter a **C** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

A dialog appears requesting a name for the copy of the transaction filter definition.

```
Enter new name: ________
or press PF3 to cancel
                              ↩
```

3 Specify a new, unique name for the copy of the transaction filter definition and press Enter.

The transaction filter definition is copied and the copy appears on the List of Transaction Filters screen.

Deleting Transaction Filter Definitions

To use the Adabas Event Replicator Subsystem to delete a transaction filter definition in the Replicator system file:

1 List the transaction filter definitions in the Adabas Event Replicator Subsystem, as described in *Listing Transaction Filter Definitions*, elsewhere in this guide.

The transaction filter definitions are listed on the List of Transaction Filters screen.

2 Locate the definition you want to delete on the screen and enter a **D** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

Note: If you want to delete a filter condition from the transaction filter definition, read *Modifying Transaction Filter Definitions* elsewhere in this section.

The transaction filter definition is deleted.

Rules for Writing Filter Conditions

There are various things you should consider when creating filter conditions. This section describes them.

So My Record Matches the Filter Conditions -- Now What?

Filter conditions are based on the values of fields (or partial fields) in an SFILE record definition. If a field or partial field meets all of the filter conditions specified, the record is selected. Once selected, the record will be either included or excluded from replication processing, based on what the transaction filter definition specifies. So selection of a record does not necessarily mean that it will be replicated -- merely that it passed the filter conditions specified by the transaction filter definition. If the transaction filter definition indicates that selected records should be excluded from replication, the record will not be replicated.

Transaction filter definitions indicate whether selected records are replicated or not via the `FRECORDS` initialization parameter in the DDKARTE statements of the Event Replicator Server startup job or via the **Exclude or Include Records** field on the Transaction Filter screen of the Adabas Event Replicator Subsystem.

Note: Include and exclude processing function in the same way for partial fields as for complete fields in your transaction filters.

Failed or Ignored Filter Conditions

A filter condition will be ignored if it cannot be evaluated. This can occur if the image to be tested (set by the `FSIMAGE` or `FTIMAGE` parameters) is not present for replication. The effect of this on filter processing varies, based on whether the filter occurs as part of include or exclude processing and, if it is included in a group of conditions, how the other conditions in the group are matched, failed, or ignored. This is best explained in a series of examples.

Note: The following four examples using an add command (N1) are also true for an initial-state record since an initial-state record contains only an after image. No before image is present for an initial-state record.

1. Suppose an add command (N1) adds a record containing field AB to which the following filter is applied:

```
FRECORDS=INCLUDE
  FFIELD='AB',FSIMAGE=BI
    FCOND=EQ
    FLIST='1916'
```

   In this case, the filter cannot be evaluated because only the after image is present for an add and the filter is for the before image (FSIMAGE=BI). So this filter is ignored and no test is done on the field to see if the before image is equal to "1916". Consequently the add transaction is not included in replication.

2. Likewise, a similar exclude filter is also ignored:

```
FRECORDS=EXCLUDE
  FFIELD='AB',FSIMAGE=BI
    FCOND=EQ
    FLIST='1916'
```

   In this case, the filter cannot be evaluated because only the after image is present for an add and the filter is for the before image (FSIMAGE=BI). So this filter is ignored and no test is done on the field to see if the before image is equal to "1916". However, because this is an exclude filter, the add transaction is not *excluded* from replication; in other words, it is included in replication, regardless of whether or not the before image of the AB field was equal to "1916".

3. Now consider the following transaction filter using **multiple filter conditions** and include processing:

```
FILTER NAME=MYINCLF
  FRECORDS=INCLUDE
    FFIELD='BA',FSIMAGE=BI,FCOND=EQ,FLIST='AAAA'
    FFIELD='BB',FSIMAGE=AI,FCOND=EQ,FLIST='VVVV'
    FFIELD='BC',FSIMAGE=AI,FCOND=EQ,FLIST='XXXX'
```

   If an add command (N1) is issued for a record containing the BA, BB, and BC fields, no before image is present for these fields -- only the after image. Therefore, the filter condition for BA is ignored because the filter is for the before image; the BA filter condition is treated as if it is not even specified. The add transaction, then, is only *included* in replication if both filters for fields BB and BC are true.

4. Finally, consider the following transaction filter using **multiple filter conditions**, exclude processing, and OR processing:

```
FILTER NAME=MYEXCLF
  FRECORDS=EXCLUDE
    FFIELD='BA',FSIMAGE=BI,FCOND=EQ,FLIST='AAAA'
    FFIELD='BB',FSIMAGE=AI,FCOND=EQ,FLIST='VVVV'
  OR
    FFIELD='CA',FSIMAGE=AI,FCOND=EQ,FLIST='EEEE'
    FFIELD='CB',FSIMAGE=AI,FCOND=EQ,FLIST='CCCC'
  OR
    FFIELD='DA',FSIMAGE=BI,FCOND=EQ,FLIST='OOOO'
    FFIELD='DB',FSIMAGE=BI,FCOND=EQ,FLIST='CCCC'
```

If an add command (N1) is issued for a record containing these fields, no before image is present for these fields -- only the after image. Therefore, the filter conditions for BA, DA, and DB are ignored because the filters are for the before image; these filter conditions are treated as if they are not even specified.

The add transaction, then, is only *excluded* in replication if the filter for BB is satisfied OR if both the filters for field CA and field CB are satisfied. Otherwise, the add transaction is included in replication.

Target (FLIST Parameter) Value Syntax

Target (`FLIST` parameter) values are the values to be compared to the source field (`FFIELD` parameter) using the condition type specified (`FCOND` parameter). When multiple values are being compared for a field, they must be specified in a comma-separated list.

Each value can be expressed in one of two ways:

- You can specify values as free-format text. This text can be any set of alphanumeric set of characters. If blanks are required in the value, you should enclose the value in single quotes.

  When the data in the text is all numeric with an optional leading "+" or "-" sign, it is flagged as a numeric value and will be handled differently depending on the source field type in the Event Replicator Server definitions.
- You can specify values as a combination of `A()` and `X()` constructs that enable you to enter data for the same variable in alphabetic format, hexadecimal format, or both, as required. If the element value starts with the string "A(" or "X(" it is treated as an `A()` or `X()` value. If the value does not start with one of these strings, the value is treated as free-format text.

This section describes rules specific to these different methods of specifying target values.

- Free-Format Value Rules
- A() and X(0) Format Value Rules

- Examples

Free-Format Value Rules

The following rules apply to free-format values.

- Free-format values can be any sequence of alphanumeric data apart from the comma character itself.
- If a blank is required for the free-format value, specify the value in single quotes.
- If an apostrophe is required as part of a free-format value, double the apostrophe (for example,'six o''clock').
- If the value consists of all numeric characters with an optional leading "+" or "-" sign, the value will be treated as numeric.
- If the value begins with a single asterisk (*), it is interpreted as a wildcard suffix (for example, '*xyz').
- If the value ends with a single asterisk, it is interpreted as a wildcard prefix (for example, 'abc*').
- If two asterisks are found together (**) in any location in the free-format value, they are interpreted as a single asterisk in the resulting data.
- If a single asterisk is found in the middle of the data, it is rejected as invalid.

Note: The asterisk wildcard can only be used if the condition for the filter expression is EQ (equal) or NE (not equal). They cannot be used for any other types of filter expression conditions.

A() and X(0) Format Value Rules

The following rules apply to A() and X() value specifications.

- The `A()` construct is specified using the following syntax:

```
A(data)
```

  In this syntax, the *data* specified can be any alphanumeric characters, except the parentheses characters.

- The `X()` construct is specified using the following syntax:

```
X(data)
```

  In this syntax, the *data* specified must be an even number of characters in the range X'F0' to X'F9' (i.e. 0 to 9) and x'C1' to X'C6' (i.e. A to F). Each pair of characters will represent the hexadecimal value for one byte in the resultant value.

- If a value starts with an `A()` or `X()` construct, the entire value must be specified using these constructs. You cannot mix them with free-format values.

- `A()` and `X()` constructs can be specified multiple times in the same value specification. They must always have matching opening and closing parentheses, or the entire value specification is treated as invalid.
- When the `A()` construct is used, the asterisk (*) wildcard character is treated in the same manner as for free-format values.
- When the `X()` construct is used, the X'5C' character (which represents an asterisk) is treated like any other hexadecimal character and is not interpreted as a wildcard.

Examples

In the following example, an FLIST value of "ABCDE" is specified:

```
FLIST='ABCDE'
```

In the following example, a numeric FLIST value of "12345" is specified:

```
FLIST='12345'
```

In the following example, a numeric FLIST value of "-678" is specified:

```
FLIST='-678'
```

In the following example, an FLIST value of "AB123" is specified:

```
FLIST='AB123'
```

In the following example, an FLIST value of "XyZ" is specified:

```
FLIST='A(XyZ)'
```

In the following example, an FLIST value of "SSS" (the alphabetic equivalent of X'E2E2E2') is specified:

```
FLIST='X(E2E2E2)'
```

In the following example, an FLIST value of "abc<<<def" is specified:

```
FLIST='A(abc)X(4C4C4C)A(def)'
```

In the following example, an FLIST value of "AX(E2E2E2)" is specified:

```
FLIST='AX(E2E2E2)'
```

In the following example, an FLIST value of "1A(BCD)" is specified:

```
FLIST='1A(BCD)'
```

In the following example, an FLIST value of "1A(BCD)" is specified:

```
FLIST='1A(BCD)'
```

In the following example, an FLIST value of "*abc*" ("abc" is the alphabetic equivalent of X'C1C2C3') is specified. Note that this FLIST value is open-ended because wildcards are specified:

```
FLIST='A(*)X(C1C2C3)A(*)'
```

In the following example, an FLIST value of "*def**" is specified. Note that the first asterisk specifies a wildcard, but the last two asterisks specify asterisk characters (the alphabetic equivalent of X'5C5C'):

```
FLIST='A(*def)X(5C5C)'
```

The following examples are invalid because they specify a wildcard asterisk in the middle of the values:

```
FLIST='*ABC*DEF*'
FLIST='X(F5F6)*X(F7F8)'
FLIST='X(F2)A(*)X(F4)'
```

The following examples are invalid because they specify invalid hexadecimal data:

```
FLIST='X(ABACFGZZAE)'
FLIST='X(ABC)'
```

The following example is invalid because it mixes free-format and hexadecimal data:

```
FLIST='X(AB)AB'
```

The following example is invalid because it misuses commas:

```
FLIST='X(ABAC),,A(123)'
```

The following example is invalid because it misuses parentheses in the A() construct:

```
FLIST='A(12(34))'
```

When You Can Specify Multiple Targets

You can only specify multiple targets if the condition operator is EQ (equal) or NE (not equal). The LT (less than), LE (less than or equal), GT (greater than), and GE (greater than or equal) operators logically assume a comparison of the field value to a single target value, so multiple target values are not allowed for these condition operators.

Since wildcards are essentially a concise way of specifying multiple targets, you can also only use wildcards when the condition operator is EQ or NE.

If your filter checks to see if the field value is equal to a list of target values, the field value need only be equivalent to *one* of the target values for the filter condition to be true. On the other hand, if your filter checks to see if the field value is not equal to a list of target values, the field value must not be equal to *any* of the target values for the filter condition to be true.

Examples

In the following example, records for which the after image of the AA field is equal to "1", "2", "3", or "4" are selected.

```
FFIELD='AA',FSIMAGE=AI
FCOND=EQ
FLIST='1,2,3,4'
```

In the following example, records for which the after image of the AA field is greater than "5" are selected.

```
FFIELD='AA',FSIMAGE=AI
FCOND=GT
FLIST='5'
```

In the following example, records for which the first three bytes of the after image of the BB field contain the characters "abc" are selected.

```
FFIELD='BB',FSIMAGE=AI
FCOND=EQ
FLIST='abc*'
```

In the following example, records for which the last three bytes of the after image of the BB field contain the characters "xyz" are selected.

```
FFIELD=’BB’,FSIMAGE=AI
FCOND=EQ
FLIST=’*xyz’
```

In the following example, records in which *no* bytes of the after image of the BB field contain the characters "klm" are selected.

```
FFIELD=’BB’,FSIMAGE=AI
FCOND=NE
FLIST=’*klm*’
```

The following example is invalid because it specifies multiple FLIST target values when the condition code is not EQ or NE.

```
FFIELD=’AA’,FSIMAGE=AI
FCOND=LE
FLIST=’1,2,3,4’
```

The following example is invalid because it specifies a wildcard in the FLIST target value when the condition code is not EQ or NE.

```
FFIELD=’AA’,FSIMAGE=AI
FCOND=GT
FLIST=’*xyz’
```

How Multiple Filter Conditions Are Interpreted

You can specify multiple filter conditions within a single transaction filter definition. Unless otherwise grouped, all of the specified filter conditions must be true for a record to be selected. In other words, the filter conditions are logically ANDed. In the following example, cond1, cond2, cond3, and cond4 must all be true for the record to be selected as they are logically ANDed:

```
FILTER NAME=MYINCLF
FRECORDS=INCLUDE
FFIELD='field',FCOND=cond1,FTARGET or FLIST values1
FFIELD='field',FCOND=cond2,FTARGET or FLIST values2
FFIELD='field',FCOND=cond3,FTARGET or FLIST values3
FFIELD='field',FCOND=cond4,FTARGET or FLIST values4
```

If, however, you want to insert some logical ORs in this example, you can. To do this you would use the OR keyword in the DDKARTE statements of the Event Replicator Server startup job or use the **Group** field on the Filter Condition screen in the Adabas Event Replicator Subsystem. As an example of using the OR keyword, consider the following modification to the example given earlier.

```
FILTER NAME=MYINCLF
FRECORDS=INCLUDE
FFIELD='field',FCOND=cond1,FTARGET or FLIST values1
FFIELD='field',FCOND=cond2,FTARGET or FLIST values2
FFIELD='field',FCOND=cond3,FTARGET or FLIST values3
OR
FFIELD='field',FCOND=cond4,FTARGET or FLIST values4
```

In this example, condition1, condition2, and condition3 must be true OR condition 4 must true for the record to be selected.

When using the **Group** field on the Filter Condition screen of the Adabas Event Replicator Subsystem to define your transaction filter definitions, simply use the same group number for those conditions you want ANDed. Conditions with different group numbers are logically ORed.

Specifying a Range of Values

You can specify a range of values in your filter condition by creating two conditions that are logically ANDed (read ***How Multiple Filter Conditions Are Interpreted***). Simply define one filter condition to test for values greater than (GT) or greater than or equal to (GE) the lowermost value. Then define the second filter condition to test for values less than (LT) or less than or equal to (LE) the uppermost value. As both conditions must be true since they are logically ANDed, your range specification is assured.

Field Type Considerations

Ideally, when a field is compared to another field, the field types will be the same. However, it is possible to compare fields of different formats. For example, you can compare a packed decimal format field with a binary format field. For a complete list of compatible Adabas field types, refer to your Adabas documentation.

This section covers the following topics related to how fields of different formats are compared

- Valid Comparison Table
- Comparison Processing by Field Type

- UES Considerations

Valid Comparison Table

An asterisk (*) in a cell in the following table indicates that a comparison of the field types is valid. A blank in a cell in the table indicates that a comparison is not supported.

Field Data Type	Alphanumeric	Unpacked	Packed	Binary	Floating Point	Wide-Character	Fixed Point
Alphanumeric	*			*		*	
Unpacked		*	*	*	*		*
Packed		*	*	*	*		*
Binary	*	*	*	*	*		*
Floating Point		*	*	*	*		*
Wide-Character	*					*	
Fixed Point		*	*	*	*		*

Comparison Processing by Field Type

When either the source or target field is of type floating point (but not both fields), the other field will be converted to floating point, and a floating point comparison will be made. SARC settings governing byte-swapping and floating point type (HFP, IEEEfloat, and VAXfloat) are honored.

Note: The conversion of very large numbers in a numeric format other than floating point to floating may result in a loss of precision because as the numbers get bigger, the range of numbers that may be represented in the floating point format is reduced. For example, the value 99,999,999,999,999,999 will be converted to the floating point value 99,999,999,999,999,984.

In all other cases the following conversions and comparisons will apply:

Source Field Data Type	Comparison Processing Notes
Unpacked	The source and target fields are converted to packed form for comparison.
Binary	Prior to comparison, the SARC byte order setting is honored for binary source and target fields. Packed and unpacked target fields are converted to binary and then compared. An alphanumeric target field is compared as is.
Packed	Prior to comparison, target fields of type fixed, unpacked, or binary are converted to packed.
Fixed	When the target field is binary, the source is converted to binary and then compared. When the target field is packed, the source field is converted to packed and then compared. When the target field is unpacked, both the source and target fields are converted to packed and then compared. When the target field is fixed, a direct comparison is made between the source and target fields (no conversion is necessary).

When a field is compared to a list of target values, the target values are converted (if they are not the same) to the data type of the source field, once the source field type is determined. This can cause problems in the accuracy of filter condition processing if a target value in the list cannot be converted or is otherwise incompatible with the required source field type. So target values and target fields must be specified carefully to avoid such problems.

Target list values entered as alphanumeric are converted to the data type of the source field, honoring the SARC, SACODE and SWCODE parameter settings.

Target list values for alphanumeric fields may be entered as alphanumeric, hexadecimal, or a mixture of both. If it is a mixture of both -- for example, FLIST=A(ABC)X(C4C5C6)A(GHI) -- it is treated as an alphanumeric field even though some of it is specified as hexadecimal.

Target list values for binary fields may be entered in hexadecimal. The hexadecimal values are assumed to be in a form that honors the SARC parameter settings.

Target list values for floating point fields may be entered in hexadecimal. The hexadecimal values are assumed to be in a form that honors the SARC parameter settings such as floating-point format and byte order.

Target list values may not be entered in hexadecimal for zoned decimal, packed decimal and fixed point fields.

UES Considerations

When a field is compared to a target value that is entered in hexadecimal, the target value is normally accepted without any conversion. It is assumed that you have taken into account the settings of the SARC, SACODE and SWCODE parameters when constructing the hexadecimal value. It is important to remember that a given hexadecimal value may have to reflect the setting of more than one of these three parameters.

- The SARC parameter defines special data architecture for fields in the record and value buffers (see the description of record buffers in your Adabas documentation). If the byte order bit of the SARC value is set, the hexadecimal value may have to be entered with low-order bytes first. If the field is a character field, the entered hexadecimal byte values must reflect the setting of the SARC encoding family bit.

  Note: If you want to transfer replicated and initial-state data to a relational database using the Event Replicator Target Adapter (the Destination Class, or DCLASS parameter, is set to "SAGTARG"), set the SARC parameter to "2" -- regardless of the location of your relational database.

- The SACODE parameter assigns special encoding for alphanumeric fields during the user session. Hexadecimal bytes values must reflect the SACODE setting.
- The SWCODE assigns special encoding for wide-character fields during the user session. Hexadecimal values must reflect the SWCODE setting.

This section covers the following topics related to UES processing:

- Internal Handling of UES Settings
- Examples Honoring UES Settings

Internal Handling of UES Settings

When FILTER FLIST value parameters are being processed, the UES settings and the FCOND values are taken into account in an attempt to minimize conversion overhead at runtime.

- If the FCOND setting is either EQ or NE then the FLIST value will be stored as entered. At runtime, this FLIST value can be compared directly with the field value with no conversion required regardless of the UES settings.
- If the FCOND setting is LT, GT, LE, or GE, the FLIST value will be converted, taking into account the UES settings, so that valid comparisons can be made for the specified FCOND value. At runtime, the field value will be similarly converted to facilitate valid comparisons.

It is important to note here that FLIST values that are entered as hexadecimal values for comparison with binary fields must be entered in a form that honors the SARC settings. In other words, they must be specified in the same form as the field is stored in the record buffer. Similarly wide-character field values entered as hexadecimal must be specified in the same form as the field is stored in the record buffer.

FLIST values that are entered as EBCDIC text, or as numbers, will be converted appropriately.

Examples Honoring UES Settings

Field Data Type	Examples
Floating Point	In the following example, an FLIST value of a short HFP floating point value of 1.0 is specified in hexadecimal: `FLIST='X(41100000)'` Regardless of the setting of the SARC byte order bit the hexadecimal value for a HFP floating point value will remain the same because HFP floating point does not honor byte swapping. In the following example, an FLIST value of a long HFP floating point value of 50,000.0 is specified in hexadecimal:

Field Data Type	Examples

	`FLIST='X(44C3500000000000)'`
	
	In the following example, an `FLIST` value of a short VAX floating point value of 1.0 is specified in hexadecimal:
	
	`FLIST='X(40800000)'`
	
	This value is valid when the `SARC` byte order bit is not set. If the SARC order bit is set, the following hexadecimal value must be entered for a value of 1.0:
	
	`FLIST='X(80400000)'`
	
	Note that the bytes are swapped in pairs.
	
	In the following example, an `FLIST` value of a long VAX floating point value of 50,000.0 is specified in hexadecimal:
	
	`FLIST='X(4843500000000000)'`
	
	This value is valid when the `SARC` byte order bit is not set. If the `SARC` order bit is set, the following hexadecimal value must be entered for a value of 50,000.0:
	
	`FLIST='X(4348005000000000)'`
	
	Note that the bytes are swapped in pairs.

Wide-character	In the following example, an `FLIST` value of the wide-character string '`ABB ABCDEF`' in WCODE=4095 is specified in hexadecimal:
	
	`FLIST='X(00410042004200400041004200430044004500460)'`
	
	In the following example, an `FLIST` value of the wide-character string '`ABB ABCDEF`' in WCODE=4095 with the `SARC` byte order bit turned on, is specified in hexadecimal:
	
	`FLIST='X(41004200420040004100420043004400450046 00)'`
	
	Note that the bytes are swapped in pairs.

Binary	In the following example, an `FLIST` value of the binary value of decimal 4 with the `SARC` byte order bit turned off, is specified in hexadecimal: `FLIST='X(000004)'` In the following example, an `FLIST` value of the binary value of decimal 4 with the `SARC` byte order bit turned on, is specified in hexadecimal:

Field Data Type	Examples
	`FLIST='X(040000)'` Note that the byte order is reversed. In other words, the first byte becomes the last byte, the second byte becomes the second-to-last byte, and so on.

Varying Field Length Considerations

When the length of the source field and target field are different, the shorter value is converted to the size of the longer value. For alphanumeric data, the value is padded on the right with blanks. For numeric data, the value is padded on the left with hexadecimal zeros.

Using Wildcards

You can use an asterisk (*) as a wildcard for target values if the condition code being used is EQ (equal) or NE (not equal). You cannot use wildcard characters for any other filter conditions (GT, LT, LE, or GE).

Note: Wildcard values are not supported for wide character fields.

- If you want to test the field for any value beginning with a specific string of characters, simply append an asterisk to the end of the value. For example, to test for a field value starting with the characters "POW", specify "POW*" as the target value.
- If you want to test the field for the occurrence of as specific string within its value, precede and supercede the string with an asterisk. For example, to test for the occurrence of the string "WER", specify "*WER*" as the target value.
- If you need to test for the occurrence of an asterisk itself in a field value, specify two asterisks in a row for the target value ("**").

8 Maintaining Subscription Definitions

A subscription definition defines a set of rules to be applied to replicated data. Subscription definitions include at least one SFILE definition and at least one destination definition. They may also specify a resend buffer definition to expedite the retransmission of a transaction.

At least one subscription definition, with its associated destination and SFILE definitions, must be created because these definitions are used to determine how replicated data is processed by the Event Replicator Server. If a subscription definition is not specified, data replication will occur, but the data will never be processed by the Event Replicator Server and, therefore, will never be delivered to the target application.

Listing Subscription Definitions

To use the Adabas Event Replicator Subsystem to list the subscription definitions stored in the Replicator system file:

- Select option **S** from the Adabas Event Replicator Subsystem Main Menu.

  The Available Subscriptions screen appears showing all of the subscription definitions in the Adabas Event Replicator Subsystem.

```
18:32:30      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                          Available Subscriptions                    M-RP1400

               Sel  Name      Ver   Description
               ---  --------  ---   ---------------------------------
                _   SI040155  OC    INITIAL STATE 040 155
                _   SI042060  OC    INITIAL STATE 042 060
                _   SI042200  OC    INITIAL STATE 042 200
                _   SI046088  OC    INITIAL STATE 046 088
                _   SI062026  OC    INITIAL STATE 062 026
                _   SI062029  OC    INITIAL STATE 062 029
                _   SI062035  OC    INITIAL STATE 062 035
                _   SI062055  OC    INITIAL STATE 062 055
                _   SI062079  OC    INITIAL STATE 062 079
                _   SI062106  OC    INITIAL STATE 062 106
                _   SI062143  OC    INITIAL STATE 062 143
                _   SI064121  OC    INITIAL STATE 064 121



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Repos Exit  Add               -     +                     Menu  ↩
 ↩
```

The function keys on this screen perform the following functions:

Function Key	Description
PF1/F1 (Help)	Provides you with help for this screen.
PF2/F2 (Repos)	Provides you with a pop-up panel that allows you to specify the name of the definition you want to locate in the list. Once you have specified a name on the pop-up panel and pressed Enter, the list is repositioned so the name you selected appears first. You can use an asterisk as a wild card character at the end of the definition name or partial definition name you specify on the pop-up panel. Or, you can simply enter the first few characters of the name to reposition the list to the first occurrence in the list of a name starting with those characters.
PF3/F3 (Exit)	Returns you to the previous screen.
PF4/F4 (Add)	Allows you to add a new definition. A new screen appears.
PF7/F7 (-)	Allows you to scroll backwards through the list of definitions.
PF8/F8 (+)	Allows you to scroll forwards through the list of definitions.
PF12/F12 (Menu)	Returns you to the main menu.

Adding Subscription Definitions

To use the Adabas Event Replicator Subsystem to add a subscription definition in the Replicator system file, complete the following steps:

- Step 1. Access the Subscription Definition Area of the Adabas Event Replicator Subsystem
- Step 2. Supply General Subscription Information
- Step 3. Specify One or More Destinations for the Subscription
- Step 4. Specify One or More SFILE Definitions for the Subscription
- Step 5. (Optional) Modify the Transaction Logging Values for the Subscription, as Necessary
- Step 6. Save the Subscription Definition

Step 1. Access the Subscription Definition Area of the Adabas Event Replicator Subsystem

To access the subscription definition are of the Adabas Event Replicator Subsystem:

1 Select option **S** from the Adabas Event Replicator Subsystem Main Menu.

The Available Subscriptions screen appears showing all of the subscription definitions in the Adabas Event Replicator Subsystem.

```
18:32:30      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                          Available Subscriptions                   M-RP1400

               Sel  Name      Ver   Description
               ---  --------  ---   ---------------------------------
                _   SI040155  OC    INITIAL STATE 040 155
                _   SI042060  OC    INITIAL STATE 042 060
                _   SI042200  OC    INITIAL STATE 042 200
                _   SI046088  OC    INITIAL STATE 046 088
                _   SI062026  OC    INITIAL STATE 062 026
                _   SI062029  OC    INITIAL STATE 062 029
                _   SI062035  OC    INITIAL STATE 062 035
                _   SI062055  OC    INITIAL STATE 062 055
                _   SI062079  OC    INITIAL STATE 062 079
                _   SI062106  OC    INITIAL STATE 062 106
                _   SI062143  OC    INITIAL STATE 062 143
                _   SI064121  OC    INITIAL STATE 064 121



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Repos Exit  Add               -     +                     Menu  ↩
 ↩
```

2 Press PF4 on the Available Subscriptions screen.

The Subscription Definition screen appears showing all of the subscription definitions in the Adabas Event Replicator Subsystem.

Note: The list of incomplete items on this screen identifies information (definitions) that must be supplied for the subscription definition before it can be read at Event Replicator Server startup. In the example below, the new subscription definition still needs the following information before it is a valid subscription: at least one destination definition, at least one SFILE definition (File-Related Parms), and format buffer specifications (either as individual format buffers or using global format buffer (GFB) definitions).

```
18:33:00      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                        Subscription Definition                      M-RP1410

Description ................. ____________________________________

Subscription Name ............ ________             TLOG Values
User Data Alpha Key .......... ___0                 ------------------------
Architecture Key ............. ___2                 Input Level .......... 0
Subscription Version ......... __                   Filter Level ......... 0
User Data Wide Key ........... ___0                 Output Level ......... 0
Resend Buffer Name ........... ________             Filter Matched ....... 0
                                                    Filter Not Matched ... 0
Destination Name List ........ _                    Filter Ignored ....... 0
File-related Parameters ...... _
                                                    Incomplete Item(s)
Subscription Active .......... Y                    ------------------
Deactivate if file deactivated  Y                   Destination
Increment Initial State Count   N                   File-Related Parms
                                                    Format Buffer


Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save                                    Menu  ↩
 ↩
```

Step 2. Supply General Subscription Information

To supply general information for the subscription definition:

- Supply values for the following fields:

Parameter Name	Specify	Default
Description	A description of the subscription. While required by the Adabas Event Replicator Subsystem, this information is for your reference only; it is not used by the Event Replicator Server.	----
Subscription Name	A unique name for the subscription. Subscription names must be between one and eight characters long and are required. This is the equivalent of specifying the `SUBSCRIPTION NAME` parameter directly in the Event Replicator Server startup job.	---
User Data Alpha Key	An appropriate output-alpha code if the Event Replicator Server has been started with Universal Encoding Support (UES) enabled. If UES has not been enabled, leave this field blank or set to zero (0). A value of zero means that no UES translation will be done. This is an optional field. This is the equivalent of specifying the `SACODE` parameter directly in the Event Replicator Server startup job.	0

Parameter Name	Specify	Default
	XML data sent to Event Replicator Target Adapter is in EBCDIC character encoding. EBCDIC values for alphanumeric characters X'40' thru X'FF' are sent unchanged. EBCDIC values for alphanumeric fields under X'40' are translated to spaces. However, support for UTF-8 encoding is available for internationalization purposes. To send XML messages to Event Replicator Target Adapter using UTF-8 format, make sure that this parameter is set to "4091" in the Event Replicator subscription definition used for Event Replicator Target Adapter processing. **Note:** If you want to use UTF-8 character encoding (i.e. SACODE=4091), you must verify that your field lengths are increased as required to accommodate UTF-8 character encoding. The Adabas UES code pages 37, 424, 813, 912, 915, 920, 922, 923, 1006, 1112, 1140, 1256, and 4091 can be specified in the SACODE parameter for Event Replicator Target Adapter.	
Architecture Key	An appropriate architecture key if the Event Replicator Server has been started with Universal Encoding Support (UES) enabled. If UES has not been enabled, leave this field set to "2". This is an optional field. The architecture key is an integer that is calculated as the sum of the following numbers (this is the same as is documented in the record buffer section of the OP command in the Adabas Command Reference Guide): ■ byte order: 0 (high-order byte first) or 1 (low-order byte first) ■ encoding family: 0 (ASCII) or 2 (EBCDIC) ■ floating point format: 0 (IBM370), 4 (VAX), or 8 (IEEE). This is the equivalent of specifying the `SARC` parameter directly in the Event Replicator Server startup job. **Note:** If you want to transfer replicated and initial-state data to a relational database using the Event Replicator Target Adapter (the Destination Class, or `DCLASS` parameter, is set to "SAGTARG"), set the SARC parameter to "2" -- regardless of the location of your relational database.	2
Subscription Version	A subscription version. Up to two characters can be specified. This is the equivalent of specifying the `SVERSION` parameter directly in the Event Replicator Server startup job. This field has no meaning to the Event Replicator Server -- it is optional. But it may be useful for the target application when handling changes in the subscription definition. It is passed as part of the header information sent to the target.	---
User Data Wide Key	An appropriate output-wide code if the Event Replicator Server has been started with Universal Encoding Support (UES) enabled. If UES has not been enabled, leave this field blank. This field is optional. This is the equivalent of specifying the `SWCODE` parameter directly in the Event Replicator Server startup job.	---

Parameter Name	Specify	Default
	Wide-character fields sent to Event Replicator Target Adapter are translated to hexadecimal. However, support for UTF-8 encoding is available for internationalization purposes. To send wide-character fields to Event Replicator Target Adapter using UTF-8 format, make sure that this parameter is set to "4091" in the Event Replicator subscription definition used for Event Replicator Target Adapter processing.	
Resend Buffer Name	The name of the resend buffer to be associated with the subscription, if any. This field is optional. For more information about resend buffer definitions, read *Maintaining Resend Buffer Definitions*, elsewhere in this guide. This is the equivalent of specifying the `SRESENDBUFFER` parameter in the Event Replicator Server startup job.	---
Subscription Active	Whether or not this subscription definition should be activated for use once it is loaded by the Event Replicator Server. Valid values are Y (load and activate the definition) or N (load, but do not activate the definition). This field is optional. This is the equivalent of specifying the `SACTIVE` parameter in the Event Replicator Server startup job.	Y
Deactivate if file deactivated	Whether or not this subscription definition should be deactivated if one of its files is deactivated. Valid values are "Y" (deactivate the subscription) or "N" (do not deactivate the subscription). This is the equivalent of specifying the `SDEACTIVATE parameter` in the Event Replicator Server startup job. While this is an optional feature, we recommend that you use the default or specify "Y" to ensure that the integrity of your replicated data is maintained when a file becomes deactivated. If the subscription is deactivated because one of its files is deactivated, any other files referenced by this subscription are also deactivated unless they are also referenced by another active subscription. If replication for this file in the subscription is unrelated to the replication for other files in the same subscription, the integrity of the replicated data may not be at risk. In this case, a value of "N" might be specified.	Y
Increment Initial State Count	Whether or not the transaction sequence number should be incremented if an initial-state request is occurring. Valid values are 'Y' (increment the transaction sequence number) or 'N' (do not increment the transaction sequence number). This field is optional. This is the equivalent of specifying the `SINCREMENTIS` parameter directly in the Event Replicator Server startup job.	N

Step 3. Specify One or More Destinations for the Subscription

Specify an "S" in the **Destination Name List** field to specify destination definitions for the subscription. When you enter an "S", a pop-up menu will appear prompting you to save the subscription, and then the **Destination List** screen appears listing the destination definitions currently assigned to the subscription definition. At least one destination definition must be specified on this screen:

- Use the **Name** fields on the **Destination List** screen to specify the name of a predefined destination definition. For more information about destination definitions, read *Maintaining Destination Definitions*, elsewhere in this guide.

  Notes:

  1. You can use PF6 to pick a destination from a list of your previously-defined destination definitions.
  2. An Adabas destination can be referenced by no more than one subscription.

- Once you have specified a destination definition name, use the **N** and **I** columns to indicate whether you want the destination to receive normal (N) data or initial-state (I) replicated data. Valid values for these columns are "Y" (receive the data) or "N" (do not receive the data). If you leave these blank, the default for both is "Y".

  Note: Only initial-state (I) replicated data is valid for subscriptions that use an Optimized Global Format Buffer, therefore Column (N) for normal data should be set to a value of "N".

- Press PF5 to save your destination list when it is finished. Then press PF3 to return to the Subscription Definition screen.

Step 4. Specify One or More SFILE Definitions for the Subscription

Specify an "S" in the **File-related Parameters** field to create the SFILE definitions for the subscription. When you enter an "S", a pop-up menu will appear prompting you to save the subscription, and then the List of Subscription SFILEs screen appears listing the SFILE definitions currently assigned to the subscription definition. At least one SFILE definition must be specified on the List of Subscription SFILEs screen:

- If the SFILE definition you want included in the subscription is not listed, press PF4 to add one. If you want to alter an SFILE definition in this list, enter an "S" in the **Sel** column next to the SFILE definition. For more information about maintaining SFILE definitions, read *Maintaining SFILE Definitions*, elsewhere in this guide.
- Once all of the SFILE definitions you want included in this subscription are listed on the List of Subscription SFILEs, press PF3 to return to rest of the subscription definition.

Step 5. (Optional) Modify the Transaction Logging Values for the Subscription, as Necessary

To modify the TLOG values for the subscription:

- Optionally, modify the following transaction logging (TLOG) fields on the **Subscription Definition** screen.

Parameter Name	Specify	Default
Input Level	The transaction logging level when a transaction is selected for subscription processing. Valid values are "0" (no logging), "1" (log event and input transaction data), "2" (log event, input transaction, and file/record data), or "3" (log event and all available input transaction data for the event). This is the equivalent of specifying the `STLINPUT` parameter in the Event Replicator Server startup job.	0
Filter Level	The transaction logging level when a record in a transaction is being excluded from replication due to extended subscription processing or the subscription user exit. Valid values are "0" (no logging), "1" (log event, filter reason, and transaction information), "2" (log event, filter reason, transaction, and file/record information), or "3" (log event, filter reason, transaction information, file/record information, and payload data of available images). This is the equivalent of specifying the `STLFILTER` parameter in the Event Replicator Server startup job.	0
Output Level	The transaction logging level when a transaction is to be output on behalf of a subscription. Valid values are "0" (no logging), "1" (log event and output transaction data), "2" (log event, output transaction, and file/record data), or "3" (log event and all available output information for the event). This is the equivalent of specifying the `STLOUTPUT` parameter in the Event Replicator Server startup job.	0
Filter Matched	The transaction logging level when a filter condition is true. Valid values are "0" (no logging), "1" (log event and filter information), or "2" (log event, filter information, and payload data or available field values). This is the equivalent of specifying the `STLMATCH` parameter in the Event Replicator Server startup job.	0
Filter Not Matched	The transaction logging level when a filter condition is false. Valid values are "0" (no logging), "1" (log event and filter information), or "2" (log event, filter information, and payload data or available field values). This is the equivalent of specifying the `STLNOMATCH` parameter in the Event Replicator Server startup job.	0
Filter Ignored	The transaction logging level when a filter condition cannot be evaluated. Valid values are "0" (no logging), "1" (log event and filter information), or "2" (log event, filter information, and payload data or available field values). This is the equivalent of specifying the `STLIGNORE` parameter in the Event Replicator Server startup job.	0

Step 6. Save the Subscription Definition

To save the subscription definition:

- Press PF5 to save the subscription definition in the Replicator system file.

Modifying Subscription Definitions

To use the Adabas Event Replicator Subsystem to modify a subscription definition in the Replicator system file:

1 Select option **S** from the Adabas Event Replicator Subsystem Main Menu.

The Available Subscriptions screen appears showing all of the subscription definitions in the Adabas Event Replicator Subsystem.

2 Locate the definition you want to modify on the screen and enter an **M** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

You are prompted to select the version of the subscription definition containing the SFILE definitions. The following is a sample pop-up menu from which you can select the version.

```
Please Choose Version

  Code  Version
  ----  ------------
   C    Current
   S    Scheduled
  ----  ------------


  Code ... _
                          ↩
```

3 Select the version by entering the appropriate code in the **Code** field.

The Subscription Definition screen appears listing the details of the Adabas subscription. For information on modifying this screen, read the description of *Adding Subscription Definitions*, elsewhere in this section.

4 When all modifications have been made, press PF5 to save the changes.

You are prompted to indicate which version of the subscription the changes should be saved to. The following is a sample pop-up menu from which you can select the version.

```
         Please Choose Save Option

Code      Description
----  ------------------------------------
 A    Replace Current
 B    Save as Current (Current becomes Old)
 C    Replace Scheduled
 D    Save as Scheduled (Scheduled become Current)
----  ----------------------------------------


Code ... _
                                        ↩
```

5 Select the version by entering the appropriate code in the **Code** field.

The subscription modification is complete.

Copying Subscription Definitions

To use the Adabas Event Replicator Subsystem to copy a subscription definition in the Replicator system file:

1 List the subscription definitions in the Adabas Event Replicator Subsystem, as described in *Listing Subscription Definitions*, elsewhere in this guide.

The subscription definitions are listed on the Available Subscriptions screen.

2 Locate the definition you want to copy on the screen and enter a **C** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

A dialog appears requesting a name for the copy of the subscription definition.

```
Enter new name: ________
or press PF3 to cancel
                              ↩
```

3 Specify a new, unique name for the copy of the subscription definition and press Enter.

The subscription definition is copied and the copy appears on the Available Subscriptions screen.

Activating and Deactivating Subscription Definitions

You can use Adabas Online System (AOS) to activate and deactivate subscription definitions. For more information, read *Activating and Deactivating Replication Definitions and Databases*, in the *Event Replicator for Adabas Administration and Operations Guide*

Caution: Be careful when you activate and deactivate replication definitions and databases, especially if replication is ongoing at the time. Whenever you activate or deactivate definitions or databases, you run the risk of altering what data is replicated and how that replication occurs. If the Event Replicator Server receives data from an Adabas database for which it has no active definitions, replication simply does not occur.

Deleting Subscription Definitions

To use the Adabas Event Replicator Subsystem to delete a subscription definition in the Replicator system file:

1 List the subscription definitions in the Adabas Event Replicator Subsystem, as described in *Listing Subscription Definitions*, elsewhere in this guide.

The subscription definitions are listed on the Available Subscriptions screen.

2 Locate the definition you want to delete on the screen and enter a **D** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

The subscription definition is deleted.

9 Maintaining SFILE Definitions

An SFILE definition defines the Adabas file containing the input data to be processed by the subscription. SFILE definitions are included within subscription definitions. They include format buffer specifications for the file data (they can reference GFB definitions instead). SFILE definitions may also specify a transaction filter definition that should be applied to the data in the SFILE file and a subscription user exit that should be used in processing the file data.

At least one SFILE definition is required in a subscription definition for every Event Replicator for Adabas run.

Listing SFILE Definitions

To use the Adabas Event Replicator Subsystem to list the SFILE definitions stored in a subscription:

1 Select option **S** from the Adabas Event Replicator Subsystem Main Menu.

The Available Subscriptions screen appears showing all of the subscription definitions in the Adabas Event Replicator Subsystem.

```
18:32:30      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                          Available Subscriptions                   M-RP1400

               Sel  Name      Ver   Description
               ---  --------  ---   ---------------------------------
                _   SI040155  OC    INITIAL STATE 040 155
                _   SI042060  OC    INITIAL STATE 042 060
                _   SI042200  OC    INITIAL STATE 042 200
                _   SI046088  OC    INITIAL STATE 046 088
                _   SI062026  OC    INITIAL STATE 062 026
                _   SI062029  OC    INITIAL STATE 062 029
                _   SI062035  OC    INITIAL STATE 062 035
                _   SI062055  OC    INITIAL STATE 062 055
                _   SI062079  OC    INITIAL STATE 062 079
                _   SI062106  OC    INITIAL STATE 062 106
                _   SI062143  OC    INITIAL STATE 062 143
                _   SI064121  OC    INITIAL STATE 064 121



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Repos Exit  Add               -     +                     Menu  ↩
 ↩
```

The function keys on this screen perform the following functions:

Function Key	Description
PF1/F1 (Help)	Provides you with help for this screen.
PF2/F2 (Repos)	Provides you with a pop-up panel that allows you to specify the name of the definition you want to locate in the list. Once you have specified a name on the pop-up panel and pressed Enter, the list is repositioned so the name you selected appears first. You can use an asterisk as a wild card character at the end of the definition name or partial definition name you specify on the pop-up panel. Or, you can simply enter the first few characters of the name to reposition the list to the first occurrence in the list of a name starting with those characters.
PF3/F3 (Exit)	Returns you to the previous screen.
PF4/F4 (Add)	Allows you to add a new definition. A new screen appears.
PF7/F7 (-)	Allows you to scroll backwards through the list of definitions.
PF8/F8 (+)	Allows you to scroll forwards through the list of definitions.
PF12/F12 (Menu)	Returns you to the main menu.

2 Enter an **M** in the **Sel** column corresponding to the Adabas subscription definition containing the SFILE definitions you want to list.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

You are prompted to select the version of the subscription definition containing the SFILE definitions. The following is a sample pop-up menu from which you can select the version.

```
Please Choose Version

  Code  Version
  ----  ------------
   C    Current
   S    Scheduled
  ----  ------------


  Code ... _
                          ↩
```

3 Select the version by entering the appropriate code in the **Code** field.

The Subscription Definition screen appears listing the details of the Adabas subscription.

4 Tab down to the **File-related Parameters** field on the Subscription Definition screen and enter an "S".

The List of Subscription SFILEs screen appears listing the SFILE definitions currently defined in the Adabas subscription definition.

```
18:32:30      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                        List of Subscription SFILEs                  M-RP1415
Current
                                                User      BI  AI  KR
            Sel   DBID   File  Ins Upd Del   Exit      FB  FB  FB
            ------------------------------------------------------
             _    1955   1      Y   Y   Y               +   +
             _    1955   2      Y   Y   Y               +   +
             _    1955   3      Y   Y   Y               +   +
             _
             _
             _
             _
             _
             _
             _
             _
             _


Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit  Add               -     +                     Menu  ↩
 ↩
```

Adding SFILE Definitions

To add an SFILE definition to a subscription, complete the steps described in this section:

- Step 1. Access the SFILE Definition Area of a Subscription
- Step 2. Specify SFILE Definition Information

Step 1. Access the SFILE Definition Area of a Subscription

To access the SFILE definition area of a subscription definition:

1 Select option **S** from the Adabas Event Replicator Subsystem Main Menu.

The Available Subscriptions screen appears showing all of the subscription definitions in the Adabas Event Replicator Subsystem.

```
18:32:30      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                              Available Subscriptions                 M-RP1400

               Sel  Name      Ver   Description
               ---  --------  ---   ---------------------------------
                _   SI040155  OC    INITIAL STATE 040 155
                _   SI042060  OC    INITIAL STATE 042 060
                _   SI042200  OC    INITIAL STATE 042 200
                _   SI046088  OC    INITIAL STATE 046 088
                _   SI062026  OC    INITIAL STATE 062 026
                _   SI062029  OC    INITIAL STATE 062 029
                _   SI062035  OC    INITIAL STATE 062 035
                _   SI062055  OC    INITIAL STATE 062 055
                _   SI062079  OC    INITIAL STATE 062 079
                _   SI062106  OC    INITIAL STATE 062 106
                _   SI062143  OC    INITIAL STATE 062 143
                _   SI064121  OC    INITIAL STATE 064 121



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Repos Exit  Add               -     +                     Menu  ↩
 ↩
```

2 Enter an **M** in the **Sel** column corresponding to the Adabas subscription definition to which you want to add an SFILE definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF2 (F2) to specify the name of the definition to which the list should be repositioned.

You are prompted to select the version of the subscription definition containing the SFILE definitions. The following is a sample pop-up menu from which you can select the version.

```
Please Choose Version

  Code  Version
  ----  ------------
   C    Current
   S    Scheduled
  ----  ------------


  Code ... _
                          ↩
```

3 Select the version by entering the appropriate code in the **Code** field.

The Subscription Definition screen appears listing the details of the Adabas subscription.

```
18:33:00      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                        Subscription Definition                     M-RP1410

Description ................. ____________________________________

Subscription Name ............ ________             TLOG Values
User Data Alpha Key .......... ___0                ------------------------
Architecture Key ............. ___2                Input Level .......... 0
Subscription Version ......... __                  Filter Level ......... 0
User Data Wide Key ........... ___0                Output Level ......... 0
Resend Buffer Name ........... ________            Filter Matched ....... 0
                                                   Filter Not Matched ... 0
Destination Name List ........ _                   Filter Ignored ....... 0
File-related Parameters ...... _
                                                   Incomplete Item(s)
Subscription Active .......... Y                   ------------------
Deactivate if file deactivated  Y                  Destination
Increment Initial State Count   N                  File-Related Parms
                                                   Format Buffer


Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save                                    Menu  ↩
 ↩
```

4 Tab down to the **File-related Parameters** field on the Subscription Definition screen and enter an "S" for the field.

The List of Subscription SFILEs screen appears listing the SFILE definitions currently defined in the Adabas subscription definition.

```
18:32:30      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                        List of Subscription SFILEs                  M-RP1415
Current
                                                 User      BI  AI  KR
            Sel   DBID   File  Ins Upd Del       Exit      FB  FB  FB
            ---------------------------------------------------------
             _    1955   1      Y   Y   Y                   +   +
             _    1955   2      Y   Y   Y                   +   +
             _    1955   3      Y   Y   Y                   +   +
             _
             _
             _
             _
             _
             _
             _
             _
             _


Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit  Add               -     +                     Menu  ↩
 ↩
```

5 Press PF4 to start adding a new SFILE definition.

The File-Related Parameters screen appears.

```
18:39:04      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                         File-Related Parameters                     M-RP1420

Description ................. INITIAL STATE 040 155
Subscription Name ........... SI040155  Current
DBID ........................ _____
  File Number ............... ____
  Replicate for Insert ...... Y
  Replicate for Update ...... Y
  Include Identical Records .. Y
  Replicate for Delete ...... Y
  Replicate Security File .... N         Transaction Filter ..... * ________
  Subscription User Exit ..... ________  Filter Before Image FB .. * ________
  Default Code ............... ____      Filter After Image FB ... * ________


  Before Image FB ............ _   -or-  Before Image GFB Name . * ________
  After Image FB ............. _   -or-  After Image GFB Name .. * ________
  Key-Related Before Image FB  _   -or-  Key-Related GFB Name .. * ________

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save  Sel                                 Menu  ↩
 ↩
```

Step 2. Specify SFILE Definition Information

To specify the information in the SFILE definition of a subscription:

1 Update the fields on the File-Related Parameters screen as described in the following table.

Parameter Name	Specify	Default
Description	The description of the subscription to which this SFILE definition applies. You cannot change this field here.	----
Subscription Name	A unique name for the subscription. You cannot change this field here.	---
DBID	The database ID associated with the input file specified by this SFILE definition. The database ID is numeric and can range from one through 65535. This is the same as specifying the `SFDBID` parameter directly in the Event Replicator Server startup job.	---
File Number	The number of the input file to be processed by this SFILE definition. This is the equivalent of specifying the `SFILE` parameter directly in the Event Replicator Server startup job.	---

Parameter Name	Specify	Default
Replicate for Insert	Whether or not you want records from this Adabas file replicated when they are inserted. Specify "Y" if you want them replicated or "N" if you do not want them replicated. This is the same as specifying the `SFREPLICATEINSERT` parameter in the Event Replicator Server startup job.	Y
Replicate for Update	Whether or not you want records from this Adabas file replicated when they are updated. Specify "Y" if you want them replicated or "N" if you do not want them replicated. This is the same as specifying the `SFREPLICATEUPDATE` parameter in the Event Replicator Server startup job.	Y
Include Identical Records	Whether an update request should be replicated if the after image (AI) is the same as the before image (BI). Valid values are "Y" or "N". If "Y" is specified and the before and after images are the same, the record is replicated subject to other field filtering that may be specified for this file. If "N" is specified, and the before and after images are the same, no further processing occurs for the record. Note that the format buffers for the before and after images must be identical. This is the same as specifying the `SFREPLICATENOTCHANGED` parameter in the Event Replicator Server startup job.	Y
Replicate for Delete	Whether or not you want records from this Adabas file replicated when they are deleted. Valid values are "Y" (Yes), "N" (No), or "U" (Update). This is the same as specifying the `SFREPLICATEDELETE` parameter in the Event Replicator Server startup job. If this parameter is set to "N" and an input record for the DBID/file is for a delete, the input record will NOT be processed for this subscription. If this parameter is set to "Y" and an input record for the DBID/file is for a delete, the input record will be processed for this subscription. If this parameter is set to "U" and an input record for the DBID/file is for a delete, the before and after images of the input record are passed to your subscription user exit. Therefore, if this parameter is set to "U", a subscription user exit name must be specified in the `Subscription User Exit` parameter. In addition, the subscription before and after image format buffers must be identical and no primary key should be defined to the file (to ensure that the before image is a copy of data storage). The purpose of the "U" value of this parameter is to allow your subscription user exit to process replicated physical delete transactions on your target database as you choose. Your subscription user exit can decide if the physical delete transaction should be: physically deleted from your target database, converted to an update, or ignored and not sent at all. For more information about using the subscription user exit with the "U" setting of this parameter, read *Controlling Delete Transaction Processing*	Y

Parameter Name	Specify	Default
	(SFREPLICATEDELETE=UPDATE Processing), in the *Event Replicator for Adabas Administration and Operations Guide*.	
Replicate Security File	Whether or not you want security definitions in the Adabas security file replicated. Valid values are "Y" or "N", with a default of "N". The setting of this parameter allows you to define a subscription file (SFILE) definition specifically for security definitions in the Adabas security file on the database. **Note:** 1. This parameter cannot be set to "Y" unless you have Adabas 8.2 or later installed. 2. Adabas Security Facilities, including the Adabas security utility (ADASCR) can be obtained only by special request. If you are interested in Adabas Security Facilities, please contact your Software AG sales representative. When this parameter is set to "N", replication of the security definitions does not occur and replication processing proceeds normally. When this parameter is set to "Y", replication of the security definitions in the Adabas security file will occur. However, the following parameter settings are also required: ■ A format buffer may *not* be specified. This means that no values may be specified for the Before Image FB, Before Image GFB Name, After Image FB, After Image GFB Name, Key-Related Before Image FB, or Key-Related GFB Name parameters. ■ No value may be specified for the Default Code parameter. ■ A transaction filter definition may *not* be specified. This means that no value may be specified for the Transaction Filter parameter. ■ The default value of *Y* must be specified for the Replicate for Insert, Replicate for Update, Include Identical Records, and Replicate for Delete parameters. ■ A subscription exit may *not* be specified. This means that no value may be specified for the Subscription User Exit parameter. ■ When you set this parameter to "Y", you indicate that the file specified in the File Number parameter is the file number of the security file for the source database identified by the SFDBID parameter. Therefore, if this parameter is set to "Y" for other subscription SFILE definitions using the same source database (with the same SFILE DBID setting), the same value must be set for each of the File Number parameters in the SFILE definition. In other words, it is invalid for a source database (DBID setting) to have different file numbers specified for the security file in different subscriptions. For example, source database 10 cannot have the security	N

Parameter Name	Specify	Default
	file specified as both file 15 in one SFILE definition and 20 in another SFILE definition. ■ If this parameter is set to "Y" for a source database (DBID parameter), any Adabas destination definitions with same database specified as the destination input database (**Input DBID** parameter), must also specify identical file numbers for both the **Input File** and **Target File** parameters. This is the same as specifying the `SFSECURITYFILE` parameter in the Event Replicator Server startup job. For complete information about replicating security definitions, read *Replicating Security Definitions*, in the *Event Replicator for Adabas Administration and Operations Guide*.	
Subscription User Exit	The name of the subscription user exit you want used (if any) when records from the file specified by this SFILE definition are replicated. This is the same as specifying the `SFSEXIT` parameter in the Event Replicator Server startup job. For more information about subscription user exits, read *Using the Event Replicator Server Subscription User Exit*, in *Event Replicator for Adabas Administration and Operations Guide*.	---
Default Code	A default, at the subscription file level, for the file's alpha character encoding. The encoding must belong to the EBCDIC encoding family; that is, the space character must be X'40'.'. This can only be set when the Event Replicator Server has been started with Universal Encoding Support (UES) enabled. This parameter is meant to be used when *all* of the following conditions are met: ■ The Event Replicator Server is UES-enabled, ■ The source database is not UES-enabled ■ The subscription definition requests UES translation (i.e. parameter `SACODE` is specified) ■ The data in the source database file is stored in a code page other than the default alpha encoding. At the time a *before* or *after* image is decompressed and translated in the Event Replicator Server, the file encoding is taken as follows: 1. from the source FCB; 2. if the encoding is not set above, from the GCB of the source database; 3. if the encoding is not set above, from the value set for this parameter (SFDEFAULTACODE); 4. if the encoding is not set above, from the GCB of the Event Replicator Server.	none

Parameter Name	Specify	Default
	Note that if the source database is UES-enabled, the file encoding will be taken from either a or b above. This is the equivalent of specifying the SFDEFAULTACODE parameter directly in the Event Replicator Server startup job.	
Transaction Filter	The name of the transaction filter definition you want used (if any) when records from the file specified by this SFILE definition are replicated. This is the equivalent of specifying the SFFILTER parameter directly in the Event Replicator Server startup job. Press the PF6 key while the cursor is on this field to pick a definition from a list. **Note:** Any filter field specified in a transaction filter definition used by a subscription file (SFILE) definition must also be included in either the format buffers for the SFILE definition (SFBAI parameter , SGFORMATAI parameter , SFBBI parameter , and SGFORMATBI parameter parameters) or in filter format buffers for the SFILE definition (SFFILTERGFBAI parameter and SFFILTERGFBBI parameter parameters).	---
Filter Before Image FB	The name of a predefined **GFB definition** containing the format buffer to be used when filtering the data storage before image as described by the SFFILTER parameter. For more information about GFB definitions, read *Maintaining GFB Definitions*, elsewhere in this guide Ordinarily this parameter is optional. It is only required if *both* of the following conditions are met: 1. A transaction filter definition is specified in the SFFILTER parameter of the SFILE definition. 2. The normal format buffer or global format buffer specified for the data storage before image in the SFILE definition does not contain one or more of the fields defined in the filter. This is the equivalent of specifying the SFFILTERGFBBI parameter directly in the Event Replicator Server startup job. This parameter can be used to improve the performance of Event Replicator processing. For example, if most records for a large format buffer are being rejected due to a filter based on the contents of a small number of fields, it may help to specify the key fields in a filter format buffer so that, for most records, only the fields required to make acceptance/rejection decisions are decompressed instead of the entire buffer.	---
Filter After Image FB	The name of a predefined **GFB definition** containing the format buffer to be used when filtering the data storage after image as described by the SFFILTER parameter. For more information about GFB definitions, read *Maintaining GFB Definitions*, elsewhere in this guide. Ordinarily this parameter is optional. It is only required if *both* of the following conditions are met:	---

Parameter Name	Specify	Default
	1. A transaction filter definition is specified in the SFFILTER parameter of the SFILE definition. 2. The normal format buffer or global format buffer specified for the data storage after image in the SFILE definition does not contain one or more of the fields defined in the filter. This is the equivalent of specifying the `SFFILTERGFBAI` parameter directly in the Event Replicator Server startup job. This parameter can be used to improve the performance of Event Replicator processing. For example, if most records for a large format buffer are being rejected due to a filter based on the contents of a small number of fields, it may help to specify the key fields in a filter format buffer so that, for most records, only the fields required to make acceptance/rejection decisions are decompressed instead of the entire buffer.	
Before Image FB	An "S" to specify the format buffer to be used when decompressing the data storage before image. The Before Image FB screen appears on which you can specify the format buffer. This format buffer specification is saved in the SFILE definition. Format buffer specifications on the Before Image FB screen must conform to the format buffer rules (for read commands) described in the Adabas Command Reference. Be sure to press PF5 to save the format buffer specifications. Then press PF3 to return to the File-Related Parameters screen. PE and MU fields cannot use the range notation 1-N in format buffers for a subscription that is sent to a destination that has specified the replication intialization parameter DCLASS=SAGTARG. The SAGTARG application invoked requires that the range of occurrences specified are contained in the record buffer even if they are empty occurrences. 1-N results in a range of 1-191, but unless there are 191 occurrences containing data, space in the record buffer is not allocated for any empty occurrences, resulting in incorrect field positioning when processing the record. **Note:** This field and the **Before Image GFB Name** field are mutually exclusive. Only use one of them. Specifying anything in these two fields is optional. If nothing is specified, the format buffer identified in the **After Image FB** or **After Image GFB Name** field is used. This is the same as specifying the `SFBBI` parameter directly in the Event Replicator Server startup job.	The after-image format buffer or GFB is used if no before image format buffer or GFB are specified.
Before Image GFB Name	The name of a predefined **GFB definition** containing the format buffer to be used when decompressing the data storage before image. For more information about GFB definitions, read *Maintaining GFB Definitions*, elsewhere in this guide.	The after-image format

Parameter Name	Specify	Default
	Note: This field and the **Before Image FB** field are mutually exclusive. Only use one of them. Specifying anything in these two fields is optional. If nothing is specified, the format buffer identified in the **After Image FB** or **After Image GFB Name** field is used. This is the equivalent of specifying the `SGFORMATBI` parameter directly in the Event Replicator Server startup job. If you are using the Event Replicator Target Adapter (if **"SAGTARG"** is specified as the destination class of some destination used by this subscription), the value used for the `After Image GFB Name` parameter and the `Before Image GFB Name` parameter must be the same. In addition, the key image global format buffer specified in the `Key-Related GFB Name` parameter must have been built from a Predict user view with the same names as the user view used to build the before and after images, but with the suffix "-KEY" on the end. (Likewise, if you use the Data Mapping Tool, the `SGFORMATKEY` parameter GFB should be built from a DDM with the same name as the DDM used to build the before and after image GFBs.) Event Replicator processing strips off the "-KEY" to ensure that any delete is associated with the before and after image file name that was used to build the table(s) in the RDBMS.	buffer or GFB is used if no before image format buffer or GFB are specified.
After Image FB	An "S" to specify the format buffer to be used when decompressing the data storage after image. The After Image FB screen appears on which you can specify the format buffer. This format buffer specification is saved in the SFILE definition. Format buffer specifications on the After Image FB screen must conform to the format buffer rules (for read commands) described in the Adabas Command Reference. Be sure to press PF5 to save the format buffer specifications. Then press PF3 to return to the File-Related Parameters screen. PE and MU fields cannot use the range notation 1-N in format buffers for a subscription that is sent to a destination that has specified the replication intialization parameter DCLASS=SAGTARG. The SAGTARG application invoked requires that the range of occurrences specified are contained in the record buffer even if they are empty occurrences. 1-N results in a range of 1-191, but unless there are 191 occurrences containing data, space in the record buffer is not allocated for any empty occurrences, resulting in incorrect field positioning when processing the record. **Note:** This field and the **After Image GFB Name** field are mutually exclusive. Only use one of them. **Caution:** This "C." option may be used *only* if the destination target file has been defined with the same fields in the same order as the fields in the source file; if there are differences in the definitions of the files, the replication of the data is likely to incur errors. The only exception to this	No default. A value for this field or the After Image GFB Name field is required.

Parameter Name	Specify	Default
	rule is that the definitions of the descriptors and superdescriptors in the target and source files may be different. This is the same as specifying the SFBAI subparameter in the Event Replicator Server startup job.	
After Image GFB Name	The name of a predefined **GFB definition** containing the format buffer to be used when decompressing the data storage after image. For more information about GFB definitions, read *Maintaining GFB Definitions*, elsewhere in this guide. **Note:** This field and the **After Image FB** field are mutually exclusive. Only use one of them. **Caution:** This "C." option may be used *only* if the destination target file has been defined with the same fields in the same order as the fields in the source file; if there are differences in the definitions of the files, the replication of the data is likely to incur errors. The only exception to this rule is that the definitions of the descriptors and superdescriptors in the target and source files may be different. This is the same as specifying the SGFORMATAI parameter in the Event Replicator Server startup job. If you are using the Event Replicator Target Adapter (if **"SAGTARG"** is specified as the destination class of some destination used by this subscription), the value used for the After Image GFB Name parameter and the Before Image GFB Name parameter must be the same. In addition, the key image global format buffer specified in the Key-Related GFB Name parameter must have been built from a Predict user view with the same names as the user view used to build the before and after images, but with the suffix "-KEY" on the end. (Likewise, if you use the Data Mapping Tool, the SGFORMATKEY parameter GFB should be built from a DDM with the same name as the DDM used to build the before and after image GFBs.) Event Replicator processing strips off the "-KEY" to ensure that any delete is associated with the before and after image file name that was used to build the table(s) in the RDBMS.	No default. A value for this field or the After Image FB field is required.
Key-Related Before Image FB	An "S" to specify the format buffer to be used when decompressing the key-related before image. The Key-Related FB screen appears on which you can specify the format buffer. This format buffer specification is saved in the SFILE definition. Format buffer specifications on the Key-Related FB screen must conform to the format buffer rules (for read commands) described in the Adabas Command Reference. Be sure to press PF5 to save the format buffer specifications. Then press PF3 to return to the File-Related Parameters screen.	The field name followed by a period will be used as a default for this format buffer.

Parameter Name	Specify	Default
	Note: This field and the **Key-Related GFB Name** field are mutually exclusive. Only use one of them. Specifying anything in these two fields is optional. If nothing is specified, the field name followed by a period will be used for the format buffer. This is the same as specifying the `SFBKEY` parameter directly in the Event Replicator Server startup job.	
Key-Related GFB Name	The name of a predefined GFB definition containing the format buffer to be used when decompressing the key-related before image. For more information about GFB definitions, read ***Maintaining GFB Definitions***, elsewhere in this guide. **Note:** This field and the **Key-Related Before Image FB** field are mutually exclusive. Only use one of them. Specifying anything in these two fields is optional. If nothing is specified, the field name followed by a period will be used for the format buffer. This is the equivalent of specifying the `SGFORMATKEY` parameter directly in the Event Replicator Server startup job. If you are using the Event Replicator Target Adapter (if **"SAGTARG"** is specified as the destination class of some destination used by this subscription), the value used for the `After Image GFB Name` parameter and the `Before Image GFB Name` parameter must be the same. In addition, the key image global format buffer specified in the `Key-Related GFB Name` parameter must have been built from a Predict user view with the same names as the user view used to build the before and after images, but with the suffix "-KEY" on the end. (Likewise, if you use the Data Mapping Tool, the `SGFORMATKEY` parameter GFB should be built from a DDM with the same name as the DDM used to build the before and after image GFBs.) Event Replicator processing strips off the "-KEY" to ensure that any delete is associated with the before and after image file name that was used to build the table(s) in the RDBMS.	The field name followed by a period will be used as a default for this format buffer.

2 Press PF5 to save the SFILE definition.

Step 3. Save the SFILE Definition

To save the SFILE definition:

- Press PF5 to save the SFILE definition in the subscription definition in the Replicator system file.

Modifying SFILE Definitions

To use the Adabas Event Replicator Subsystem to modify an SFILE definition in a subscription:

1 List the subscription definitions in the Adabas Event Replicator Subsystem, as described in *Listing SFILE Definitions*, elsewhere in this guide.

The SFILE definitions are listed on the List of Subscription SFILEs screen.

2 Locate the definition you want to modify on the screen and enter an **M** in the **Sel** column for that definition.

The File-Related Parameters screen appears listing file-related parameters for the Adabas subscription. For information on modifying this screen, read the description of *Adding SFILE Definitions*, elsewhere in this section.

3 When all modifications have been made, press PF5 to save the changes.

The SFILE modification is complete.

Deleting SFILE Definitions

To use the Adabas Event Replicator Subsystem to delete an SFILE definition in a subscription:

1 List the SFILE definitions in the Adabas Event Replicator Subsystem, as described in *Listing SFILE Definitions*, elsewhere in this guide.

The SFILE definitions are listed on the List of Subscription SFILEs screen.

2 Locate the definition you want to delete on the screen and enter a **D** in the **Sel** column for that definition.

The SFILE definition is deleted.

10 Maintaining GFB Definitions

A GFB definition defines a global format buffer (GFB) stored separately for reference in SFILE definitions. GFB definitions can be used to decompress replicated data from a specific database file for one or more subscriptions.

While a format buffer specification is required in a subscription's SFILE definition, a stored GFB definition does not need to be used. The SFILE definition could simply include the format buffer specifications it needs.

Notes:

1. If you will be using the Event Replicator Target Adapter to replicate data to an RDBMS, you must generate your GFB definitions, as described in *Generating a GFB*, elsewhere in this chapter.
2. If you want to use UTF-8 character encoding, you must verify that your GFB field lengths are increased as required to accommodate the character set referenced by the code page you select and the data requested in the GFB. You can increase these field lengths manually by editing the GFB itself or by editing the Predict file or data definition module (DDM) used when the GFB is generated.

Using the Adabas Event Replicator Subsystem you can manually add, modify, and delete GFB definitions. If you have an appropriate version of Predict installed (see *Predict Requirements*), you can also generate a GFB. If the correct version of Predict is not installed, you will not be able to use this feature.

Listing GFB Definitions

To use the Adabas Event Replicator Subsystem to list the general format buffer (GFB) definitions stored in the Replicator system file:

- Select option **G** from the Adabas Event Replicator Subsystem Main Menu.

  The List of Global Format Buffers screen appears showing all of the GFB definitions in the Adabas Event Replicator Subsystem.

```
18:40:09      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                        List of Global Format Buffers                M-RP1130

 Sel   Name         Sel   Name         Sel   Name         Sel   Name
 ------------       ------------       ------------       ------------
  _  FILE200         _                  _                  _
  _  F040155         _                  _                  _
  _  F042060         _                  _                  _
  _  F046088         _                  _                  _
  _  F062026         _                  _                  _
  _  F062029         _                  _                  _
  _  F062035         _                  _                  _
  _  F062055         _                  _                  _
  _  F062079         _                  _                  _
  _  F062106         _                  _                  _
  _  F062143         _                  _                  _
  _  F064121         _                  _                  _
  _  F120248         _                  _                  _
  _  F215168         _                  _                  _

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Gen   Exit  Add   Repos        -     +                     Menu  ↩
 ↩
```

The function keys on this screen perform the following functions:

Function Key	Description
PF1/F1 (Help)	Provides you with help for this screen.
PF2/F2 (Gen)	Allows you to generate GFB if you have Predict 4.4.1 with Service Pack 3 (or higher) installed. A new screen appears.
PF3/F3 (Exit)	Returns you to the previous screen.
PF4/F4 (Add)	Allows you to add a new definition. A new screen appears.
PF5/F5 (Repos)	Provides you with a pop-up panel that allows you to specify the name of the definition you want to locate in the list. Once you have specified a name on the pop-up panel and pressed Enter, the list is repositioned so the name you selected appears first. You can use an asterisk as a wild card character at the end of the definition name or partial definition name you specify on the pop-up panel. Or, you can simply enter the first few characters of the name to reposition the list to the first occurrence in the list of a name starting with those characters.
PF7/F7 (-)	Allows you to scroll backwards through the list of definitions.
PF8/F8 (+)	Allows you to scroll forwards through the list of definitions.
PF12/F12 (Menu)	Returns you to the main menu.

Adding GFB Definitions

You can add global format definitions by typing in the format buffer definition manually or using Predict to generate one for you. This section describes how to add one manually.

To use the Adabas Event Replicator Subsystem to add a global format definition manually:

1 Select option **G** from the Adabas Event Replicator Subsystem Main Menu.

The List of Global Format Buffers screen appears.

2 Press the PF4 function key.

The Global Format Buffer screen appears.

```
18:40:31      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                           Global Format Buffer                     M-RP1120

GFB Name .. ________  Predict Parameters .. _
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________________________________________



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit  Mode  Save        -     +                      Menu  ↩
 ↩
```

3 Tab to the **GFB Name** field and specify a unique name for the global format buffer definition. The name must be between one and seven alphanumeric characters long.

Note: If, at this point, you want to generate the GFB and field table (GFFT), rather than enter it manually, enter an "S" in the **Predict Parameters** field. The Predict Parameters screen appears. For more information about generating a GFB, skip the rest of the documentation in this section and read *Generating a GFB* elsewhere in this section.

Caution: This "C." option may be used *only* if the destination target file has been defined with the same fields in the same order as the fields in the source file; if there are differences in the definitions of the files, the replication of the data is likely to incur errors. The only exception to this rule is that the definitions of the descriptors and superdescriptors in the target and source files may be different.

4 Use the blank lines below the **GFB Name** field to manually specify the format buffer for this definition. The format buffer must conform to the format buffer requirements (for read commands) documented in the Adabas command reference documentation (read *Adabas Command Reference*).

Use the PF7 and PF8 keys to scroll backwards and forwards through the format buffer definition.

PE and MU fields cannot use the range notation 1-N in format buffers for a subscription that is sent to a destination that has specified the replication intialization parameter DCLASS=SAGTARG. The SAGTARG application invoked requires that the range of occurrences specified are contained in the record buffer even if they are empty occurrences. 1-N results in a range of 1-191, but unless there are 191 occurrences containing data, space in the record buffer is not allocated for any empty occurrences, resulting in incorrect field positioning when processing the record.

5 Press PF5 to save the GFB definition.

Generating a GFB

If you have Predict installed, you can generate format buffers and corresponding field tables (GFFTs) from Predict file definitions using the Adabas Event Replicator Subsystem.

Notes:

1. If you will be using the Event Replicator Target Adapter to replicate data to an RDBMS, you must generate your GFB definitions, as described in this section.
2. MU and PE fields are supported in the generated format buffers and their corresponding field tables (GFFTs), although MU fields within PE fields are not. This support requires that counters for MU and PE fields be available in the Predict data dictionary definition. Counters for MU and PE fields are available if the Predict data dictionary definition either specifies fields with field types of MC (MU fields with an automatic counter) and PC (PE fields with an automatic counter) *OR* includes explicit CM field (MU counter field) or CP field (PE counter field) definitions. For more information about CM, CP, MC, MU, PC, and PE field types, refer to your Predict documentation.
3. PE and MU fields cannot use the range notation 1-N in format buffers for a subscription that is sent to a destination that has specified the replication intialization parameter

DCLASS=SAGTARG. The SAGTARG application invoked requires that the range of occurrences specified are contained in the record buffer even if they are empty occurrences. 1-N results in a range of 1-191, but unless there are 191 occurrences containing data, space in the record buffer is not allocated for any empty occurrences, resulting in incorrect field positioning when processing the record.

4. Exception: This release of the Event Replicator Target Adapter Data Mapping Tool introduces a new option that allows the user to optimize a buffer containing MU's, PE's, and MU's within PE's.
5. LA and LB fields are not supported in generated format buffers.

Optimized Buffer Option Restrictions

Optimized buffer periodic group fields are generated using the 1-N notation. This indicates that all occurrence values for this field be returned. In order to process the number of occurrences returned in the record buffer correctly, the periodic group field count is requested first, followed by all occurrences of each field defined as a member of the periodic group within the global format buffer (GFB) and global format field table (GFFT).

The count provides the number of occurrences that were returned for each field requested within the periodic group and is used when processing the record buffer.

Some customers may wish to combine 2 or more logically related periodic group fields under one periodic group name so that they are inserted into one RDBMS table by the Event Replicator Target Adapter.

Combining related fields from 2 or more periodic groups under one periodic group requires that all fields have the same number of occurrences. If they do not, then the record buffer will be processed incorrectly starting with the first field that has more or less occurrences than what the count field reported. This may cause the Event Replicator Target Adapter to fail, or bad data to be loaded into the RDBMS.

Optimized Buffer Option Restrictions - Example

```
1 | AP | | | PE | |
2 | AQ | 4 | A | NU | |
2 | AR | 1 | A | NU | |
1 | AS | | | PE | |
2 | AT | 4 | A | NU | |
2 | AU | 8 | U | NU | |
```

For this example PE group AS is logically related to PE group AP. PE Group AS fields AT and AU are to be included in the group AP within the DDM used by the Event Replicator Target Adapter Data Mapping Tool to generate the optimized GFB and GFFT. The portion of the GFB for the AP PE group would look like this:

```
'AP,2,B,AQ1-N,4,A,AR1-N,1,A,AT1-N,4,A,AU1-N,8,U.'
```

Notice the count for the AP group is first, followed by the 4 PE group fields that are to be defined as members of the periodic AP group.

When the GFB was generated, an internal global format field table (GFFT) was also generated. This table defines the 4 fields AQ, AR, AT, and AU as belonging to the periodic group AP. In order for this to work correctly, the fields AT and AU from the AS PE group must have the same occurrence count as the fields in the PE group AP.

If the count field returned for the periodic group AP is 5, but the periodic group AS fields AT and AU only have 3 occurrences each, then they should not be included as being members of the AP periodic group in the DDM or Predict user view used to generated the GFB and GFFT definitions.

To use the Adabas Event Replicator Subsystem to generate global format buffer definition using Predict:

1 Select option **G** from the Adabas Event Replicator Subsystem Main Menu.

The List of Global Format Buffers screen appears.

2 Press the PF2 function key.

The Predict Parameters screen appears.

```
18:40:54      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
FDIC=(1955,13)               Predict Parameters                     M-RP1121

GFB Name .......... _______

File ID ......... * ________________________________
Target file ID .. * ________________________________

----------  Generation Information ----------

User ..............
Date ..............
Time ..............
FDIC .............. DBID ..       FNR ..

Adabas version ....
Occurrences used ..
Full format .......


Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Exec  Sel   Unlnk                        Menu  ↩
 ↩
```

3 Update the following fields on this screen as described in this table.

Parameter Name	Specify	Default
File ID	The name of a Predict file with a file type or Adabas (A) or Adabas user view (U). Place your cursor on this field and press PF6 to select a file from a List of Predict Files screen. To select from the list screen, type an "S" next to the file you want to use and press PF5.	---
GFB Name	A unique name for the global format buffer definition. The name must be between one and seven characters long.	---
Target file ID	The name of a Predict file with a file type of sequential (S). This file may be used to insert space notation (nX) into the format buffer. For fields in the target file with matching definitions in the File ID file (in other words, if a field exists with the same field long name), a short name clause is generated. For fields that do not have a matching definition in the File ID file, an appropriate space notation (nX clause) is generated. The spaces defined by nX clauses can be filled using a user exit. Place your cursor on this field and press PF6 to select a file from a List of Predict Files screen. To select from the list screen, type an "S" next to the file you want to use and press PF5.	---

When you first create a definition, the remaining fields on this screen are blank. However, when you modify the definition later, they are filled in, although you cannot modify them. These display-only fields are described in the following table:

Parameter Name	Displays
Adabas version	The version of Adabas for which the global format buffer was generated.
Date	The date the global format buffer was generated.
FDIC (top of the screen)	The current database and file number of the Predict file.
FDIC...DBID...FNR	The database and file number of the Predict file.
Full format	Whether the full format buffer was generated. The full format buffer includes the length and format of Adabas fields. A value of "Y" indicates that the full format buffer was generated; a value of "N" indicates it was not.
Occurrences used	How multiple occurrences of PE and MU fields are generated in the GFB and resulting field table (GFFT). A value of "M" indicates that the maximum number of occurrences should be generated (191); a value of "N" indicates that no occurrences will be generated; a value of "Y" indicates that the number of occurrences defined by the Predict `Occ` attribute should be generated.
Time	The time of day the global format buffer was generated.
User	The user ID of the user who generated the global format buffer.

4 When you have supplied values for the `File ID`, `GFB Name`, and `Target file ID` fields, press PF5 to start generating the global format buffer.

A small window appears requesting more information.

```
+----------------------------+
! Adabas Version ...* I7      !
! Occurrences used..* Y       !
! Full format ..... Y (Y/N) !
+----------------------------+
```

5 Update the fields on this screen as described in the following table:

Parameter Name	Specify	Default
Adabas Version	The version of Adabas for which the global format buffer will be generated. The version should be expressed as "I7" or "R7". If you want special fields and descriptors included in the generated GFB and corresponding field tables (GFFTs), specify "R7".	I7
Full format	You cannot edit this parameter. It indicates that the full format buffer should be generated.	Y
Occurrences used	How multiple occurrences of PE and MU fields are generated in the GFB. A value of "M" indicates that the maximum number of occurrences should be generated (191); a value of "N" indicates that no occurrences will be generated; a value of "Y" indicates that the number of occurrences defined by the Predict `Occ` attribute should be generated.	Y

6 When these fields are set to your liking, press ENTER.

The global format buffer definition and field table (GFFT) are generated and the **Global Format Buffer** screen appears.

```
18:40:54      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                            Global Format Buffer                     M-RP1124

GFB Name .. GBF037_  Predict Parameters .. _ +                    19 of 78

Ty L Field ID                          R  K KNum Flatten  Format Buffer
      F  Cs  Length (2nd line)
   1 MARCA-DEFINITIVA                  _  _  __   _       AY,1,U
      N           1.0
   1 FECHA-INFORMATICA                 _  _  __   _       AZ,4,P
      D           4.0
   1 FX-FSE-SEG-ACC                    _  _  __   _       BD,4,U
      N           4.0
MU 1 ID-RESUL-UN(1-191)                _  _  __   _       CU1-191,1,U
      N           1.0
MU 1 ID-RESUL-FC(1-191)                _  _  __   _       CF1-191,1,U
      N           1.0
CP 1 C_PRACTICAS-NOLAB                 _                  BGC,2,B
      B           2.0

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit  Mode  Save        -     +                      Menu  ↩
 ↩
```

Note: *SQL Significance Indicator* fields are marked with an "_S" in the `Ty` field (field type column).

7 Optionally, modify the **Cs**, **R**, **K**, **Knum**, or **Flatten** settings for the fields, as described in the following table:

Setting	Valid Values	Description
Cs	"C" or blank	This setting indicates whether or not the field content should be converted to character format in the destination user exit. The Cs setting is only modifiable for non-counter fields in binary format with lengths of 1-8 characters. Valid values are C (convert to the field content to character) or blank (do not convert the field content). **Note:** This field cannot be modified for counter fields (`Ty` field set to "CM" or "CP") and SQL Significance Indicator fields (`Ty` field set to "_S").
R	"Y" or blank	This setting indicates whether or not the field is read-only. Read-only fields are not replicated. A value of "Y" indicates that the field is read-only (not replicated); a blank value indicates that the field is not read-only and is replicated. **Note:** When you initially generate or regenerate the GFB and field table (GFFT), the value of the **R** setting for all fields is blank.

Setting	Valid Values	Description
K	"D", "P", "U", or blank	This setting indicates the kind of key field that the field represents. Possible values are: ■ "D" -- identifies a descriptor key field ■ "P" -- identifies a primary key field. Only one field in the table can be marked with a "P" unless you also supply a **KNum** value, indicating that the field is part of a composite key. **Note:** At this time, multiple-value (MU) and periodic group (PE) fields cannot be specified as primary keys or in composite keys if Event Replicator Target Adapter processing will be invoked by the subscription (if the DCLASS parameter of the destination used by the subscription is set to "SAGTARG"). ■ "U" -- identifies a UQ descriptor key field ■ blank -- identifies a non-key field **Note:** 1. When you initially generate or regenerate the GFB and field table (GFFT), the key fields are set to "D", "U", or blank, depending on the Predict definitions for those fields. 2. This field cannot be modified for counter fields (`Ty` field set to "CM" or "CP") and SQL Significance Indicator fields (`Ty` field set to "_S").
KNum	1 - 63 or blank	KNum can be specified when the K setting is "P" and when a composite key is needed. Specify a value from 1-63 to identify the order of the fields in the composite key. If no composite key is needed, leave the KNum setting blank.
Flatten	"F" or blank	Use the "F" setting to indicate that an MU or PE field should be flattened when it is replicated to any relational database tables using the Event Replicator Target Adapter. Otherwise, leave this field blank. By default, when MU and PE fields and subfields and superfields are included in replicated data in RDBMS targets (via the Event Replicator Target Adapter), additional tables are created. However, you can request that individual MU and PE fields, subfields, and superfields be flattened in the replicated data instead. This process of flattening fields will replicate them as columns in the main RDBMS table, rather than as separate subtables. If you want to flatten a field in the resulting RDBMS table, you must identify it in the GFB and field table (GFFT) for the field. If you do not explicitly trigger a field to be flattened in the GFB and GFFT, it will not be flattened in the resulting RDBMS tables.

8 Once all changes are made, press PF5 to save the generated global format buffer definition.

Modifying GFB Definitions

To use the Adabas Event Replicator Subsystem to modify a global format buffer (GFB) definition in the Replicator system file:

1 List the GFB definitions in the Adabas Event Replicator Subsystem, as described in *Listing GFB Definitions*, elsewhere in this guide.

The GFB definitions are listed on the List of Global Format Buffers screen.

2 Locate the definition you want to modify on the screen and enter an **M** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF5 (F5) to specify the name of the definition to which the list should be repositioned.

The Global Format Buffer screen for the GFB you selected appears .

3 Manually modify the GFB definition on this screen.

For information on manually modifying this screen, read the description of *Adding GFB Definitions*, elsewhere in this section.

Or:

If the **Predict Parameters** field has a plus (+) symbol next to it, the GFB was originally generated by the Adabas Event Replicator Subsystem. Enter an "S" in the `Predict Parameters` field to regenerate the GFB on the Predict Parameters screen. For information about generating a GFB, read *Generating a GFB* elsewhere in this section

4 When all modifications have been made, press PF5 to save the changes.

Unlinking a Generated GFB from its Predict Generation Information

If a GFB definition has been generated from Predict and it has been saved, you can unlink the Predict generation information from the generated GFB definition.

To use the Adabas Event Replicator Subsystem to unlink the Predict generation information from the GFB:

1 Access the GFB definition, as described in *Modifying GFB Definitions*earlier in this section. Be sure to enter an "S" in the `Predict Parameters` field of the Global Format Buffer screen. This will allow you to access the Predict Parameters screen for that GFB.

2 Once the Predict Parameters screen associated with the GFB appears, click PF7 to unlink the Predict generation information from the GFB.

The Global Format Buffer screen appears.

3 Press PF5 to save the GFB definition.

Copying GFB Definitions

To use the Adabas Event Replicator Subsystem to copy a global format buffer (GFB) definition in the Replicator system file:

1 List the GFB definitions in the Adabas Event Replicator Subsystem, as described in *Listing GFB Definitions*, elsewhere in this guide.

The GFB definitions are listed on the List of Global Format Buffers screen.

2 Locate the definition you want to copy on the screen and enter a **C** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF5 (F5) to specify the name of the definition to which the list should be repositioned.

A dialog appears requesting a name for the copy of the GFB definition.

```
Enter new name: _________
or press ENTER to cancel

                                ↩
```

3 Specify a new, unique name for the copy of the GFB definition and press Enter.

The GFB definition is copied and the copy appears on the List of Global Format Buffers screen.

Deleting GFB Definitions

To use the Adabas Event Replicator Subsystem to delete a global format buffer (GFB) definition in the Replicator system file:

1 List the GFB definitions in the Adabas Event Replicator Subsystem, as described in *Listing GFB Definitions*, elsewhere in this guide.

The GFB definitions are listed on the List of Global Format Buffers screen.

2 Locate the definition you want to delete on the screen and enter a **D** in the **Sel** column for that definition.

You can locate the definition you want in the list by pressing the PF7 (F7) or PF8 (F8) keys to scroll through the list. You can also press PF5 (F5) to specify the name of the definition to which the list should be repositioned.

The GFB definition is deleted.

11 Initiating a Replay Request Using the Adabas Event Replicator Subsystem

This chapter describes how you can initiate synchronized and replay-only replay processing. This method involves a combination of the Adabas Event Replicator Subsystem and a batch ADARPL utility job or automated replay. You first use the Adabas Event Replicator Subsystem to generate a replay request. The replay request is assigned a token that you then use in the batch ADARPL utility job. For complete information about the ADARPL utility, read *ADARPL Utility: PLOG Replication Replay* in *Event Replicator for Adabas Reference Guide*.

To generate a synchronized or replay-only replay request using the Adabas Event Replicator Subsystem and the ADARPL utility, complete the following steps:

1 Select option **A** from the Adabas Event Replicator Subsystem Main Menu.

The Administration menu appears.

```
14:53:47      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                              Administration                         M-RP1100


                        Code    Function
                        ----    --------------------------------
                          D     Database ID
                          I     Perform Initial-State
                          P     PLOG Information
                          R     Initiate Replay
                          S     System Functions
                          T     Target Adapter
                          V     Global Values
                          ?     Help
                          .     Exit
                        ----    --------------------------------

                 Code ...  _


Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu  ↩
 ↩
```

2 Select option **R** on the Administration menu.

The Initiate Replication Replay menu appears.

```
18:44:01      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                        Initiate Replication Replay                  M-RP2005
                                                                                ↩
                                                                                ↩
                                                                                ↩
                      Mode    Definition                                        ↩
                      ----    --------------------------------                  ↩
                        S     Synchronized                                      ↩
                        R     Replay Only                                       ↩
                        ?     Help                                              ↩
                        .     Exit                                              ↩
                      ----    --------------------------------                  ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
                 Mode ...  _                                                    ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
                                                                                ↩
Command ==>                                                                     ↩

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu ↩
 ↩
```

3 Select a mode on the Initiate Replication Replay menu. For more information about replay modes, read *Understanding Replay Modes*, in *Event Replicator for Adabas Administration and Operations Guide*.

An Initiate Replication Replay screen appears for the replay mode you selected. For example, if you selected S (synchronized) mode, the following screen appears:

```
18:44:25      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                        Initiate Replication Replay                  M-RP2010

                                  Synchronized

          DBID ............................ _____
          Automated ....................... N (Y or N)
          Timeout ......................... 900_______

          From Date/Time .................. ___________  _________
          To Date/Time .................... ___________  _________
          Start Date/Time ................. ___________  _________

          Destination Name List ..... _   * _________  _________  _________
          Subscription Name List .... _   * _________  _________  _________




Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Sub   Sel                                Menu  ↩
 ↩
```

4 Update the following fields on this screen as described in this table.

Parameter Name	Specify	Default
Automated	Indicate whether or not you want the replay automated or not. Valid values are "Y" (perform an automated replay) or "N" (do not perform an automated replay). An automated replay will automatically perform steps **6 through 8** of this procedure. A non-automated replay will not perform these steps automatically, and you will need to perform them manually. For complete information about automating replay processing, read *Automating Replay Processing*, in the *Event Replicator for Adabas Administration and Operations Guide*. **Note:** If the RECORDPLOGINFO parameter has been set to NO, you cannot run an automated replay.	N
DBID	The database ID of the Adabas database from which you want replicated transactions replayed.	---
Destination Name List	A list of destinations for which the replay request should be initiated. When the replay request is initiated, transactions will be replayed that were originally destined for the destinations on this list. Tab to the larger spaces for the **Destination Name List** field and type in the names of up to three destinations for replay processing. If you would prefer to	---

Parameter Name	Specify	Default
	select the names from a list, place the cursor on one of the three larger spaces and press **PF6**. If you want to select all destinations for replay processing, tab to the first large space for the **Destination Name List** field and enter an asterisk (*). Note that once you have entered an asterisk in the first field, you can no longer select any specific destinations (errors will occur if you try). Alternatively, if you need to enter more than three destinations or if you want to review the complete list of destinations selected for replay processing, type an "X" in the single-character space for the **Destination Name List** field and press Enter. A screen appears on which you can maintain the complete list of destinations. Using this screen: ■ Type in the destination definition names in the spaces provided. If you would prefer to select the names from a list, place the cursor on one of the spaces and press **PF6**. A list appears from which you can select destinations. ■ When you are satisfied with the list of destinations on the Destination List screen, press **PF5** to accept them and return to the Initiate Replication Replay screen.	
From Date/Time	The date and time from which replicated transactions should be replayed. Dates should be specified in YYYY/MM/DD format; times should be specified in HH:MM:SS format. Replay processing will start with transactions in the PLOG that ended at or after this date and time. From dates and times must be earlier than the current date and time and earlier than the specified end date and time.	---
Start Date/Time	The date and time of the PLOG entries that should be used as a starting point for the replay processing. This date and time are used to identify the PLOG with which to start replay processing. Dates should be specified in YYYY/MM/DD format; times should be specified in HH:MM:SS format. Replay processing will search the PLOG with this start date and time first for records that match the other replay processing criteria listed on this screen. A start date and time must be specified if an automated replay is requested.	---
Subscription Name List	A list of subscriptions for which the replay request should be initiated. When the replay request is initiated, transactions will be replayed that were originally initiated by the subscriptions on this list. Tab to the larger spaces for the **Subscription Name List** field and type in the names of up to three subscriptions for replay processing. If you would prefer to select the names from a list, place the cursor on one of the three larger spaces and press **PF6**. If you want to select all subscriptions for replay processing, tab to the first large space for the **Subscription Name List** field and enter an asterisk (*). Note that	---

Parameter Name	Specify	Default
	once you have entered an asterisk in the first field, you can no longer select any specific subscriptions (errors will occur if you try). Alternatively, if you need to enter more than three subscriptions or if you want to review the complete list of subscriptions selected for replay processing, type an "X" in the single-character space for the **Subscription Name List** field and press Enter. A screen appears on which you can maintain the complete list of subscriptions. Using this screen: ■ Type in the subscription definition names in the spaces provided. If you would prefer to select the names from a list, place the cursor on one of the spaces and press **PF6**. A list appears from which you can select subscriptions. ■ When you are satisfied with the list of subscriptions on the Subscription List screen, press **PF5** to accept them and return to the Initiate Replication Replay screen.	
Timeout	Optionally, specify the length of time, in seconds, at which the replay request should time out.	900 seconds
To Date/Time	The date and time to which replicated transactions should be replayed. Dates should be specified in YYYY/MM/DD format; times should be specified in HH:MM:SS format. Replay processing will stop with transactions in the PLOG that ended before this date and time. End dates and times must be later than the specified start date and time. If no end date and time are specified, the end time is the current time (the time the replay request is issued).	The current time

5 When all fields on the Initiate Replication Replay screen have been filled in to your satisfaction, press **PF5** to initiate the replay request.

The replay request is generated and a replay token is assigned to it. This replay token is displayed in an Adabas Event Replicator Subsystem message and in the Event Replicator Server job log.

Make a note of this token number as it is used in **step 8** if you are initiating replication replay using a batch ADARPL job.

If you have automated replication replay processing, this token number is picked up automatically by the generated replay jobstream and you can skip the remaining steps in this procedure. For complete information about automating replay processing, read *Automating Replay Processing*, in the *Event Replicator for Adabas Administration and Operations Guide*.

6 This step should not be performed if an automated replay is requested (Automated = Y on the screen).

If necessary, issue a force-end-of-PLOG request to the Adabas database and wait until the resulting PLCOPY job has copied or merged the latest PLOG data set. This is necessary only when the PLOG for the selected replay end date and time has not yet been copied or merged, for example, if no end date and time were specified in the replay request. This is also only necessary if an automated replay was not selected (Automated = N on the screen).

7 This step should not be performed if an automated replay is requested (Automated = Y on the screen).

Identify the sequential PLOG data sets that contain the protection data for the replicated records you need replayed. The PLOG data sets must build a complete sequence from the PLOG that includes the replay processing start time to the latest PLOG you copied or merged in the previous step.

8 This step should not be performed if an automated replay is requested (Automated = Y on the screen).

Run an ADARPL utility job, using the syntax described in *Syntax for Initiating ADARPL With A Token* in *Event Replicator for Adabas Reference Guide*. Be sure to specify:

- A concatenated list of the PLOG data sets you identified in the previous step.
- The replay request token assigned in **step 5**. This token should be specified in the ADARPL TOKEN parameter.
- The Event Replicator Server ID of the Event Replicator Server to which the replayed transactions should be sent. This token should be specified in the ADARPL RPLTARGETID parameter.

For more information about using the ADARPL Utility, in general, read *ADARPL Utility: PLOG Replication Replay* in *Event Replicator for Adabas Reference Guide*.

The replay process is initiated using the replay request generated in the Adabas Event Replicator Subsystem.

12 Reviewing and Managing the PLOG Data Set List

Adabas Event Replicator Subsystem replay processing provides a list of PLOG data sets for which replay processing can be initiated. This list also displays the date and time of the earliest transaction in each PLOG data set. You can use this list to review information about each PLOG and determine which PLOGs you need to replay. You can also delete PLOG data set entries from this list. This might be useful if the PLOG data set list becomes too long. For example, very old PLOG data sets appearing in the list may no longer be of use to you. In this case, you can use this screen to remove their PLOG entries from the list.

Note: The list of PLOG data sets may not appear or may contain inaccurate information, if the `RECORDPLOGINFO` initialization parameter is not set to "YES". You can control this parameter setting from the Adabas Event Replicator Subsystem using the **Record PLOG information** field on the Global Values screen. For more information about the `RECORDPLOGINFO` initialization parameter, read *RECORDPLOGINFO Parameter* in *Event Replicator for Adabas Reference Guide*. For more information about the **Record PLOG information** field, read ***Setting Global Values***, elsewhere in this guide.

Listing the PLOG Data Sets

To review the list of PLOG data sets for which replay processing can be initiated:

1 Select option **A** from the Adabas Event Replicator Subsystem Main Menu.

The Administration menu appears.

```
14:53:47      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                                 Administration                      M-RP1100


                          Code    Function
                          ----    --------------------------------
                            D     Database ID
                            I     Perform Initial-State
                            P     PLOG Information
                            R     Initiate Replay
                            S     System Functions
                            T     Target Adapter
                            V     Global Values
                            ?     Help
                            .     Exit
                          ----    --------------------------------

                       Code ...  _


Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu  ↩
 ↩
```

2 Select option **P** on the Administration menu.

The Replication Replay -- PLOG List screen appears.

```
14:53:47      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                      Replication Replay - PLOG List                M-RP2055
                                                                              ↩
                                                                              ↩
 Replay    Date        Time        PLOG Dataset Name                          ↩
 ------  ----------------------   ------------------------------------------- ↩
   _      2007-06-01  17:50:17   RD.SAMPC.SEQ.PLOG1                          ↩
   _      2007-06-01  17:50:29   RD.SAMPC.SEQ.PLOG1                          ↩
   _      2007-06-01  17:50:37   RD.SAMPC.SEQ.PLOG1                          ↩
   _      2007-06-01  17:50:47   RD.SAMPC.SEQ.PLOG1                          ↩
   _      2007-06-01  17:50:54   RD.SAMPC.SEQ.PLOG1                          ↩
   _      2007-06-01  17:51:01   RD.SAMPC.SEQ.PLOG1                          ↩
   _      2007-06-01  17:51:07   RD.SAMPC.SEQ.PLOG1                          ↩
   _      2007-06-01  17:51:15   RD.SAMPC.SEQ.PLOG1                          ↩
   _      2007-06-01  17:51:23   RD.SAMPC.SEQ.PLOG1                          ↩
   _      2007-06-01  17:51:32   RD.SAMPC.SEQ.PLOG1                          ↩
   _      2007-06-01  17:51:39   RD.SAMPC.SEQ.PLOG1                          ↩
   _      2007-06-01  17:51:46   RD.SAMPC.SEQ.PLOG1                          ↩
                                                                              ↩
                                                                              ↩
 Command ==>                                                                  ↩
                                                                              ↩
                                                                              ↩
                                                                              ↩
 Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
       Help        Exit                    -     +                       Menu ↩
  ↩
```

3 The following information is displayed on this screen:

Field	Description
Date	The date of the first transaction in the PLOG data set.
PLOG Data Set Name	The name of a PLOG data set for which replay processing can be initiated.
Replay	Enter an "I" in this column to review more information about this PLOG data set. Enter a "D" in this column if you want to delete this PLOG data set entry from the list of PLOG data sets in the Replicator system file *as well as* all entries for PLOG data sets with transaction dates and time earlier than the deleted PLOG data set entry.
Time	The time of the first transaction in the PLOG data set.

Reviewing PLOG Information

You can review more information about each PLOG data set. This information can help you determine whether you want to replay records from the data set and whether or not you want to remove the PLOG data set entry from the list of PLOG data sets in the Replicator system file.

To review more information about a PLOG data set:

1 Access the list of PLOG data sets from which replay processing can be requested, as described in ***Listing the PLOG Data Sets***, earlier in this section.

2 Locate the PLOG data set for which you want more information, type an "I" in the corresponding **Replay** column, and press Enter.

The Replication Replay - PLOG Information screen appears.

```
14:53:47      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                    Replication Replay - PLOG Information           M-RP2056




   PLOG Data Set Name ......... RD.SAMPC.SEQ.PLOG1
   Start Date and Time ........ 2007-06-01 17:49:26
   End Date and Time .......... 2007-06-01 17:49:26
   PLOG Session Number ........          5
   From Block Number ..........        125
   To Block Number ............        127
   Last PLOG in session .......          N









Command ==>


Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu
```

The following information is provided on this screen:

Screen Field	Description
PLOG Data Set Name	The name of the PLOG data set.
Start Date and Time	The date and time of the first transaction in the PLOG data set.
End Date and Time	The date and time of the last transaction in the PLOG data set.
PLOG Session Number	The number of the PLOG session for this PLOG data set.
From Block Number	The starting block number in this PLOG data set.
To Block Number	The ending block number in this PLOG data set.
Last PLOG in session	Indicates whether or not the PLOG data set is the last PLOG data set in the session. A value of "Y" in this field indicates that it is the last PLOG data set in the session; a value of "N" indicates that it is not the last PLOG data set in the session.

Deleting PLOG Data Set Entries From the List

You may find it useful to delete PLOG data set entries from the list of PLOG data sets -- especially if the list gets too long and the Replicator system file starts to fill up. For example, very old PLOG data sets appearing in the list may no longer be of use to you. In this case, you can use this screen to remove their entries from the list.

Notes:

1. When you delete a PLOG data set entry from the list, all earlier PLOG data set entries are also deleted.
2. When you delete a PLOG data set entry from the list, the PLOG data set still exists. Only its entry in the Replicator system file is deleted.

To delete PLOG data set entries from the list of PLOG data sets:

1 Access the list of PLOG data sets from which replay processing can be requested, as described in *Listing the PLOG Data Sets*, earlier in this section.

2 Locate the latest PLOG data set entry you want to delete, type a "D" in the corresponding **Replay** column, and press Enter.

The PLOG data set entry and all earlier PLOG data set entries are removed from the list.

13 Submitting Event Replicator Target Adapter Requests

You can use the Adabas Event Replicator Subsystem to submit requests to the Event Replicator Target Adapter when it is activated. Ordinarily, once the Event Replicator Target Adapter processing is activated, replicated transaction data for the subscriptions and destinations are transformed and transferred to the Event Replicator Target Adapter to be applied to your RDBMS. In addition, though, you can request that the Event Replicator Target Adapter:

- Initiate a request for initial-state data for a specific subscription, database, Adabas file, and data values to populate the RDBMS tables.
- Clear the data in an RDBMS table, based on a specific subscription, database, and Adabas file.
- Delete the RDBMS tables for a specific subscription, database, and Adabas file.

Notes:

1. None of these requests can be processed if the Event Replicator Target Adapter is not fully activated. For complete information on fully activating Event Replicator Target Adapter processing, read *Activating Event Replicator Target Adapter Processing*, in *Event Replicator Target Adapter User Guide*.
2. Be sure you have the proper authorization privileges to maintain the RDBMS tables. User authorization to maintain any new RDBMS tables via Event Replicator Target Adapter is inherited from the site's privilege settings for the database. Authorization is managed by the user's RDBMS privileges and not by Event Replicator Target Adapter. Event Replicator Target Adapter does not grant RDBMS privileges to the user. Therefore, if you want to use Event Replicator Target Adapter to maintain tables in an RDBMS, verify that your RDBMS authorization privileges are correct for the maintenance you want to perform.

This chapter describes the methods by which you can control Event Replicator Target Adapter behavior.

Populating the RDBMS Tables

Using the Adabas Event Replicator Subsystem, you can request that the Event Replicator Target Adapter initiate a request to populate the RDBMS tables with initial-state data for specific subscriptions, databases, Adabas files, and data values. When you submit the RDBMS population request from the Adabas Event Replicator Subsystem, it is sent to the Event Replicator Target Adapter, which receives it and submits its own initial-state request to the Event Replicator for the data. The data is then transferred to the RDBMS via the usual processing of the Event Replicator Target Adapter.

Note: If you submit a request to the Event Replicator Target Adapter to populate an RDBMS table that does not exist, Event Replicator Target Adapter will create the table in the course of its normal processing. However, if you submit a request to the Event Replicator Target Adapter to populate an existing RDBMS table, it is your responsibility to clear or drop the

existing table prior to populating the RDBMS table with new data. For information on clearing an RDBMS table, read *Clearing (Refreshing) the RDBMS Table Data*, elsewhere in this guide; for information on dropping an RDBMS table, read *Deleting (Dropping) RDBMS Tables*, elsewhere in this guide.

To submit a request to the Event Replicator Target Adapter to populate the RDBMS:

The Event Replicator Target Adapter must be installed, configured, and started or the request will not be processed until it is. For more information on installing and starting the Event Replicator Target Adapter, read *Event Replicator Target Adapter Installation*, *Event Replicator Target Adapter Administration*, and *Starting the Event Replicator Target Adapter*, in your Event Replicator Target Adapter documentation.

1 Select option **A** from the Adabas Event Replicator Subsystem Main Menu.

The Administration menu appears.

```
14:53:47      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                                 Administration                      M-RP1100


                        Code    Function
                        ----    --------------------------------
                          D     Database ID
                          I     Perform Initial-State
                          P     PLOG Information
                          R     Initiate Replay
                          S     System Functions
                          T     Target Adapter
                          V     Global Values
                          ?     Help
                          .     Exit
                        ----    --------------------------------

                    Code ...  _


Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu  ↩
 ↩
```

2 Select option **T** on the Administration menu.

The Target Adapter menu appears.

```
18:46:03      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                              Target Adapter                        M-RP3010


                        Code    Function
                        ----    --------------------------------
                          P     Populate
                          R     Refresh
                          D     Drop
                          ?     Help
                          .     Exit
                        ----    --------------------------------




                 Code ...  _



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu  ↩
 ↩
```

3 Select **P** on the Target Adapter menu.

The Target Adapter - Populate screen appears.

```
18:46:23      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                               Target Adapter                        M-RP3020

                                  Populate

                  Initial-State .................. _________
                  DBID ........................... _____
                  File ........................... _____
                  Subscription ................... _________
                  Perform Drop Table ............. N

                                Value Buffer
               ------------------------------------------------
                 ____________________________________________
                 ____________________________________________
                 ________________________________________




Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Exec                                      Menu  ↩
 ↩
```

4 Supply values for the fields on this screen, as described in the following table:

Field Name	Required?	Description
Initial-State	Yes	The name of the initial-state definition in the Replicator system file that you want used to populate the RDBMS.
DBID	Yes	The DBID of the Adabas database you want used to populate the RDBMS.
File	Yes	The number of the file in the Adabas database you want used to populate the RDBMS.
Perform Drop Table	No	Indicate whether a request to the Event Replicator Target Adapter to delete the tables (Drop) in the RDBMS should be performed before the RDBMS is populated (Populate). Valid values are "Y" (Yes) or "N" (No); the default is "N".
Subscription	Yes	The name of the subscription definition in the Replicator system file that you want used to populate the RDBMS.
Value Buffer	No	This field is not available for use at this time.

5 When values for all required fields have been supplied, press PF5.

The request is submitted to the Event Replicator Target Adapter.

Clearing (Refreshing) the RDBMS Table Data

Using the Adabas Event Replicator Subsystem, you can request that the Event Replicator Target Adapter initiate a request to clear the data in the RDBMS tables for specific subscriptions, databases, and Adabas files. When you submit the RDBMS refresh request from the Adabas Event Replicator Subsystem, it is sent to the Event Replicator Target Adapter, which receives it and clears the data out of the related tables, leaving the tables themselves in place.

To submit a request to the Event Replicator Target Adapter to clear data in the RDBMS:

The Event Replicator Target Adapter must be installed, configured, and started or the request will not be processed until it is. For more information on installing and starting the Event Replicator Target Adapter, read *Event Replicator Target Adapter Installation*, *Event Replicator Target Adapter Administration*, and *Starting the Event Replicator Target Adapter*, in your Event Replicator Target Adapter documentation.

1 Select option **A** from the Adabas Event Replicator Subsystem Main Menu.

The Administration menu appears.

```
14:53:47      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                              Administration                         M-RP1100


                        Code    Function
                        ----    -------------------------------
                          D     Database ID
                          I     Perform Initial-State
                          P     PLOG Information
                          R     Initiate Replay
                          S     System Functions
                          T     Target Adapter
                          V     Global Values
                          ?     Help
                          .     Exit
                        ----    -------------------------------

                 Code ...  _


Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu  ↩
 ↩
```

2 Select option **T** on the Administration menu.

The Target Adapter menu appears.

```
18:46:03      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                              Target Adapter                         M-RP3010


                        Code    Function
                        ----    --------------------------------
                          P     Populate
                          R     Refresh
                          D     Drop
                          ?     Help
                          .     Exit
                        ----    --------------------------------




                 Code ...  _



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu  ↩
 ↩
```

3 Select **R** on the Target Adapter menu.

The Target Adapter - Refresh screen appears.

```
18:48:02      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                              Target Adapter                        M-RP3021

                                 Refresh


                    DBID ............................ _____
                    File ............................ _____
                    Subscription .................... ________









Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
```

4 Supply values for the fields on this screen, as described in the following table:

Field Name	Required?	Description
DBID	Yes	The DBID of the Adabas database associated with the RDBMS data you want to cleared.
File	Yes	The number of the file in the Adabas database of the file associated with the RDBMS data you want to cleared.
Subscription	Yes	The name of the subscription definition in the Replicator system file associated with the RDBMS data you want cleared.

5 When values for all required fields have been supplied, press PF5.

The request is submitted to the Event Replicator Target Adapter.

Deleting (Dropping) RDBMS Tables

Using the Adabas Event Replicator Subsystem, you can request that the Event Replicator Target Adapter delete the RDBMS tables (and their associated data) for specific subscriptions, databases, and Adabas files. When you submit the RDBMS refresh request from the Adabas Event Replicator Subsystem, it is sent to the Event Replicator Target Adapter, which receives it and processes it.

To submit a request to the Event Replicator Target Adapter to delete tables in the RDBMS:

The Event Replicator Target Adapter must be installed, configured, and started or the request will not be processed until it is. For more information on installing and starting the Event Replicator Target Adapter, read *Event Replicator Target Adapter Installation*, *Event Replicator Target Adapter Administration*, and *Starting the Event Replicator Target Adapter*, in your Event Replicator Target Adapter documentation.

1 Select option **A** from the Adabas Event Replicator Subsystem Main Menu.

The Administration menu appears.

```
14:53:47      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                                 Administration                      M-RP1100


                          Code    Function
                          ----    -------------------------------
                            D     Database ID
                            I     Perform Initial-State
                            P     PLOG Information
                            R     Initiate Replay
                            S     System Functions
                            T     Target Adapter
                            V     Global Values
                            ?     Help
                            .     Exit
                          ----    -------------------------------

                       Code ...  _


Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu  ↩
 ↩
```

2 Select option **T** on the Administration menu.

The Target Adapter menu appears.

```
18:46:03      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                              Target Adapter                        M-RP3010


                        Code    Function
                        ----    --------------------------------
                          P     Populate
                          R     Refresh
                          D     Drop
                          ?     Help
                          .     Exit
                        ----    --------------------------------




                 Code ...  _



Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                                 Menu  ↩
 ↩
```

3 Select **D** on the Target Adapter menu.

The Target Adapter - Drop screen appears.

```
18:49:55      ***** A D A B A S  EVENT REPLICATOR SUBSYSTEM *****    2013-02-28
                              Target Adapter                         M-RP3021

                                   Drop


                    DBID ........................... _____
                    File ........................... _____
                    Subscription ................... ________









Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Exec                                    Menu  ↩
 ↩
```

4 Supply values for the fields on this screen, as described in the following table:

Field Name	Required?	Description
DBID	Yes	The DBID of the Adabas database associated with the RDBMS tables you want to delete.
File	Yes	The number of the file in the Adabas database of the file associated with the RDBMS tables you want to delete.
Subscription	Yes	The name of the subscription definition in the Replicator system file associated with the RDBMS tables you want to delete.

5 When values for all required fields have been supplied, press PF5.

The drop request is submitted to the Event Replicator Target Adapter for processing.

Index

F

G

H

I

L

M

N

P

R

S

T

U

W