iWay Software

An Information Builders Company

iWay

iWay Server Administration for UNIX, Windows,
OpenVMS, OS/400, OS/390, and z/OS
Version 5 Release 3.2

DN3501278.0904

Preface

This documentation describes how to use the Web Console to configure, operate, monitor, tune, and troubleshoot the iWay Server. It is intended for server administrators, database administrators, and application developers. This documentation is part of the iWay Server documentation set.

How This Manual Is Organized

This manual includes the following chapters:

Chapter		Contents
1	Introduction	Describes how to use the console and provides an overview of server configuration.
2	Server Security	Describes security modes and their authentication process.
3	Configuring Data Adapters	Provides general information on configuring adapters and outlines configuration options.
4	Configuring a Remote Server	Contains an item-by-item description of the Web Console Remote Servers page, with reference topics that provide information on server parameters.
5	Accessing Application Files	Describes how to work with Application Files, how to control the APP environment, and how to use APP methods to simplify the process of moving a user application from one platform to another.
6	Managing Metadata	Discusses how to create a synonym using the Web Console.
7	Editing and Running Procedures	Explains how using procedures enables application logic to be written once and executed many times, and how the Procedures page enables the creation and testing of stored procedures.
8	Configuring Resource Analyzer and Resource Governor	Describes how to set up the server to use two add-on tools—Resource Analyzer and Resource Governor—for monitoring and controlling data usage activity. This chapter applies only to users who have installed Resource Analyzer and Resource Governor.

Chapter		Contents
9	DataMigrator Server Configuration	Describes configuration of a DataMigrator Server. This chapter applies only to users who have installed a DataMigrator Server.
10	Running and Monitoring Your Server	Describes how to run and monitor your server.
11	Managing Listeners and Special Services	Describes how to manage listeners and special services.
12	Troubleshooting	Explains how to perform problem analysis tasks.

Documentation Conventions

The following conventions apply throughout this manual:

Convention	Description
`THIS TYPEFACE` or `this typeface`	Denotes syntax that you must enter exactly as shown.
`this typeface`	Represents a placeholder (or variable) in syntax for a value that you or the system must supply.
`underscore`	Indicates a default setting.
this typeface	Represents a placeholder (or variable) in a text paragraph, a cross-reference, or an important term. It may also indicate a button, menu item, or dialog box option you can click or select.
this typeface	Highlights a file name or command in a text paragraph that must be lowercase.
Key + Key	Indicates keys that you must press simultaneously.
`{ }`	Indicates two or three choices; type one of them, not the braces.
`[ ]`	Indicates a group of optional parameters. None are required, but you may select one of them. Type only the parameter in the brackets, not the brackets.

Convention	Description
|	Separates mutually exclusive choices in syntax. Type one of them, not the symbol.
. . .	Indicates that you can enter a parameter multiple times. Type only the parameter, not the ellipsis points (…).
. . .	Indicates that there are (or could be) intervening or additional commands.

Related Publications

To view a current listing of our publications and to place an order, visit our World Wide Web site, http://www.iwaysoftware.com. You can also contact the Publications Order Department at (800) 969-4636.

Customer Support

Do you have questions about iWay Server Administration for UNIX, Windows, OpenVMS, OS/400, OS/390, and z/OS?

Call Information Builders Customer Support Service (CSS) at (800) 736-6130 or (212) 736-6130. Customer Support Consultants are available Monday through Friday between 8:00 a.m. and 8:00 p.m. EST to address all your iWay Server Administration for UNIX, Windows, OpenVMS, OS/400, OS/390, and z/OS questions. Information Builders consultants can also give you general guidance regarding product capabilities and documentation. Please be ready to provide your six-digit site code (*xxxx.xx*) when you call.

You can also access support services electronically, 24 hours a day, with InfoResponse Online. InfoResponse Online is accessible through our World Wide Web site, http://www.informationbuilders.com. It connects you to the tracking system and known-problem database at the Information Builders support center. Registered users can open, update, and view the status of cases in the tracking system and read descriptions of reported software issues. New users can register immediately for this service. The technical support section of www.informationbuilders.com also provides usage techniques, diagnostic tips, and answers to frequently asked questions.

To learn about the full range of available support services, ask your Information Builders representative about InfoResponse Online, or call (800) 969-INFO.

Information You Should Have

To help our consultants answer your questions most effectively, be ready to provide the following information when you call:

- Your six-digit site code (*xxxx.xx*).
- Your iWay Software configuration:
    - The iWay Software version and release.
    - The communications protocol (for example, TCP/IP or LU6.2), including vendor and release.
- The stored procedure (preferably with line numbers) or SQL statements being used in server access.
- The database server release level.
- The database name and release level.
- The Master File and Access File.
- The exact nature of the problem:
    - Are the results or the format incorrect? Are the text or calculations missing or misplaced?
    - The error message and return code, if applicable.
    - Is this related to any other problem?
- Has the procedure or query ever worked in its present form? Has it been changed recently? How often does the problem occur?
- What release of the operating system are you using? Has it, your security system, communications protocol, or front-end software changed?
- Is this problem reproducible? If so, how?
- Have you tried to reproduce your problem in the simplest form possible? For example, if you are having problems joining two data sources, have you tried executing a query containing just the code to access one data source?
- Do you have a trace file?
- How is the problem affecting your business? Is it halting development or production? Do you just have questions about functionality or documentation?

User Feedback

In an effort to produce effective documentation, the Documentation Services staff welcomes your opinions regarding this manual. Please use the Reader Comments form at the end of this manual to relay suggestions for improving the publication or to alert us to corrections. You can also use the Documentation Feedback form on our Web site, http://www.iwaysoftware.com.

Thank you, in advance, for your comments.

iWay Software Training and Professional Services

Interested in training? Our Education Department offers a wide variety of training courses for iWay Software and other Information Builders products.

For information on course descriptions, locations, and dates, or to register for classes, visit our World Wide Web site (http://www.iwaysoftware.com) or call (800) 969-INFO to speak to an Education Representative.

Interested in technical assistance for your implementation? Our Professional Services department provides expert design, systems architecture, implementation, and project management services for all your business integration projects. For information, visit our World Wide Web site (http://www.iwaysoftware.com).

iWay Software Training and Professional Services

Contents

CHAPTER 1

Introduction

Topics:

- Using the Web Console
- Configuration Overview
- Preparing for Communications Configuration
- Running and Configuring the New FOCUS Database Server (OS/390 and z/OS Only)
- Server Profiles
- Profile Commands

Release 5 introduces an expanded Web Console that enables you to configure, operate, monitor, tune, and troubleshoot your server from a single, easy-to-use interface.

Using the Web Console

The HTTP Web Console enables you to remotely view and manage the server environment. From a single, easy-to-use interface, you can:

- Select, add, and configure data adapters.
- Create and manage adapter metadata.
- Configure remote servers.
- Configure, edit, and run applications and deferred query processing.
- Configure communications and special services.
- Operate, monitor, tune, and troubleshoot your server.
- Edit configuration files.
- Migrate from a previous server release.

Procedure: How to Open the Web Console

To open the Web Console:

1. Start the server.
2. Type the following URL in the address space of your Web browser

   ```
   http://ip_address:http_service
   ```

   where:

   ip_address

   Is the IP address of the machine on which the server is installed.

   http_service

   Is the value for HTTP Service entered during the server configuration procedure.

   Note: If you have trouble contacting the server, see *Troubleshooting the Console* in Chapter 12, *Troubleshooting*.

3. If you are running your server with security on, type the user name and password used to access the operating system.

   Logging on with the server administrator ID activates the management features and the diagnostic pages. Non-administrator IDs can view only status and statistics, and run the test tools. The non-administrator pages are similar to the administrator pages, but they hide the features required to manage the server. The administrator versions of the pages appear in this manual.

   The Web Console Home Page appears.

The navigation pane on the left provides access to the different features of the Web Console. The top of the navigation pane displays the last date and time of communication between the browser and the server.

4. Select an option from the navigation pane to access the corresponding console page. For tasks that display information or require no additional navigation, the console displays corresponding information in the right pane. For tasks that require additional, task-centered navigation, the console opens a new window. Use the new window's navigation pane on the left to access the corresponding information in its right pane.

The console offers online help by providing two kinds of links to appropriate sections of this documentation:

- Clicking Help on the navigation pane provides general help in the context of the current page.
- Clicking the ? icon, when available, provides contextual help specific to the associated item.

Configuration Overview

The characteristics of an individual server are defined by a set of configuration files. These configuration files define the protocols, services, and data sources supported by the server. In the UNIX, OS/390 USS, Windows, OpenVMS, and OS/400 environments, these files are maintained in directories designated by the environment variable EDACONF.

The initial installation procedure creates an installation instance, represented in manuals as the logical name EDAHOME, and one default configuration instance, known as EDACONF. After you successfully install the server, you can configure a default configuration instance to create an operational instance, or create an additional one at your site. An operational instance of the server is one that is configured to support specific protocols, services, and access to data sources.

You must run the Installation/Configuration Utility for each new instance of the server that you want to configure, and then use the Web-based Administration Console to configure all necessary functionality.

To configure your server instance, use the Web Console to:

1. Select and configure data adapters.
2. Optionally configure remote servers.
3. Configure communications nodes and protocols.
4. Optionally set parameters for deferred query processing.

If necessary, you may edit configuration files. Some configuration errors can make the server start in a limited mode called safe mode, in which the Administration Console is still operational to enable you to correct the errors.

Preparing for Communications Configuration

Depending on your environment and network configuration, inter-node communications may consist of TCP/IP, HTTP, SNA (LU6.2), or PIPE communication protocols. Inter-node communications are configured through the Listeners and Remote Servers links in the navigation pane of the Web Console.

Procedure: How to Prepare to Configure TCP/IP or HTTP

To prepare to configure for TCP/IP or HTTP:

1. Verify that TCP/IP is installed, configured, and running on both the server and client platforms.
2. List any TCP service names/port numbers and IP addresses you are using for inbound server and outbound client or subserver access.

   This list is used during the server configuration.

Procedure: How to Prepare to Configure SNA (LU6.2)

To prepare to configure for SNA (LU6.2):

1. Verify that SNA (System Network Architecture) is installed, configured, and running on both the server and client platforms.
2. Ensure that all installation prerequisites are met.

   SNA (LU6.2) is not supported on all platforms.

   For more information, see the *Server Installation* manual.
3. List all SNA profiles and parameters you are using for inbound server and outbound subserver access.

   This list is used during the server configuration.

4. Create an SNA configuration for intra-node communications.

   Depending on your platform, you create some or all of the following SNA (LU6.2) definitions for the server:

   - Local LU name.
   - Partner LU and Partner LU Alias.
   - Partner LU and Partner LU Alias for local loop back.
   - Mode Name. The default is PARALLEL. Do not change this parameter unless you are instructed to do so by the site administrator.
   - TP Name. The default for an OS/390 or z/OS subserver is MVSSRVR. Do not change this parameter unless you are instructed to do so by the site administrator.

Procedure: How to Configure PIPE Communications

No preparation is required for PIPE communications.

Note: The client and the server must reside on the same physical machine.

Communications Configuration Worksheets

Example:

Configuring for Inbound Communications Using TCP/IP

Configuring for Inbound Communications Using SNA (LU6.2)

Configuring for Outbound Communications Using TCP/IP

Configuring for Outbound Communications Using SNA (LU6.2)

Complete one or several of the remaining worksheets, depending on your:

- Communications protocol.
- Inbound or outbound communications.
- MVS Remote Server (subserver) access.

In the following examples, the terms inbound and outbound are used relative to the server; inbound refers to communications from the client, and outbound refers to communications to remote subservers.

Example: Configuring for Inbound Communications Using TCP/IP

If you choose to configure for inbound communications using TCP/IP, complete this worksheet.

Prompt Description	**Default**	**Supply Your Value Here**
Inbound TCP/IP Port Number.	Depends on server type. The Full-Function Server default is 8100.	

Note: This port should be the first of up to five consecutive ports that the server uses.

Example: Configuring for Inbound Communications Using SNA (LU6.2)

If you choose to configure for inbound communications using SNA (LU6.2), complete this worksheet.

Prompt Description	**Default**	**Supply Your Value Here**
TP NAME for server.	EDATP	
LOCAL LU NAME for local test client test tool (RDAAPP).	None	
PARTNER LU NAME for local client test tool (RDAAPP).	None	
MODE NAME for local test client test tool (RDAAPP).	None	

Example: Configuring for Outbound Communications Using TCP/IP

If you choose to configure for outbound communications using TCP/IP (Hub Server and Full-Function Server with Hub Services only), complete this worksheet.

		Supply Your Subserver Values Here		
Prompt Description	**Default**	**Sub1**	**Sub2**	**Sub3**
Outbound host name or IP address.	Local host name			
Outbound TCP/IP service name or port number.	8100			

		Supply Your Subserver Values Here		
Prompt Description	**Default**	**Sub1**	**Sub2**	**Sub3**
User ID and password for connecting to the server on this port.	Leave blank for trusted node access			
Is server located on MVS? If Yes, type the MVS service name.	No			

Example: Configuring for Outbound Communications Using SNA (LU6.2)

If you choose to configure for outbound communications using SNA (LU6.2) (Hub Server and Full-Function Server with Hub Services only), complete this worksheet.

Prompt Description	Default	Supply Your Value Here
LOCAL LU NAME used by the server.		
PARTNER LU NAME for the Remote Server.	None	
TP NAME for the Remote Server.	None	
PARTNER MODE NAME used by the client test tool (RDAAPP).	None	
User ID and password for connecting to the server on this port.	Leave blank for trusted node access.	
Located on MVS (Yes or No)? If Yes, type the MVS service name.	No	

Running and Configuring the New FOCUS Database Server (OS/390 and z/OS Only)

As of Version 5 Release 3, the iWay Server for OS/390 and z/OS comes with its own FOCUS Database Server, also called a Sink Machine. You can either use this new FOCUS Database Server or keep the FOCUS files on the legacy Focus Database Server, which runs as a separate batch job.

If you choose to use the new FOCUS Database Server, the implementation of the USE command is left unchanged. The FOCSBS value is now a node name in the server's communications configuration file, odin.cfg.

Node FOCSU01 is used to identify the new FOCUS Database Server, and so cannot be used for any other use.

Note that a single file can not be in both Legacy and New Focus Database Servers. You must select one or the other to manage access, or you will encounter queue conflicts.

Example: Using the New FOCUS Database Server with a DYNAM Statement

Using the Web Console, edit the suprof.prf profile ALLOC allocations for each file to be managed. For example, add following command on suprof.prf profile:

```
DYNAM ALLOC FI CAR DA <dsname> SHR REU
```

The following USE command will enable the FOCUS Database Server to control access:

```
USE CAR ON FOCSU01
END
```

Example: Using the New FOCUS Database Server with a JCL Statement

ISTART JCL allocation:

Add JCL statements for the FOCUS files to be managed

```
   //CAR          DD DISP=SHR,DSN=<dsname>
```

The following USE command will enable the FOCUS Database Server to control access:

```
USE CAR ON FOCSU01
END
```

Example: Using the New FOCUS Database Server with a DYNAM Statement

DYNAM ALLOC command with keyword NOALIAS is equivalent to have a JCL ALLOCATION on ISTART JCL and it makes a global server allocation, available to the Focus Database Server. For example, use the following DYNAM ALLOC statement on edasprof.prf or user procedure (focexec):

```
DYNAM ALLOC FI CAR DA <dsname> SHR REU NOALIAS
```

The following USE command will enable the FOCUS Database Server to control access:

```
USE CAR ON FOCSU01
END
```

Example: Using Legacy FOCUS Database Server (Separated Batch Job)

The following sample code identifies a legacy FOCUS Database Server via odin.cfg node block FOCSBS.

```
USE
CAR ON FOCSBS
END
```

This sample node block resides in odin.cfg and can be constructed using the Web Console.

```
NODE=FOCSBS  <===== USS server (odin.cfg file)
BEGIN
    PROTOCOL=SBS
    CLASS=SUCLIENT
    PORT=X.Y.Z <====== communications dataset
 END
```

Server Profiles

You can create profiles to customize the server environment for all users, a category of users, or an individual user. Server customization is achieved through profile level and profile search order.

A profile can include almost any command that a client application can send to the server, from an environment SET command to a data retrieval command. When you create a profile, take into account that the more processing performed by the profile, the more time it takes for an application to connect to the server. An exception is the case of pooled deployment, in which only one profile is processed upon server agent initial startup or refresh.

The server processes all profiles found in the search order. If it finds duplicate settings or commands, the last setting or command processed will be active for the connection.

iWay creates a standard global profile, edasprof.prf, during installation. Creation of other profiles is optional.

Profile Level

The server supports various levels of profiles to provide flexibility in designing and running production applications.

- **Global profile**

  The first level of profile, the global profile, is a startup file automatically created during installation and configuration of the server. It contains default environment settings required for the correct operation of the server.

  The global profile remains in effect throughout a user session. You can modify the global profile default settings. You can also add any commands or code that all connected users require before application processing begins.

Characteristic	**Description**
Name	edasprof.prf
Location	UNIX: $EDACONF/etc directory Windows: $EDACONF\etc directory OS/400: IFS $EDACONF/etc directory OpenVMS: EDACONF [.ETC] directory
Override	Set the environment variable EDASPROF to point to an alternate location of edasprof.prf (you cannot rename edasprof.prf).
Sequence in search order	First
Scope	Applies to all connected users.

- **Service profile**

  A service profile specifies settings for the server environment, but the settings in this level of profile apply only to users associated with a specific service. When a user connects to the server, the service profile settings are applied and remain in effect throughout the user session.

  A service profile may contain settings that are the same as those in a global profile. You can specify a service profile in the Web Console by choosing *Workspace* and *Configure* from the navigation panel, and *Service* at the top of the next window.

You can create a service profile using the Web Console Procedures option or any standard system editor. If you use an editor, create the service profile in the correct location.

Characteristic	Description
Name	*name*.fex
Location	Follows the normal search path for procedures (focexecs).
Override	None
Sequence in search order	Second
Scope	Applies to all users of the same service.

- **Group profile**

  A group profile is not available on Windows.

  A group profile specifies settings for the server environment, but the settings in this level of profile apply only to users associated with a specific security group. When a user connects to the server, the group profile settings are applied and remain in effect throughout the user session.

  This level of profile applies only if security is ON.

  A group profile may contain settings that are the same as those in a global profile. You can manually create a group profile at any time using any operating system editor.

  On z/OS UNIX System Services, a user can be a member of many security groups but iWay uses only one group for profile processing. For the group profile, iWay uses the group ID placed in the security control block (ACEE) by the security package established upon successful user ID validation. For the Resource Access Control Facility (RACF), iWay uses the default group that is returned.

You can create a group profile using any standard system editor. Create the group profile in the correct location.

Characteristic	**Description**
Name	*group_name*.prf
Location	UNIX: *base_directory_name*/ibi/profiles Windows: *base_directory_name*\ibi\profiles OS/400: IFS $EDACONF/etc directory OpenVMS: EDACONF [.ETC] directory
Override	Set the edaserve.cfg keyword edaprfu to point to an alternate location of *group_name*.prf.
Sequence in search order	Third
Scope	Applies to all users of the same security group.

- **User profile**

  A user profile specifies settings for the server environment, but the settings in this level of profile apply only to a specific user ID. When a user connects to the server, the user profile settings are applied and remain in effect throughout the user session.

  A user profile may contain settings that are the same as those in a global profile.

  You can create a user profile at any time using one of the following options on the Web Console:

  - Workspace/Edit Files/User Profile for
  - Data Adapters/Configure

You can also use any standard system editor. If you use an editor, create the user profile in the correct location.

Characteristic	**Description**
Name	*user_id*.prf
Location	UNIX: *base_directory_name*/ibi/profiles Windows: *base_directory_name*\ibi\profiles OS/400: IFS $EDACONF/etc directory OpenVMS: EDACONF [.ETC] directory
Override	Set the edaserve.cfg keyword edaprfu to point to an alternate location of *user_id*.prf.
Sequence in search order	Fourth
Scope	Applies to a specific user.

Profile Commands

The following topics describe commands that you can include in any of the supported server profiles. These commands affect the behavior of the server for the duration of the connected session. You can code additional commands, such as data access commands, in any of the supported server profiles. For more information, see the specific data adapter topics in this manual.

Note: For FOCUS Table Services, the server profiles play the same role as the FOCUS profile.

Profile Command Formats

Each server command takes one of three possible formats. The syntax and an example of these formats follow. Ensure that you use the correct format for any server command you use.

Syntax: How to Use a Direct SET Command

To use a direct SET command

```
SET command=value
```

where:

command

Is the server command.

value

Is the value you select from the available choices.

Example: Using a Direct SET Command

The following is an example of how to use a direct SET command:

```
SET SQLENGINE=DB2
```

Syntax: How to Use the SQL Engine SET Command

To use the SQL Engine SET command

```
ENGINE [sqlengine] SET command value
```

where:

sqlengine

Is the data source. You can omit this parameter value if you previously issued the SET SQLENGINE command.

command

Is the server command.

value

Is the value you select from the available choices.

Example: Using the SQL Engine SET Command

The following is an example of how to use the SQL Engine SET command:

```
ENGINE SQLORA SET OWNERID EDAUSER
```

Syntax: How to Use an SQL Translator Command

To use an SQL Translator command

```
SQL
SET command=value
END
```

where:

command

Is the SQL Translator command.

value

Is the value you select from the available choices.

Example: Using an SQL Translator Command

The following is an example of how to use the SQL Translator command:

```
SQL
SET APT=OFF
END
```

Profile Commands Reference

The Server Profile Commands document provides the syntax for the profile commands. Individual data adapter topics include a description of the applicable profile commands for that data adapter.

CHAPTER 2

Server Security

Topics:

- Security Modes
- Security ON
- Security PTH
- Security DBMS
- Security LDAP
- Security OFF
- Security Scope

This section describes the security modes in which a server can run and the authentication process for each mode. It also describes how an administrator enables a particular security mode.

Part of security implementation is scope. A user can have access to the Web Console, or access to all the resources provided by the server.

Security Modes

There are several possible settings for server security, which can be configured with the EDAEXTSEC environment variable:

- **Security ON:** Each connecting user is defined in the operating system on which the server is running. The server uses operating system services to authenticate connecting users, impersonate them, and ensure access control to resources such as files and DBMS objects. Access to the Web Console's administrative functions is protected by user authentication at the operating system level.
- **Security PTH:** Access to the Web Console is controlled by authentication against the user list defined at the configuration level (user IDs and passwords have to be configured with the server_admin_id keyword in edaserve.cfg). No control of the data resources is possible.
- **Security DBMS:** Access to the Web Console and data resources is controlled by authentication against the database list of user IDs (for example, Oracle users).
- **Security LDAP (Lightweight Directory Access Protocol):** Access to the Web Console and data resources is controlled by authentication through the established directory. This mode currently applies only to a server running under Windows.
- **Security OFF:** All user information is ignored, and there is no security restriction. Access to the Web Console is not protected.

Security ON

In this section:

To enable this security mode, the server process must first be granted sufficient privileges. Specifically:

- On UNIX, the startup tscom300 executable must have setuid of root.
- On Windows NT, the server must be started as a service under SYSTEM account.
- On z/OS, the MVS load library must be APF-authorized and hfs executables must be given +a option.
- On OS/400, the ownership of libraries must be changed to QSECOFR.
- On OpenVMS, the account starting the server must have a specific set of privileges. See the installation instructions for OpenVMS.

Secondly, this mode is enabled by exporting the environment variable EDAEXTSEC=ON, or leaving it unset (this is the default mode).

On Windows NT, the Server Administrator password is required for this security mode. If the password is not provided in the configuration, the server starts in safe mode and displays a message to that effect.

In this mode the user credentials from the client connection are authenticated via the operating system's native security system, and then the server allocates a data access agent that fully impersonates that user so that access control to files or other objects is guaranteed by the native system.

Authentication of the Client Connection

User credentials can be:

- **Explicit.** The client connection supplies both user ID and password.
- **IWA (Windows NT only).** The client and server exchange the security token, but not the password.
- **Trusted.** The client is trusted, and the server impersonates the user without further authentication.

File Access Control

Impersonation of the client user means that access to files is controlled by the operating system security features in the native file systems. Basically, files are protected according to the effect of their permissions for the impersonated user. Permissions for new files depend on how the server is installed and started.

There is some form of inheritance of permissions from the global server environment, but details depend on the particular operating system. The user who is being impersonated at object creation, combined with the default permissions inherited at the level at which the object is created, define whether and how other users will be able to access the object.

On UNIX, zOS, and OS/400, the umask setting of the process that started the server is inherited down to the creation of each subdirectory or file, and thus defines the permissions used for object creation.

On OpenVMS, the default file protection from the process that started the server is inherited and used to define permissions for new objects.

On Windows NT, the permissions are inherited down from directory to subdirectory, and the permissions for any object created are taken from its parent directory. So unless specific permissions are applied at any level, default permissions will depend on where the server was installed.

Consequences for Files Under EDAHOME/EDACONF Hierarchy

When defining the permissions under a specific security policy, care must be taken to ensure that all users of the server who may need it have read or execute access to files under that hierarchy. Usually it is not necessary to provide write permissions for all users in the server admin role because they will still be able to modify files through the Web Console since the HTTP listener runs as a super user.

Consequences for Files Under APPROOT Hierarchy

Care must be taken when defining default permissions for application administrators who will create files under the Web Console that need to be accessed by regular users. When application files are written under the Web Console, they are created according to the rules of ownership and default permissions applicable to the user who logged on to the Web Console.

Consequences for Files Under EDATEMP Hierarchy

In a server that runs with security on, special conditions apply to the edatemp subdirectory and its contents. The basic principle is that agent subdirectories under edatemp are owned at any given moment by the user that the agent is impersonating.

In private mode, this means that once a user connects to an agent, he is explicitly given ownership of his agent subdirectory, regardless of which user the agent impersonated during previous connections. This ownership and the inherited permissions thus define how other users can access the connected user's temporary files, if at all.

On UNIX, zOS, and OS/400, the edatemp subdirectory itself is an exception because it is set up similarly to the /tmp directory: while it allows any user to add files and directories underneath, only the creator of a file can later rename or delete it. Files directly under edatemp (for example, traces) and listener subdirectories are owned by the super user (root, QSECOFR).

On Windows NT, the permissions of the parent directory of edatemp (usually EDACONF), which were applied when the server was installed, are inherited when edatemp and then the agent subdirectories are created, defining the kind of access other users have to an agent subdirectory and its contents.

DBMS Access Control

User credentials for DBMS access control can be:

- **Explicit.** The user ID and password for a DBMS connection are explicitly supplied in edasprof.prf via SET CONNECTION_ATTRIBUTES.
- **Trusted.** The DBMS connection will use the same credentials that were used for impersonation. For these credentials to be carried over from the agent to the DBMS, the edasprof.prf file must contain an explicit empty user, as in:

  ```
  ENGINE SQLORA SET CONNECTION_ATTRIBUTES connection_name/,
  ```

- **Passthru.** The user ID and password coming from the client connection are passed to the DBMS connection when edasprof.prf does not contain any credentials for the DBMS connection.

Explicit and passthru are always available in any security mode, while trusted is specific to security ON since it relies on impersonation being recognized by the DBMS via native system security.

For details, see the documentation for specific data adapters.

In addition to access control provided by the DBMS, there is another level of control using Master File attributes that can restrict data access by user, file, and even data source fields.

Web Console Access Control (Security ON)

Access to the console is protected by user authentication at the operating system level. Only authenticated users matching a user name listed in server_admin_id are allowed control operations on the console, with the level of control depending on their definition in server_admin_id.

Security PTH

To run the server in this mode, export the environment variable EDAEXTSEC=PTH before startup.

In this mode, there is no impersonation or authentication at the level of the operating system. All the server processes run as a single user ID from the operating system point of view. There is no impersonation of data agents.

The Server Administrator password must be configured *before* starting a server in this security mode, either by providing it at installation time or by running the following command from the EDACONF/bin directory:

```
edastart -change server_admin_id
```

Web Console Access Control (Security PTH)

Access to the console is protected by user authentication that compares the incoming user ID and password with the list defined in edaserve.cfg. Only authenticated users matching a user name listed in server_admin_id are allowed control operations on the console, with the level of control depending on their definition in server_admin_id.

Security DBMS

To run the server in this mode, export the environment variable EDAEXTSEC=DBMS before startup.

In this mode there is no impersonation or authentication at the level of operating system security for any user credentials supplied by the client connection. Instead, the users may be defined on the DBMS server or iWay subserver. This technique is called passthru, as user IDs and passwords supplied by the client are passed to the next level for authentication.

Procedure: How to Configure DBMS Mode

1. Bring the server up in the default security mode (OFF).
2. From the Adapters page, configure the DBMS adapter and the connection that will pass the user ID to the DBMS.
3. Test the connection.
4. From the Workspace > Configure > Access Control option, do the following:
   a. Select *security_dbms* and *security_connection*.
   b. Add your DBMS user ID (for example, oracle1) to the server admin list on the SRV level. You do not need to store the password.
5. Save and restart.
6. Stop the server.
7. Start the server with DBMS mode. Now you will always need a valid DBMS user ID and password.
8. Log on to the Web Console using your DBMS user ID (for example, oracle1).
9. Test the adapter connections.

If an error is made in the security_dbms definition, if the DBMS is down, or if the DBMS user ID is invalid, you cannot access the Web Console. You need to bring the server down via the text console and correct the situation.

The server may have other DBMS connections in addition to the one used for security passthru.

All the server processes run as a single user ID from the operating system point of view. There is no operating system impersonation of data agents.

Security LDAP

LDAP (Lightweight Directory Access Protocol) security mode currently applies only to a server running under Windows.

In LDAP mode, the user credentials from the client connection are authenticated through the established directory.

To run the server in this mode, export the environment variable EDAEXTSEC=LDAP before startup.

There is no impersonation at the level of the operating system. All the server processes run as a single user ID from the operating system point of view. There is no impersonation of data agents.

You must configure the Server Administrator password before starting a server in this security mode, either by providing it during installation or by running the following command from the EDACONF/bin directory:

```
edastart -change server_admin_id
```

Security OFF

To run the server in this mode, omit the authorization steps during installation, or export the environment variable EDAEXTSEC=OFF before startup.

This mode is applicable when the installation does not need any security features.

The server will run unsecured (security OFF) if the desired security setting cannot be established for any reason. This feature allows the administrator to change the configuration through the Web Console and restart the server or stop it.

Security Scope

Access control for a given security mode is only enforced for resources included in the security_scope defined in the edaserve.cfg file. The scope can be limited to the Web Console only, or it can include all server resources (in which case data access connections are also subject to the same access control). See security_scope for more detail.

The following table reflects the supported combinations of security mode and security scope.

Security Mode	**security_scope=ALL**	**security_scope=WC**
Security ON	Supported	Not supported
Security PTH	Not supported	Supported
Security DBMS	Supported	Supported
Security LDAP (currently Windows only)	Supported	Supported
Security OFF	—	—

CHAPTER 3

Configuring Data Adapters

Topics:

The following section describes how to configure a data adapter using the Web Console.

Configuring a Data Adapter

Clicking *Data Adapters* in the Full-Function Server or DataMigrator Server navigation pane opens the Configuring Data Adapters page, which presents a list of data adapters available for a given operating system and operating system release.

Most of the data adapter configuration information is contained in two configuration files, EDASPROF.PRF and EDASERVE.CFG. EDASPROF.PRF or the global profile contains data adapter connection information, and EDASERVE.CFG contains database release and access method information.

An experienced Server Administrator can edit the global profile manually, but it is not recommended, as syntax may change from one release to another.

Select *Release and Access Method* and click to configure.

All Database Configuration pages have the same look and feel, but the parameters are specific to each database.

Changing the Adapter Configuration

The Data Adapters page displays a tree view of data adapters already configured for the current operational instance. Under the Configured subtree it lists each data adapter with all connections already configured.

Clicking a data adapter enables you to:

Function	Description
Add connection	Adds a connection for data adapters that have the functionality to support multiple connections.
Settings	Displays various settings in effect for the given adapter. This is for display only. **Note:** This only applies to adapters for relational DBMSs.
Remove	Removes adapter configurations and all connection settings from the list of configured adapters.

Any adapter definitions in user profiles must be maintained manually.

Clicking a connection enables you to:

Function	Description
Test	Runs select statement for name/owner against catalog table with limit up to 15 records.
Properties	Enables you to view and change all of the connection parameters. **Note:** Duplicate entries are not verified.
Delete	Deletes connection settings. **Note:** Entries are deleted from the server global profile, EDASPROF.PRF, but not from the workspace configuration file.

CHAPTER 4

Configuring a Remote Server

Clicking *Remote Servers* in the *Full-Function Server* or *DataMigrator Server* navigation pane opens the Configuring Remote Servers page, which enables you to add remote server nodes to the communications configuration. The interface to input parameters for each protocol is similar to the Listeners page except that the CLASS parameter is fixed to CLIENT.

Clicking *Remote Servers* in the *Application Adapter Server* navigation pane opens the Remote Servers page, which enables you to configure the Application Adapter Server Communications Nodes.

CHAPTER 5

Accessing Application Files

Topics:

- Application Files
- Controlling the APP Environment
- Accessing Metadata and Procedure Files
- Allocating Temporary Files
- Temporary Space Allocation for UNIX and z/OS UNIX System Services
- Temporary File Directory Structure
- Specifying the edatemp Variable
- Pre-allocation of Temporary Files
- Dynamically Allocating FOCUS Files on z/OS

An Application File (APP) is a platform-independent repository for the group of related files—procedure files, Master and Access Files, data files, PDFs, GIFs, and so on—which are part of a specific user application. Using APP methods simplifies the process of moving a user application from one platform to another.

Application Files

On most platforms (NT, UNIX, VMS), the Application File is a directory. It can be stored in the approot directory or it can be mapped to a directory anywhere in the file system. On z/OS, the application can be mapped to a set of allocated ddnames.

The physical location of the APP hierarchy is determined by a configuration parameter (approot= ...) in the server configuration file, edaserve.cfg. The default value for approot built during installation is ibi/apps under the installation directory. The default application is assumed to be baseapp under APPROOT.

The application name can be up to 18 characters long; the file name can be up to 64 characters long.

Controlling the APP Environment

The following platform-independent commands enable you to specify parameters for the APP environment:

APP Management

`APP CREATE` *`app1[, app2,...]`*	Creates directory app1[,app2, ...] under APPROOT.
`APP DELETE` *`app1[, app2, ...]`*	Deletes directory app1[, app2, ...] under APPROOT with its content.
`APP MAP` *`app1 directory1`*	Defines a virtual application app1 pointing to the physical directory direcorty1.
`APP MAP` *`app`* `[file_extension=//dd:ddname;...]`	For z/OS: Defines a virtual application app pointing to a group of DDNAMEs, linked with a file extension, and pointing to a PDS where files of that type are located. MVSAPP is a special APP created at installation time to point to a group of special ddnames -- FOCEXEC, MASTER, ACCESS, HTML, FOCSTYLE, GIF, FOCPSB -- should they be allocated by JCL or DYNAMs.
`APP RENAME` *`app1 app2`*	Renames application *app1* to *app2*.
`APP HOLDMETA` *`app`*	Designates *app* as the location for Master Files and Access Files created with the HOLD command.
`APP HOLDDATA` *`app`*	Designates *app* as the location for data files created with the HOLD command.

APP Paths Management

`APP PATH` *`app1 app2...appn`*	Sets the directory search path to *app1*, *app2*, ..., *appn*.
`APP PREPENDPATH` *`app`*	Prepends *app* to the list of directories already in APP PATH.
`APP APPENDPATH` *`app`*	Appends *app* to the list of directories already in APP PATH.
`APP PATH`	Resets the APP PATH directory list to the initial list.
`APP SHOWPATH`	Shows the search paths.

APP File Management

`APP COPY` *`app1 app2`*	Copies all files from directory *app1* under APPROOT into directory *app2*.
`APP COPYF[ILE]` *`app1 fn1 ft1 app2 fn2 ft2`*	Copies the file with *filename=fn1 filetype=ft1* from the application *app1* to a file with *filename=fn2 filetype=ft2* in the application *app2*.
`APP MOVEF[ILE]` *`app1 fn1 ft1 app2 fn2 ft2`*	Moves the file with *filename=fn1 filetype=ft1* from the application *app1* to a file with *filename=fn2 filetype=ft2* in the application *app2*.
`APP RENAMEF[ILE]` *`app fn1 fn2 ft`*	Renames the file within the application *app* from *filename=fn1 filetype=ft* to a file with *filename=fn2 filetype=ft*.
`APP DELETEF[ILE]` *`app1 fn1 ft1`*	Deletes the file within the application *app1* with *filename=fn1* and *filetype=ft1*.

APP Query

APP LIST	Lists all applications.
APP LIST HOLD	Writes a list of all applications to a file focappl.ftm in a temporary directory; the output is described by focappq.mas.
APP QUERY *app*	Lists all files in the application *app*.
APP QUERY *app* HOLD	Writes a list of all files in the application *app* in the file focappq.ftm in a temporary directory; the output is described by focappq.mas.

APP File Allocation

FI[LEDEF]*ddname* DISK *app/file.dat*	Allocates a data source *file* in application *app*. This command is platform independent.
USE *app/datasource*.FOC AS *name*	Same for FOCUS data sources.

APP Help

APP HELP	Displays a list of APP commands with a brief description of each command.

Accessing Metadata and Procedure Files

Permanent files are files that either were created before the session by another application or remain after the session is over for use by another application.

Search Rules

Unless a file name is fully qualified with the application name, the search rules are:

1. The directory set using APP HOLDMETA command.
2. The list of directories set in APP PATH (contains MVSAPP for z/OS).
3. BASEAPP directory.
4. EDAHOME/catalog.
5. For stored procedures only: if the file is not found, the server checks to see if the file was allocated with a FILEDEF or DYNAM command, and if so, tries to execute it.

Example: Search Paths

For example, the following command

```
EX ABC
```

follows the search path starting with the directory set using APP HOLDMETA command.

The following command

```
EX APP1/ABC
```

first executes profile.fex in the directory pointed to by APP1, then searches for and executes the FOCEXEC ABC.

Syntax: How to Locate a Non-Data File

To locate a non-data file, issue the following command

```
WHENCE filename filetype
```

where:

filename

Is the name of the file you are trying to locate.

filetype

Is the type of file you are trying to locate.

Creation Rules for Procedure Files

Unless a file name is fully qualified, a file is created in the directory application specified with the APP HOLDMETA command.

For z/OS, if DYNAM allocation for HOLDMAST or HOLDACC is present, the metadata files are created in the corresponding PDSs (for example, for a CREATE SYNONYM or TABLE FILE file with HOLD).

Accessing Existing Data Files

There are several methods to allocate existing data files:

- DATASET keyword in the Master File.
- FILEDEF command for non-FOCUS data sources - FIX, VSAM, XML.
- USE command for FOCUS data sources.
- For z/OS server, native operating system services, when supported.
- DYNAM command.
- Superseded by JCL DD card.

It is recommended that you use only one method for each allocation.

Creation Rules for Data Files

For a newly created data file, the location is determined as follows:

1. An application directory set by APP HOLDDATA applies to all HOLD files.
2. For FILEDEF command, one for each data file.
3. For z/OS, native operating system allocations when supported.

The request that caused the file to be created determines the file's DCB parameters, such as record length, record format, and so on.

Sample Allocations by JCL

The following table contains sample allocations by JCL.

VSAM	//VSAM01 DD DISP=SHR, DSN=*qualif*.DATA.VSAM **Note:** This type of allocation requires szero = y parameter on edaserve.cfg file to support sharing of BufferPool Zero.
Fixed	//FIX01 DD DISP=SHR,DSN=*qualif*.FIXED.DATA
PDS	//MASTER DD DISP=SHR,DSN=*qualif*.MASTER.DATA
FOCUS	//CAR DD DISP=SHR,DSN=*qualif*.CAR.FOCUS

Samples of DYNAM Command

The following table contains samples of DYNAM command.

VSAM	DYNAM ALLOC FILE QAVSM DA *qualif*.QAVSM.VSAM SHR REUSE
Fixed	DYNAM ALLOC FILE FILE1 DA *qualif*.FILE1.DATA SHR REUSE
PDS	DYNAM ALLOC FILE MASTER DA *qualif*.MASTER.DATA SHR REUSE
FOCUS	DYNAM ALLOC FILE CAR DA *qualif*.CAR.FOCUS SHR REU

Sample FILEDEF Command

The following is an example of a FILEDEF command

```
FI filedes DISK app/physfile.ftm
```

where:

filedes

Is a file designation.

physfile.ftm

Is a physical file located in the directory pointed to by app.

Samples of USE Command

The USE command supports renaming of Master Files and concatenation of data sets. The USE command is the only mechanism for accessing files on the sink machine.

Renaming a Master File

```
USE
 CAR1 ON CAR
END
```

Concatenating Master Files

```
USE
 CAR1 as CAR
 CAR2 as CAR
END
```

Accessing Files on a Sink Machine

```
USE
 CAR1 AS FOCSU01
END
```

Data Set Names

If a data set name satisfies one of the following conditions, the server assumes that it is an MVS file name:

- Data set name starts with “//”.
- Data set name contains no “/” and contains at least one “.”

In all other cases, the name is interpreted as an HFS file name.

The following syntax is supported:

```
DATASET=APP1/physfile.ftm
DATASET='qualif.car.data'
DATASET=qualif.car.data
```

In addition, on z/OS, you can use the following:

GDG files	FILENAME=CARGDG,SUFFIX=FOCUS,DATASET='qualif.CARGDG.FOCUS(0)'
PDS members	FILENAME=CARMEMB,SUFFIX=FOCUS,DATASET=qualif.CARPDS.DATA(CARMEMB)
FOCUS, VSAM, Fixed	FILENAME=CAR,SUFFIX=FOCUS,DATASET=//'qualif.CAR.FOCUS'

Allocating Temporary Files

How to:

Allocate Temporary Files

Example:

Allocating Temporary Files to MVS Data Sets

Reference:

Usage Notes for Allocating Temporary Files

Temporary files are transient files that disappear after you end a session. By default, all temporary data files (for HOLD and FOCSORT files) and temporary metadata files, such as temporary Master Files and Access Files, are created in an edatemp directory that corresponds to your TSCOM ID. For example, if your TSCOM ID is TS000001, your temporary files are located in //edatemp/ts000001.

For z/OS, you can control the size and location of these temporary metadata files and data files. You can specify that the temporary files reside in MVS data sets, in hiperspace, or in the hierarchical file system.

Syntax: How to Allocate Temporary Files

To specify the allocation of your temporary files, issue the following command

```
DYNAM SET TEMP[ALLOC] {HFS|MVS|HIPER}
```

where:

`HFS`

Allocates temporary files to the hierarchical file system. This is the default value.

`MVS`

Allocates temporary files to MVS data sets.

`HIPER`

Allocates temporary files to hiperspace.

Reference: Usage Notes for Allocating Temporary Files

For z/OS, temporary metadata files can be allocated using a similar procedure to allocating permanent metadata files:

- If DYNAM allocation for HOLDMAST or HOLDACC is present, temporary files are stored in the designated PDSs.
- If DYNAM SET TEMP[ALLOC] MVS is issued; in the default temporary PDSs.
- If DYNAM SET TEMP[ALLOC] HIPER is issued; in the HIPERSPACE.

Example: Allocating Temporary Files to MVS Data Sets

To alter the default allocation parameters for temporary files for MVS data sets, issue the following command

```
DYNAM SET TEMP[ALLOC] FOR <type> <dynam_parms>
```

where:

type

Is one of the following: HOLDACC ,HOLDMAST, HOLD SAVE, REBUILD, FOCUS, FOCSORT, OFFLINE, or FOC$HOLD

dynam parms

Are regular DYNAM ALLOC parameters to be used as default for that type. Note that DCB parameters, if provided here, will be ignored, since they must be compatible with file type being written.

This is similar to the functionality of IBITABLA in the SSCTL Server. The defaults should be overwritten for all cases when in older versions a private copy of IBITABLA existed containing different values.

Reference: System Defaults for Allocating Temporary Files to MVS Data Sets

System defaults for HOLDMAST and HOLDACC are:

```
TRKS 5  5 DSORG PO DIR 36 NEW REU
```

System defaults for all other types are:

```
CYLS 5 10 DSORG PS NEW REU
```

Temporary Space Allocation for UNIX and z/OS UNIX System Services

When you perform a standard server installation, a file system is created and mounted. The amount of space that is available to a running server using the file system is approximately n - 200 megabytes (where n is the size of available space in the file system before the server is installed). It is important to understand the server's utilization of this space on behalf of the user community and the various techniques that can be employed to control the space used.

Temporary File Directory Structure

Temporary space, by default, is created under the configuration directory (ffs, wfs or dm) in the file system used to install the product. The edatemp directory is two levels deep.

The first level directory is used for trace files if tracing is active. By default the directory is named /edatemp.

The subdirectory is used for users temporary file. Each user will have his own directory so that each user's work is isolated from all others. The subdirectory is named /edatemp/ts*nnnnnn* where *nnnnnn* is the running data agent's id number.

The edatemp can be a separate file system to which the variable edatemp points. The edatemp variable can be exported before the server is started or set using the Web Console workspace Edit Files option for editing the edaserve.cfg configuration file.

Specifying the edatemp Variable

For UNIX, the edatemp variable can be coded in the edastart.sh file:

```
export EDATEMP=/u/iway/edatemp
```

For UNIX System Services, it can be coded in the ISTART JCL member under the EDAENV dd statement:

```
//EDAENV      DD   *
EDATEMP=/u/iway/edatemp
```

For z/OS UNIX System Services only, you can set the temporary area to MVS by using the following command in edasprof.prf or user profile.

```
DYNAM SET TEMPALLOC MVS
```

The above are global settings which affect temporary file allocations for all users.

Pre-allocation of Temporary Files

An individual file for a user can be pre-allocated to a separate directory as follows:

- UNIX

  ```
  filedef xxx disk /u/another/area/xxx.dat
  ```

  where /u/another/area has enough free space to hold the file

- z/OS UNIX System Services

  You can use the above same filedef as for UNIX, or you can use the DYNAM command to direct the temporary to MVS.

  ```
  DYNAM ALLOC FILE xxx SPACE ........
  ```

  Note that for the DYNAM command, you have to specify the amount of space required.

- FOCUS files

  The FILEDEF and USE commands can be used to create the FOCUS file outside the edatemp area.

  ```
  FILEDEF filename DISK /pathname ……
  USE
  Filename NEW
  END
  ```

Dynamically Allocating FOCUS Files on z/OS

You can dynamically allocate FOCUS files on z/OS with the USE command. The command is

```
DYNAM ALLOC FILE ddname SPACE
USE ddname AS masterfile
END
```

where:

ddname

Is the DDNAME.

masterfile

Is the Master File name.

If the DDNAME and Master File name are the same, use just the command:

```
DYNAM ALLOC
```

CHAPTER 6

Managing Metadata

Clicking *Metadata* in the navigation pane opens the Managing Metadata page and allows you to perform tasks such as create a synonym. Metadata management is fully described in the *iWay Adapter Administration for UNIX, Windows, OpenVMS, OS/400, OS/390, and z/OS* manual.

CHAPTER 7

Editing and Running Procedures

Topic:

- Using the Procedures Page

Procedures are reusable logic components written in 4GL language. Using procedures enables application logic to be written once and executed many times. The Procedures page enables the creation and testing of stored procedures. It also enables the submission of stored procedures for execution at a later time, in deferred mode.

Using the Procedures Page

The Procedures page enables you to:

- Configure application path and manage application files.
- Set optional parameters for running procedures.

Parameter	Description
Procedure Parameter	Passes parameters to a procedure. They can be positional or keyword.
Connect	Required for the first query on the remote server. If Disconnect After Execution is selected, then it is required for every execution.
Disconnect after execute	Terminates connection to remote server after query execution.
Server	Node name with communication definitions to the remote server.
Service name	Service name to connect to on the server.
User ID	User ID on the remote server.
Password	Password for the user on the server.

- Create, edit, and run procedures.

Function	Description
Create New Procedure	Click a directory and type the name of the new procedure without the extension.
Edit Procedure	Click the procedure name and select *Edit* to open in an edit mode.
Run	Runs the procedure.
Run Deferred	Submits for deferred execution.
Run Stress	Enables a stress test of the procedure by generating an HTI script.

The Application Adapter Server Procedures page enables you to test synonyms.

CHAPTER 8

Configuring Resource Analyzer and Resource Governor

Topics:

- Resource Analyzer and Resource Governor Add-on Tools
- Granting Access to Resource Analyzer/Resource Governor Database Tables
- Running the Resource Governor Configuration Verification Program
- Using Resource Analyzer/Resource Governor Trace Files

The following section describes how to configure and use the Resource Analyzer and Resource Governor.

Important: For all platforms except OS/390, you perform configuration from the Web Console. For OS/390, you perform configuration from the Administration tool.

Resource Analyzer and Resource Governor Add-on Tools

How to:

Add Resource Analyzer/Resource Governor to an Existing Server

Disable Resource Analyzer/Resource Governor

Create Resource Analyzer/Resource Governor Internal Tables

Reconfigure Resource Analyzer/Resource Governor

Resource Analyzer and Resource Governor are add-on tools for servers that help IS organizations analyze and control end-user data access.

Resource Analyzer monitors data usage activity by collecting attributes of requests and storing them in a set of Usage Monitoring Databases. It automatically keeps track of the request, which data sources and columns are accessed, when the request was run, how long it took, which resources it consumed, and so on. Resource Analyzer enables you to report on, graph, and analyze end-user request traffic.

Resource Governor puts these statistics to use, enabling you to stop any request that Resource Governor estimates will exceed predetermined usage thresholds.

For more information on using these products, see the *Resource Analyzer Administrator's and User's Manual* and the *Resource Governor Administrator's and User's Manual*.

To configure the server to use Resource Analyzer and Resource Governor:

1. Add Resource Analyzer/Resource Governor to an existing server.

   Note: If you installed Resource Analyzer/Resource Governor during the initial installation of your server, then this step is not necessary. Proceed to the next step.

2. Create the Resource Analyzer internal tables.

3. If you are storing your Resource Analyzer/Resource Governor internal tables in a relational database management system, you may want to grant access to these tables to different users.

4. Run the configuration verification program (CVP).

   Note: If you are only planning on running Resource Analyzer on your server, this step is not necessary. You are ready to begin using Resource Analyzer.

Procedure: How to Add Resource Analyzer/Resource Governor to an Existing Server

If Resource Analyzer/Resource Governor was not automatically configured during the server installation, you may add it at any time by following these steps:

1. From the Web Console, select *Workspace* in the left pane, and then *License Management*.
2. Type the appropriate license code for the server that contains Resource Analyzer/Resource Governor and click *Save and Restart*.

   Note: If you do not have a license code, contact Customer Support or your local sales office for information about obtaining a new license code.

Syntax: How to Disable Resource Analyzer/Resource Governor

If, at any time, you wish to disable Resource Analyzer/Resource Governor, you can edit the server profile, EDASPROF. PRF:

1. To open the Edit Configuration Related Files pane, select the *Workspace* option in the Web Console, then *Edit Files*.
2. Click the *Edit* button next to EDASPROF.PRF.

   The file opens as text.
3. Turn SMARTMODE off by commenting out the statement SMARTMODE=ON:

   ```
   -*SET SMARTMODE=ON
   ```
4. Click *Save*.

Procedure: How to Create Resource Analyzer/Resource Governor Internal Tables

1. Do one of the following:
   - From the Web Console, select *Workspace*, and then *Resource Analyzer/Governor*.

   or

   - Go to the Resource Analyzer/Resource Governor Administrator and connect to the appropriate server.
2. In the available fields, type the information that is required to create the Resource Analyzer and Resource Governor internal tables.

   The internal tables are created based on your entries. For more information on any of these fields, click the *?* icon. A message appears when this process is complete.

3. If you:

    - Are storing your internal tables in a relational database management system, see *Granting Access to Resource Analyzer/Resource Governor Database Tables* on page 8-5.
    - Selected Resource Governor during installation, see *Running the Resource Governor Configuration Verification Program* on page 8-6 before you use the Resource Governor Administrator to control monitoring and build rules.
    - Are configuring for Resource Analyzer only, you need not run the CVP after the internal tables are created. You may begin to use the Resource Analyzer Administrator to control the monitoring of data sources.

4. Click *Configure*.

Procedure: How to Reconfigure Resource Analyzer/Resource Governor

1. Follow the steps in *How to Create Resource Analyzer/Resource Governor Internal Tables* on page 8-3.

    Make the desired changes.

2. When you are done, click *Re-Configure*.

Granting Access to Resource Analyzer/Resource Governor Database Tables

Resource Analyzer/Resource Governor has two separate databases; one for administration and the other for usage-monitoring data. If you are using a relational database (that is, not FOCUS) as your Resource Analyzer/Resource Governor repository, a GRANT command is issued for all users (PUBLIC) when the internal tables are created. This command enables all users to write to the usage-monitoring database.

The owner ID specified when the internal tables are created is granted user access rights to the administrative database tables (SMCONTROL, SMKBASE, SMPRL, and SMPARAMETERS).

To grant other users access to the Resource Analyzer/Resource Governor administrative database tables, you must issue additional GRANT commands for those user IDs. A sample data file, gkegrant.dat, is located in the etc directory in EDAHOME. This file contains the commands required to grant users access to the Resource Analyzer and Resource Governor administrative database tables.

Syntax: How to Grant Access to the Administrative Database Tables

The following syntax grants users access to the Resource Analyzer/Resource Governor administrative database tables

```
GRANT SELECT, UPDATE, INSERT ON owner.admin_database TO user1, user2 ..
```

where:

owner

Is the owner name of tables that are used for system administration and collection.

admin_database

Is one of the Resource Analyzer/Resource Governor administrative database tables (SMCONTROL, SMKBASE, SMPRL, and SMPARAMETERS).

user1, user2 ..

Are the IDs of the users to whom you wish to grant access.

Note: To grant certain users access to the Resource Analyzer and Resource Governor administrative database tables, you must issue the GRANT command for each database table.

Running the Resource Governor Configuration Verification Program

Note: The CVP is applicable only to Resource Governor, not to Resource Analyzer.

The Configuration Verification Program (CVP) is a program that verifies the Resource Governor installation and configuration. The following remote procedures, located in the catalog directory of EDAHOME, are executed:

- gkecol
- gkeparm
- gkegov
- gkerule
- gkedel

The CVP verifies the following:

- Request usage monitoring (collection).
- Request governing based on collection.

The CVP uses a temporary table, GKEIVP, for validating usage monitoring and governing.

Procedure: How to Run the Resource Governor Configuration Verification Program

1. Run the IVP procedure, gkeivp.fex, located in the catalog subdirectory of the EDAHOME directory.
2. In the Web Console, add the EDAHOME directory to the server path, and then click the procedure in the Procedures page and select *Run*.
3. Examine the output.

   If the Resource Governor installation and configuration is successful, the following messages appear among the output:

   - This statement indicates that the test data was put into the test database:

     ```
     ***********************************
     * INSERTS COMPLETED FOR TEST DATA *
     ***********************************
     ```

   - This statement indicates that the Resource Governor utility procedure, gkecol, completed successfully, and that data about test requests is temporarily logged in the Usage Monitor:

     ```
     *********************************
     * GKECOL COMPLETED SUCCESSFULLY *
     *********************************
     ```

- This statement indicates that the test requests using SELECT against the GKEIVP test database completed successfully, and that Usage Monitor data was populated in the Resource Governor Usage Monitor databases:

  ```
  ***********************************************
  * REQUESTS COMPLETED FOR TEST COLLECTION DATA *
  ***********************************************
  ```

- This is a cancellation message that Resource Governor generates to indicate that a SELECT statement was issued and canceled (based on rules built by the CVP and used for governing):

  ```
  (GKE36048) RESOURCE GOVERNOR CANCELED THIS REQUEST. Governing Mode
  = GOVERN
  KBName = IVP RuleNum = 3 Threshold Type = ROWS Thresh Exceeded = 10
  ```

If the Resource Governor installation and configuration is unsuccessful, error messages (preceded by the keyword ERROR) appear along with the output. These messages are specific to the type of error encountered. For example:

- The following error message results from the incorrect execution of the gkecol procedure:

  ```
  ERROR SETTING COLLECTION ON
  ```

- The following error message results from the incorrect execution of the GKETABLE procedure:

  ```
  ERROR CREATING GKETABLE
  ```

Note: If the success messages do not appear in the TS3OUT listing, or if error messages appear, turn on tracing and examine the messages in the trace for any problems in setup.

Using Resource Analyzer/Resource Governor Trace Files

The internal trace component of Resource Analyzer/Resource Governor is called GKE.

For more information on trace files, see *Enabling and Viewing Trace Files* on page 12-4.

CHAPTER 9

DataMigrator Server Configuration

Topics:

- DataMigrator Server
- Migrating ETL Requests
- DataMigrator Web Console
- Working With DataMigrator Flows
- Testing Sample Data and Process Flows

The following topics describe configuring your DataMigrator Server. This section applies only to users who have installed a DataMigrator Server.

DataMigrator was formerly known as ETL Manager.

DataMigrator Server

The DataMigrator Server houses and executes the data and process flows from which a data warehouse, data mart, or other data targets are constructed. The server consists of the following components:

- Engines for data sorting, transformations, e-mail notification, scheduling, and security rules.
- Read and write adapters, which are used to extract and load the data.
- Metadata or synonyms (.mas and .acx files) that describe column information and access rules to data sources and/or data targets.
- Data flows, which are procedures that contain the necessary instructions for extracting, transforming, and loading the data targets. These procedures are built by the Data Management Console and stored on the server in two files:
    - A procedure with the extension .fex, which is ready to run.
    - A file with the extension .etg, which contains detailed information needed for the graphical display.
- Process flows, which contain the necessary instructions for interleaving data flows with other objects. Like data flows, these instructions are built by the Data Management Console and stored on the server as .fex and .etg files.
- DataMigrator internal tables, containing the DM log and run-time statistics.
- iWay remote server (optional). Access to a remote server is required when source data is on a different platform from the DataMigrator Server. Adapters to extract the source data reside on the remote server (subserver). Though optional, one or more subservers are used at many DataMigrator sites.

Migrating ETL Requests

How to:

Migrate Requests on UNIX and Windows

Migrate Requests on OS/390

Migrate Requests on OS/400

You can migrate ETL Requests from version 4.x and 5.1.x of ETL Manager when you upgrade to a newer version of DataMigrator. To migrate requests, you must do the following:

1. Migrate synonyms and Access Files.
2. Prepare the ETL Manager internal database for migration.
3. Migrate the requests.
4. Test the migrated requests.

Procedure: How to Migrate Requests on UNIX and Windows

1. Locate the Synonyms and Access Files used by your requests. The default locations are:

Operating System and ETL Release	Default Location
UNIX, Release 4.3	home/ibi/srv43/cpm/catalog
UNIX, Release 5.1	home/ibi/srv51/etl/catalog
Windows, Release 4.3	c:\ibi\srv43\cpm\catalog
Windows, Release 5.1	c:\ibi\srv51\etl\catalog

2. Copy the Synonyms and Access Files to the baseapp directory of APPROOT. This is the default directory for Release 5.3 flows.

   or

   Copy the Synonyms and Access Files to any directory under APPROOT, then add that directory to APP PATH.

3. Prepare the ETL internal database for migration. If your DM internal tables are stored in:

   - **A relational database** (such as ORACLE, Microsoft SQL Server, or DB2). Configure an adapter for the data source. For more information, see the *iWay Adapter Administration for UNIX, Windows, OpenVMS, OS/400, OS/390 and z/OS* manual.
   - **FOCUS.** No preparation is required if the internal database file (CMSCHED.FOC) and master file (CMSCHED.MAS) are in their original locations.

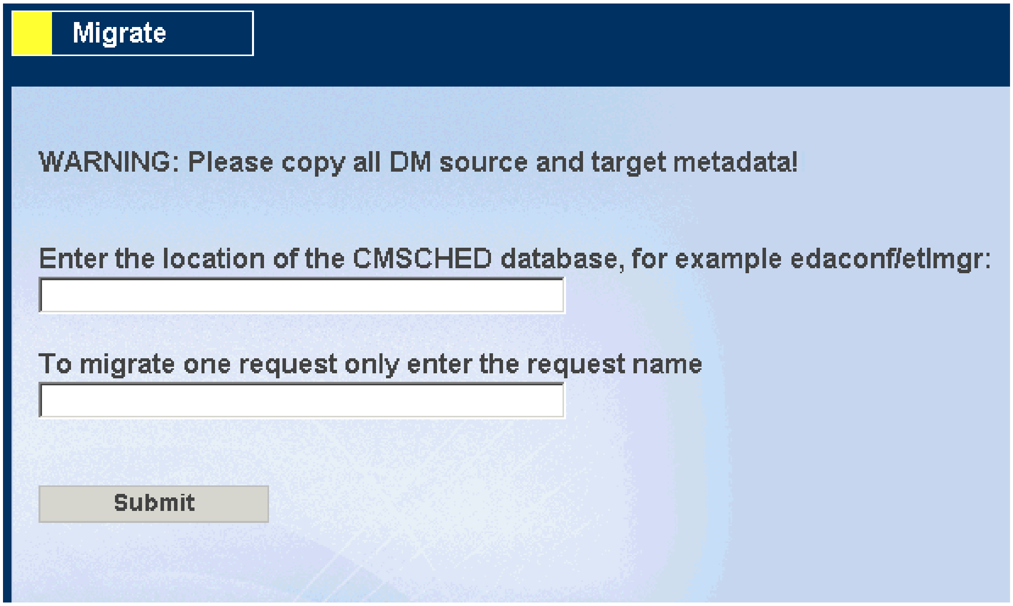
Note that if you are migrating from a disconnected computer and have to copy these files, they should be copied to the *same* directory.

4. From the Data Management Console, right click the server, and select *Access Web Console>Procedures*.

   The Procedures page opens.

5. Open the *DM Utilities* folder, and click *Migrate 4.x or 5.1 Procedures*.

   The Migrate pane opens:



Migrate

WARNING: Please copy all DM source and target metadata!

Enter the location of the CMSCHED database, for example edaconf/etlmgr:

To migrate one request only enter the request name

6. By default, all requests will be migrated.

   To migrate all requests, in the top box of the Migrate pane enter the path to the directory where CMSCHED.MAS and either CMSCHED.ACX or CMSCHED.FOC files reside on the 4.*x* or 5.1.*x* Server.

   The default locations for each platform are as follows:

Operating System and ETL Release	**Default Location**
UNIX, Release 4.3	`home/ibi/srv43/cpm/copymgr`
UNIX, Release 5.1	`home/ibi/srv51/etl/etlmgr`
Windows, Release 4.3	`c:\ibi\srv43\cpm\copymgr`

Operating System and ETL Release	Default Location
Windows, Release 5.1	`c:\ibi\srv51\etl\etlmgr`

To migrate a single request, enter the path in the top box of the Migrate pane, then enter the request name in the bottom box.

7. Click *Submit*.

   The message "Migrate procedure submitted" appears.

8. When the migration procedure finishes, review the log created to ensure that all requests migrated successfully. The following message appears for each flow:

   ```
   (ICM18164) Request NAME was created/updated successfully for user:
   USER.
   ```

   The following message should also appear:

   ```
   (ICM18354) Migration procedure completed successfully.
   ```

   If any requests have not migrated successfully, an error message is displayed. For example, if the synonym for a target table in a request cannot be found for a file named *target01*, the message will read:

   ```
   (FOC205) THE DESCRIPTION CANNOT BE FOUND
   ```

   ```
   Target synonym target01 is not found
   ```

   You must correct the error and rerun the migration procedure without errors before you can open the flow in DataMigrator.

9. Review the list of new application directories. By default, flows are located in the APPROOT\baseapp directory. If a category name was previously assigned in the Properties window, a directory with that category's name is created under APPROOT and the following message appears:

   ```
   *-------   New Created APP directories  -------*/
   newapp
   *--------------------------------------------*/
   ```

   Note: You must add any new application directories to the APP PATH of each user who needs to access it.

10. In the Procedures page's navigation pane, click *Refresh Page*, then verify that the application directories and requests are listed.

Procedure: How to Migrate Requests on OS/390

1. Locate the Synonyms and Access Files used by your requests. These comprise all the data sets concatenated under the ddnames MASTER and ACCESS.

2. Allocate Master and Access PDSes using startup JCL at *qualif*.ETL.DATA(ISTART). For example:

   ```
   //MASTER DD DISP=SHR, DSN=qualif.MASTER.DATA
   //ACCESS DD DISP=SHR, DSN=qualif.ACCESS.DATA
   ```

   All the files from the concatenated datasets under ddnames MASTER and ACCESS appear in the mapped application directory MVSAPP.

3. If the Synonym and Access Files for the CMSCHED tables are in a PDS allocated in step 2, you can skip this step.

   Optionally, copy the Synonym and Access Files for the CMSCHED tables from the PDS (for example, *qualif*.COPYMGR.MASTER.DATA) to an HFS directory (for example, home /ibi/apps/baseapp/).

   In an OS/390 ISPF Command Shell session (option 6 from ISPF menu), enter the following command to copy the CMSCHED Synonyms and Access Files:

   To copy the Synonyms:

   ```
   oput 'qualif.copymgr.master.data(cmsched)'
   'home/ibi/apps/baseapp/cmsched.mas'
   ```

   To copy the Access File:

   ```
   oput 'qualif.copymgr.access.data(cmsched)'
   'home/ibi/apps/baseapp/cmsched.acx'
   ```

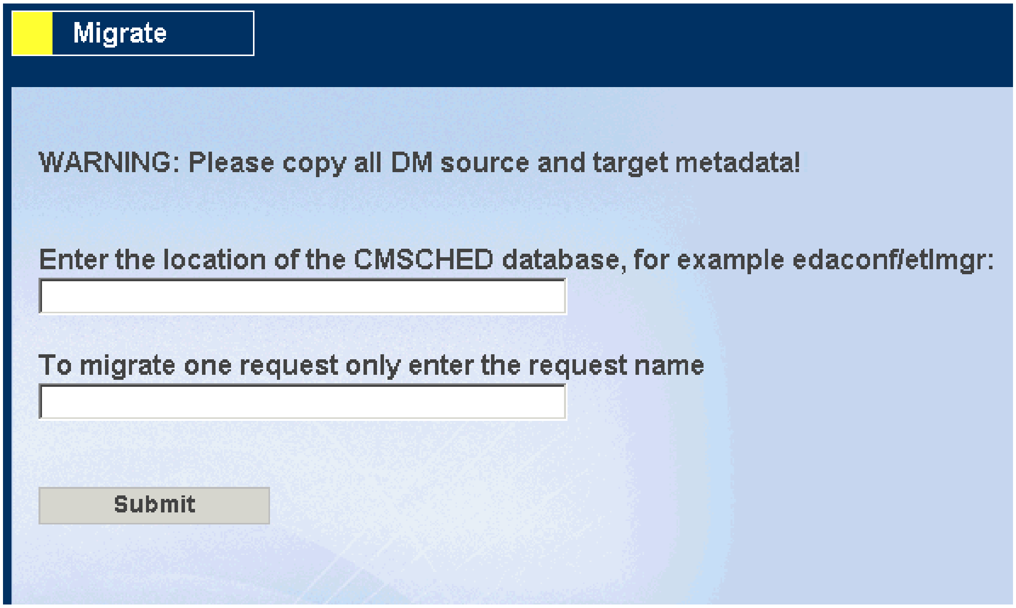
4. Prepare the ETL internal database for migration. To do this, configure the Adapter for DB2 where the internal tables are stored.

5. From the Data Management Console, right click the server, and select *Access Web Console>Procedures*.

   The Procedures page opens.

6. Open the *DM Utilities* folder, and click *Migrate 4.x or 5.1 Procedures*.

The Migrate pane opens.



WARNING: Please copy all DM source and target metadata!

Enter the location of the CMSCHED database, for example edaconf/etlmgr:

To migrate one request only enter the request name

7. By default, all requests will be migrated.

   To migrate all requests, in the top box of the Migrate pane enter the path to the directory where CMSCHED.MAS and either CMSCHED.ACX or CMSCHED.FOC files reside on the 4.*x* or 5.1.*x* Server.

   The default locations for this platform are as follows:

Operating System and ETL Release	Default Location
OS/390, Release 4.3	`home/ibi/apps/baseapp`
OS/390, Release 5.1	`home/ibi/apps/baseapp`

   To migrate a single request, enter the path in the top box, then enter the request name in the bottom box.

   Note: If you did not copy the files in Step 3, and if the PDS was allocated in the start up JCL under MASTER and ACCESS ddnames, the default location is:

   ```
   'mas=//dd:master;acx=//dd:access'
   ```

8. Click *Submit*.

   The message "Migrate procedure submitted" appears.

9. When the migration procedure finishes, review the log created to ensure that all requests migrated successfully. The following message should appear for each flow:

   ```
   (ICM18164) Request NAME was created/updated successfully for user:
   USER.
   ```

   The following message should also appear:

   ```
   (ICM18354) Migration procedure completed successfully.
   ```

   If any requests have not migrated successfully, an error message is displayed. For example, if the synonym for a target table in a request cannot be found for a file named *target01*, the message will read:

   ```
   (FOC205) THE DESCRIPTION CANNOT BE FOUND
   ```

   ```
   Target synonym target01 is not found
   ```

   You must correct the error and rerun the migration procedure without errors before you can open the flow in DataMigrator.

10. Review the list of new application directories. By default, requests are located in the APPROOT\baseapp directory. If a category name was previously assigned in the Properties window, a directory with that category's name is created under APPROOT and the following message appears:

    ```
    *-------    New Created APP directories   -------*/
    newapp
    *-------------------------------------------------*/
    ```

    Note: You must add any new application directories to the APP PATH of each user who needs to access it.

11. In the Procedures page's navigation pane, click *Refresh Page*, then verify that the application directories and requests are listed.

Procedure: How to Migrate Requests on OS/400

1. Locate the Synonyms and Access Files used by your requests. The default location is /QSYS.LIB/EDAMAIN.LIB.

2. For each QSYS library, map the directory the new application will be located in into APPROOT using the Configure Application Path window, and then add the new directory to APP PATH.

3. Copy the Synonyms and Access Files for the CMSCHED tables from the QSYS library (the default location for this library is /QSYS.LIB/EDAMAIN.LIB/MASTER.FILE) to the IFS directory. To do this, in an OS/400 session, enter the following command:

   To copy the Synonyms:

   ```
   CPYTOSTMF FROMMBR('/QSYS.LIB/EDAMAIN.LIB/MASTER.FILE/CMSCHED.MBR')
   
   TOSTMF('home/ibi/apps/baseapp/cmsched.mas')
   ```

   To copy the Access Files:

   ```
   CPYTOSTMF FROMMBR('/QSYS.LIB/EDAMAIN.LIB/ACCESS.FILE/CMSCHED.MBR')
   
   TOSTMF('home/ibi/apps/baseapp/cmsched.acx')
   ```

   With the "replace" option:

   ```
   CPYTOSTMF FROMMBR('/QSYS.LIB/EDAMAIN.LIB/ACCESS.FILE/CMSCHED.MBR')
   
   TOSTMF('home/ibi/apps/baseapp/cmsched.acx') STFMOPT(*REPLACE)
   ```

4. After the Synonyms and Access Files are copied, edit CMCSHED.ACX and change the file name so that it points to the physical location of the table.

   ```
   from TABLENAME=CMCONFIG
   to TABLENAME=EDAMAIN/CMCONFIG
   ```

   where:

   EDAMAIN

   Is the name of the library where the DB2 CMSCHED tables reside.

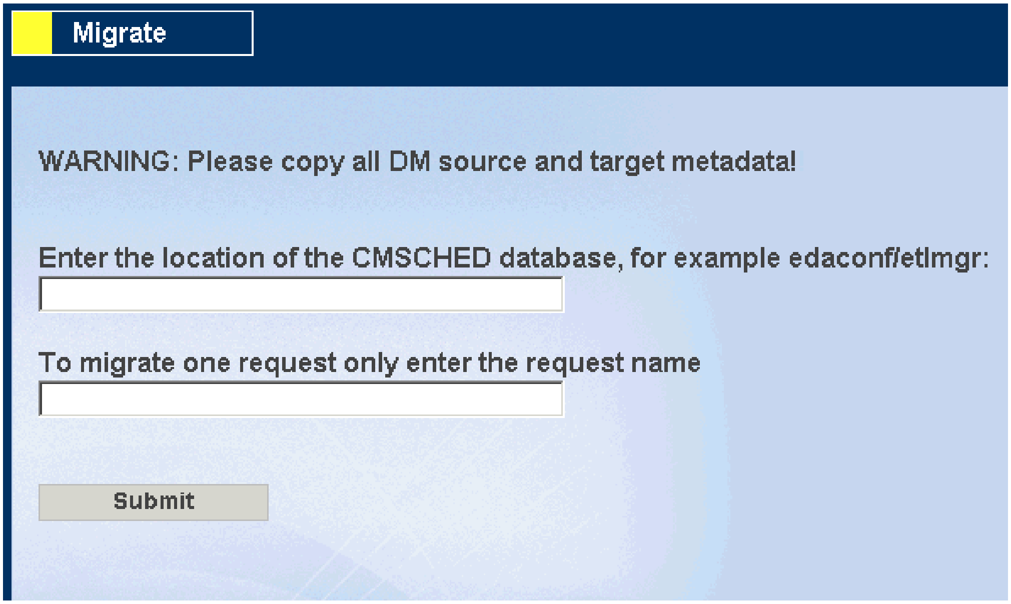
5. Prepare the ETL internal database for migration. To do this, configure the Adapter for DB2 for the internal tables.

6. From the Data Management Console, right click the server, and select *Access Web Console>Procedures*.

   The Procedures page opens.

7. Open the *DM Utilities* folder, and click *Migrate 4.x or 5.1 Procedures*.

The Migrate pane opens.



WARNING: Please copy all DM source and target metadata!

Enter the location of the CMSCHED database, for example edaconf/etlmgr:

To migrate one request only enter the request name

8. By default, all requests will be migrated.

   To migrate all requests, in the top box of the Migrate pane enter the path to the directory where CMSCHED.MAS and either CMSCHED.ACX or CMSCHED.FOC files reside on the 4.*x* or 5.1.*x* Server.

   The default location for this platform is:

   ```
   home/ibi/apps/baseapp
   ```

   To migrate a single request, enter the path in the top box, then enter the request name in the bottom box.

9. Click *Submit*.

   The message "Migrate procedure submitted" appears.

10. When the migration procedure finishes, review the log created to ensure that all requests migrated successfully. The following message should appear for each flow:

    ```
    (ICM18164) Request NAME was created/updated successfully for user:
    USER.
    ```

    The following message should also appear:

    ```
    (ICM18354) Migration procedure completed successfully.
    ```

If any requests have not migrated successfully, an error message is displayed. For example, if the synonym for a target table in a request cannot be found for a file named *target01*, the message will read:

```
(FOC205) THE DESCRIPTION CANNOT BE FOUND
```

```
Target synonym target01 is not found
```

You must correct the error and rerun the migration procedure without errors before you can open the flow in DataMigrator.

11. Review the list of new application directories. By default, requests are located in the APPROOT\baseapp directory. If a category name was previously assigned in the Properties window, a directory with that category's name is created under APPROOT and the following message appears:

```
*-------   New Created APP directories  -------*/
newapp
*---------------------------------------------*/
```

Note: You must add any new application directories to the APP PATH of each user who needs to access it.

12. In the Procedures page's navigation pane, click *Refresh Page*, then verify that the application directories and requests are listed.

DataMigrator Web Console

In this section:

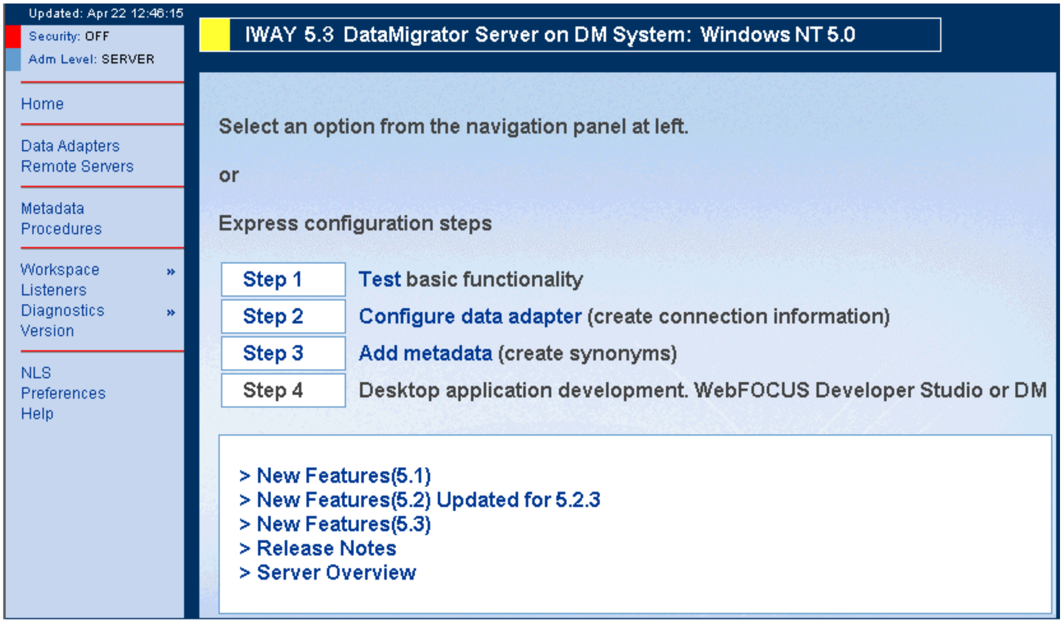
All required server administration and monitoring tasks can be performed by a designated administrator directly from the DataMigrator Server Web Console, also called the Web Console.

The Web Console saves your configuration information in the edaserve.cfg file. You can edit this file directly if you wish; however we recommend that you use the Web Console to change your settings.

Note: DataMigrator requires that you set the application path using APP PATH.

Components of a Web Console Page

Web Console pages consist of two sections, the left pane, used for navigation, and the right pane for specifying information. For example, in the Web Console Home page, the left pane provides navigation links for configuring Data Adapters, managing Metadata etc.



Updated: Apr 22 12:46:15
Security: OFF
Adm Level: SERVER

IWAY 5.3 DataMigrator Server on DM System: Windows NT 5.0

Select an option from the navigation panel at left.

or

Express configuration steps

Step 1	Test basic functionality
Step 2	Configure data adapter (create connection information)
Step 3	Add metadata (create synonyms)
Step 4	Desktop application development. WebFOCUS Developer Studio or DM

> New Features(5.1)
> New Features(5.2) Updated for 5.2.3
> New Features(5.3)
> Release Notes
> Server Overview

When you click a link in the navigation pane, the pane at the right adjusts to display and/or prompt for the appropriate information.

Authorizing DataMigrator Server Administration

DataMigrator Server administration functions require a connection with a server administration user ID, such as the user ID that was used to install and configure the server. This user can add additional server administration user IDs.

Procedure: How to Add Server Administration IDs

1. In the Data Management Console's navigation pane, right-click the server name and choose *Access Web Console* followed by *Home*.
2. In the Web Console, click *Workspace*, then click *Configure*.

3. Click the *Access Control* tab at the top of the page and select one of the following administration levels.

   SERVER

   Permits all functions.

   APPLICATION

   Permits management of flows. This administration level displays flows in the Web Console. It is, therefore, required for DM developers who want to use the Web Console to view their flows.

   OPERATOR

   Permits monitoring of server processes.

Configuring DataMigrator Server Operations

How to:

Edit Scheduler and E-mail Notification Options

Edit the Server Configuration File

Set Up Server Security, Run-time IDs, Logs and Statistics

Reference:

Scheduler Configuration Page

Procedures Page

You can control many key aspects of DataMigrator Server operations from the Web Console, which is directly accessible from the Data Management Console.

The Scheduler Configuration page allows you to:

- Set Scheduler settings.
- Set e-mail notification.

The Procedures page allows you to:

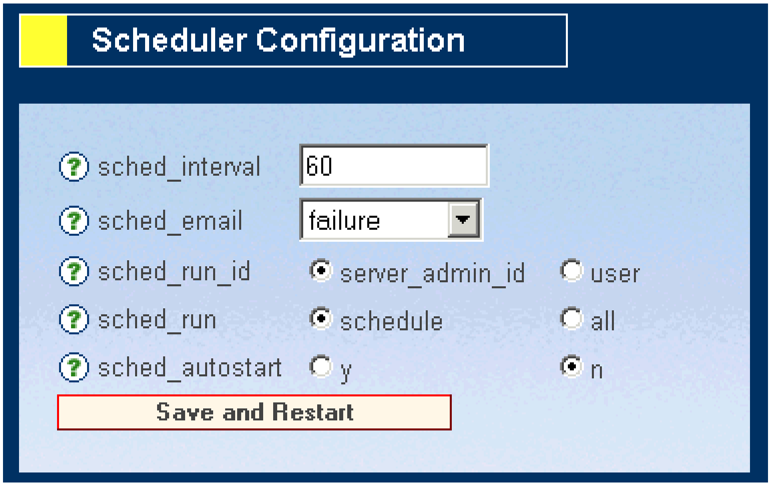
- Configure the application path.
- Stop console requests.
- View and run flows by expanding the approot folder and its application directories.
- Set up DataMigrator Server security and run-time user IDs.
- Start/stop the Scheduler.
- Create sample procedures and data.

- Migrate procedures created in an earlier release.
- Set up log and statistics options
- Run reports.

Procedure: How to Edit Scheduler and E-mail Notification Options

1. In the Data Management Console's navigation pane, right-click the server name and select *Access Web Console>Scheduler Configuration*.

   The Scheduler Configuration page opens.



Scheduler Configuration

sched_interval	60	
sched_email	failure	
sched_run_id	● server_admin_id	○ user
sched_run	● schedule	○ all
sched_autostart	○ y	● n

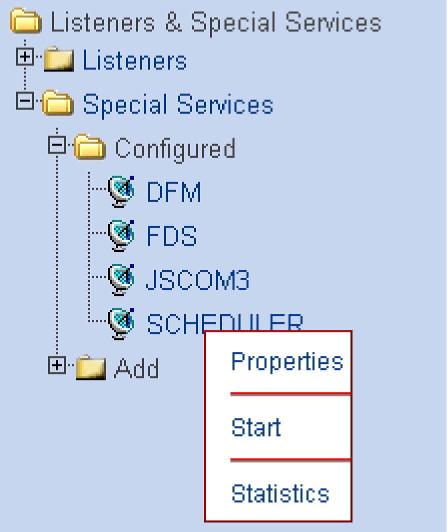
2. Make your changes, and click *Save and Restart*.

You can also access the Scheduler Configuration page from the Web Console Home page.

1. In the Data Management Console's navigation pane, right-click the server name and select *Access Web Console>Home*.
2. Click the *Listeners* link in the navigation pane of the Web Console Home page.

   The Listeners & Special Services page opens.

3. Click the *Scheduler* link in the Configured folder under Special Services.

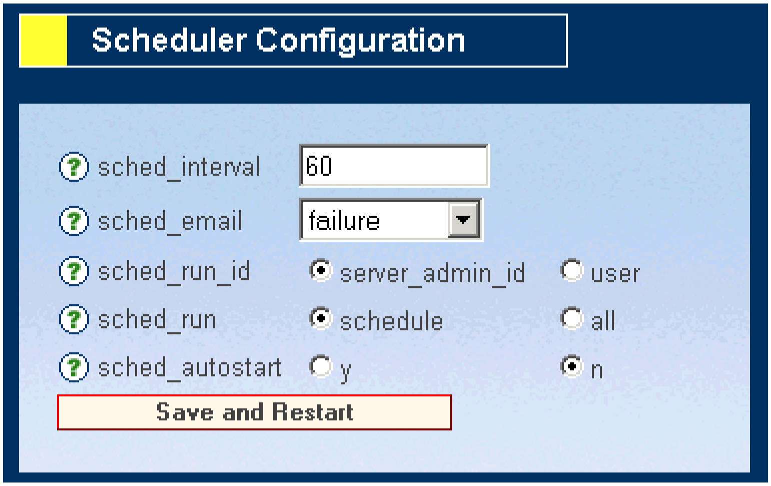


4. Click the *Properties* link in the popup.

   The Scheduler Configuration page opens.

5. Make your changes, and click *Save and Restart*.

Reference: Scheduler Configuration Page



Scheduler Configuration

sched_interval: 60

sched_email: failure

sched_run_id: ● server_admin_id ○ user

sched_run: ● schedule ○ all

sched_autostart: ○ y ● n

The Scheduler Configuration pane contains the following fields/options:

Save and Restart

Saves your changes and restarts the DataMigrator Server.

sched_interval

Is how often, in seconds, you want the Scheduler to check for execution of flows. The default value is 60 seconds.

sched_email

Indicates when to send e-mail for all flows that run. The options are:

off does not send e-mail for every flow, however, you can send e-mail per flow from the Data Management Console. This value is the default.

Tip: If you wish to send e-mail notification on a per-flow basis, select the *Notify* check box in the Properties panel of your Data Management Console.

completion sends e-mail when every flow has been completed.

failure sends e-mail when every flow fails.

Note: Before you can use e-mail notification, you must configure the server to support it.

sched_run_id

Determines the user ID that is associated with a *scheduled* flow when it is run. sched_run_id affects the application directories available to the flow, as well as access to relational data sources and source servers in the profile used by the administrator.

Note: Flows submitted from the Data Management Console, the Web Console, or

CMRUN are run under the userid that submitted the flow.

Valid values are:

server_admin_id

Runs flows under the first ID that appears in the list of server administrators displayed on the Access Control tab of the Workspace Configure page. If the ID does not have a password specified in the Access Control tab, a profile for that user ID must be created. server_admin_id is the default.

This setting enables access to all data sources and source servers in the path specified by that server administrator's profile.

user

Runs flows under the user ID that was used to save the flow. The Application Path specified in the user's profile is utilized.

If you set sched_run_id to User and receive a communications error or -12 when you run a flow, make sure that a user profile exists, and that a valid user ID and password are in the user profile.

Note: If a user's ID has been set to one of the identified server administration access levels (SERVER, APPLICATION, OPERATOR) on the Access Control tab, adding a user connection (profile) is optional. However, if server administration access level for the user's ID has not been set, or if it has been set without a password, a user connection must be added. This user connection must specify both a user ID and password. This applies to scheduled flows and flows submitted for deferred execution from the Data Management Console, the Web Console, or CMRUN.

You can establish a user connection (profile) from the Web Console by selecting Procedures, expanding the DM Utilities folder, and then selecting Set User Connection. Enter the new values in the input areas labeled User name and Password and click Configure. You can add a password for a profile that already exists by selecting the user name from the Selected User drop-down list.

sched_run

Determines whether to run flows as scheduled, or to run all flows. The options are:

schedule runs flows at their scheduled time. This is the default value.

all runs all flows immediately, regardless of their scheduled execution time. This option enables you to do stress testing on the DataMigrator Server.

sched_autostart

Specifies whether the Scheduler should check for flows automatically, or only when prompted. The options are:

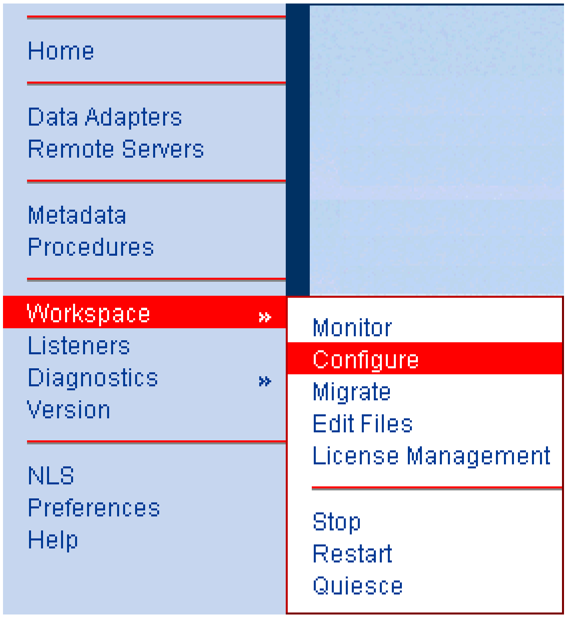
yes automatically starts the Scheduler when the server starts. This is the default value.

no does not start the Scheduler when the server starts.

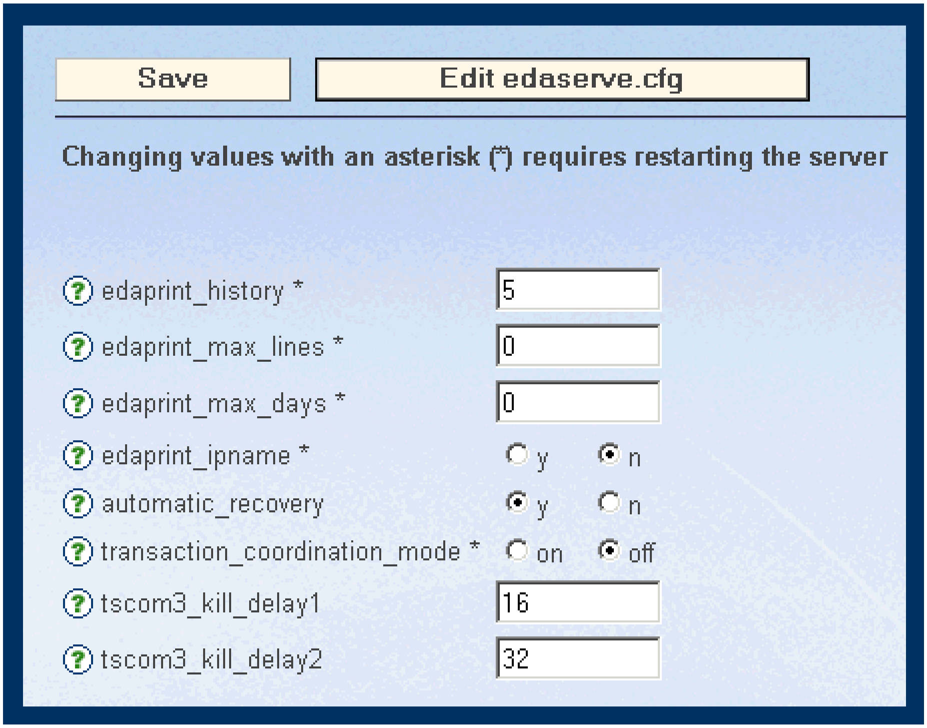
Procedure: How to Edit the Server Configuration File

1. In the Data Management Console's navigation pane, right-click the server name and select *Access Web Console>Home*.

2. Click the *Workspace>Configure* link in the navigation pane.



The Server Configuration page opens.



Changing values with an asterisk (*) requires restarting the server

edaprint_history *	5
edaprint_max_lines *	0
edaprint_max_days *	0
edaprint_ipname *	○ y ● n
automatic_recovery	● y ○ n
transaction_coordination_mode *	○ on ● off
tscom3_kill_delay1	16
tscom3_kill_delay2	32

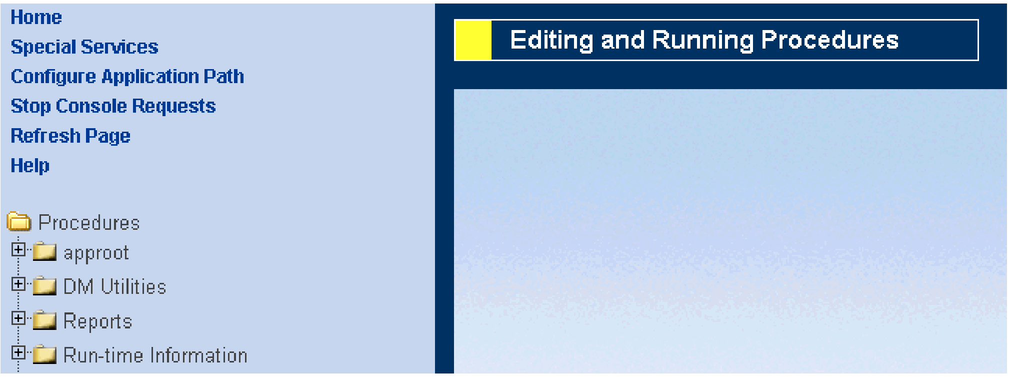
3. Make your changes, and click *Save and Restart*.

For details on the server configuration, see the *iWay Server Administration Guide*.

Procedure: How to Set Up Server Security, Run-time IDs, Logs and Statistics

1. In the Data Management Console’s navigation pane, right-click the server name and select *Access Web Console>Procedures*.

   The Procedures page opens.



2. Expand the folder for the task you want.

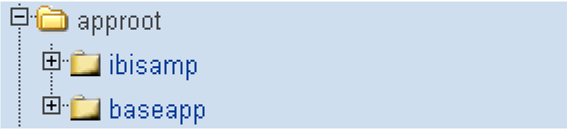
   - **The DM Utilities folder** allows you to set user connections, start the Scheduler, create sample procedures and data, and migrate procedures created with earlier versions. It also provides access to the Manage Log and Statistics folder.
   - **The Manage Log and Statistics folder** allows you to recreate, clean and backup logs and statistics as well as recover deferred statistics.
   - **The Reports folder** provides access to reports on flows, scheduler logs, and events.
   - **The Run-time Information folder** provides access information about current and deferred processes.

   Note: There are also other folders not related to DataMigrator.

Reference: Procedures Page

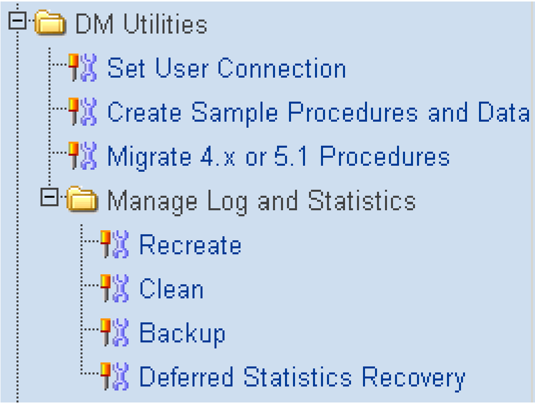
The Procedures page has the following folders:

approot:



The approot folder shows the application directory structure, and provides access to the flows and procedures within those directories.

DM Utilities:



- DM Utilities
  - Set User Connection
  - Create Sample Procedures and Data
  - Migrate 4.x or 5.1 Procedures
  - Manage Log and Statistics
    - Recreate
    - Clean
    - Backup
    - Deferred Statistics Recovery

The **DM Utilities** folder has the following options:

Set User Connection

Displays options for configuring a user profile consisting of a user name and password in the Set User Connections pane. This option applies when a flow is run under the User ID of the flow's creator.

Create Sample Procedures and Data

Creates sample test procedures, and creates the sample data used in the tutorial and other runnable examples in the *iWay DataMigrator User's Guide*.

Migrate 4.x and 5.1 Procedures

Prompts for information to begin the migration of 4.x or 5.1 requests to 5.3 flows.

Note that migration is not required when you upgrade from Release 5.2.x since requests developed in 5.2.x are automatically runnable as Release 5.3 flows.

Manage Log and Statistics folder

Recreate

Creates an empty log and statistics tables. This is the default value.

Clean

Deletes all rows from the log and statistics tables up to a specified date.

Backup

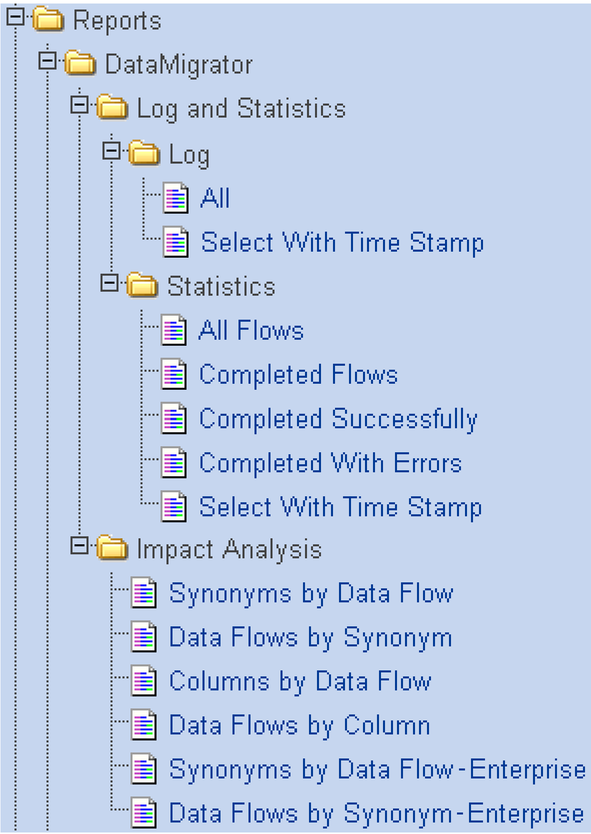
Creates a backup of the log and statistics tables named ETLLOGBK and ETLSTABK, respectively.

Deferred Statistics Recovery

Moves information about deferred flows to the log and statistics tables if the internal tables become unavailable.

Note: If your log and statistics tables are deleted, the Manage Log and Statistics folder is replaced with a Create Log and Statistics option.

Reports/DataMigrator:



The **Reports/DataMigrator** folder has the following options:

The **Log and Statistics folder** contains **Log** and **Statistics** folders.

Log folder

This folder provides access reports showing messages generated from running flows.

All

Generates a report containing all messages.

Select with Time Stamp

Generates a report containing messages from a specific time period.

Statistics folder

These reports contain statistical information about flows, such as how many rows were written, when the flow ran, and how long it took. They also provide access to details about the individual flows.

All Flows

Generates a report containing statistics for all flows.

Completed Flows

Generates a report containing statistics for completed flows.

Completed Successfully

Generates a report containing statistics for flows completed successfully.

Completed with Errors

Generates a report containing statistics for flows completed with errors.

Select with Time Stamp

Generates a report containing statistics from a specific time period.

The **Impact Analysis folder** contains the following reports:

Synonyms by Data Flow

Provides information on the flows and synonyms available on a particular server sorted by data flow.

Data Flows by Synonym

Provides information on the flows and synonyms available on a particular server sorted by synonym.

Columns by Data Flow

Provides information on the columns used in flows and their synonyms sorted by data flow.

Data Flows by Column

Provides information on the columns used in flows and their synonyms sorted by column.

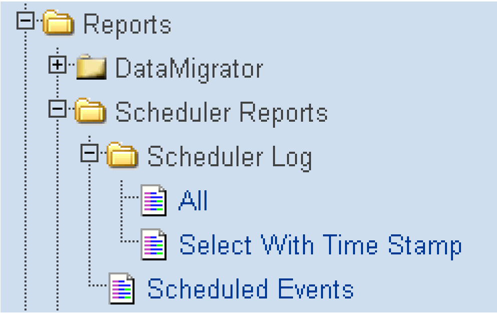
Synonyms by Data Flow - Enterprise

In a hub/subserver environment, provides information on the flows and synonyms available sorted by data flow for the server and all its subservers. This report does not appear if the server you’re reporting on doesn't have any remote server's configured.

Data Flows by Synonym - Enterprise

In a hub/subserver environment, provides information on the flows and synonyms available sorted by synonym for the server and all its subservers. This report does not appear if the server you're reporting on doesn't have any remote server's configured.

Reports/Scheduler Reports:



The **Reports/Scheduler Reports** folder has the following options:

Scheduler Log folder

This folder provides access to reports showing what flows are scheduled to run on a specified day. The default is to display information about the current date.

All

Generates a report containing all messages.

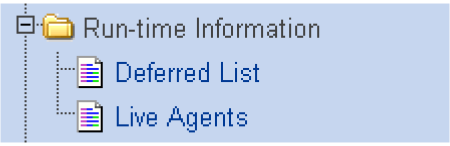
Select with Time Stamp

Generates a report containing messages from a specific time period.

Scheduled Events

Shows what flows are scheduled to run on a specified day. The default is to display information about the current date.

Run-time Information:



The **Run-time Information** folder has the following options:

Deferred List

Displays information about deferred flows.

Live Agents

Displays information about agents that are currently running.

Configuring the Application Path

How to:

Configure the Application Path

Create an Application Directory

Set the Server Application Path

Reference:

Options Pane for Configuring the Application Path

By default, the server profile (EDASPROF.PRF) is run for all users when they connect to the DataMigrator Server to provide access to all application directories in the server's search path.

However, you can control a user's access to application directories by creating individual user profiles. Each user can then:

- Access only the application directories specified in the application path specified for that profile.
- Use synonyms in the specified application path.

You can group flows and metadata into application directories for better control. For your convenience, two directories are included in the server's search path during installation:

- BASEAPP is included in the search path for every user profile the DM administrator creates. This directory cannot be removed.
- IBISAMP contains sample files you can use to create flows, as well as sample flows.

You can, of course, set up additional application directories in the server's search path.

APP PATH is a list of subdirectories beneath a common parent directory called APPROOT. APP PATH is the default search criteria for the server. The server searches each of the directories listed in APP PATH sequentially to find the target procedure.

You can also use the Configure Application Path window to set the viewable directories for DataMigrator, as well as add new directories and mappings to APPROOT. To use DM procedures that reside in a directory other than ibi\apps\baseapp, the directory must be added to the list of viewable directories.

Procedure: How to Configure the Application Path

1. In the Data Management Console's navigation pane, right-click the server name and choose *Access Web Console>Procedures*. The Procedures page opens.
2. Click the *Configure Application Path* link in the navigation pane.

   The corresponding options pane opens.
3. Make your changes, and click *Set APP PATH*.

Procedure: How to Create an Application Directory

1. In the Configure Application Path window, click *New Directory*.
2. Enter the name of the new directory, and click *OK*.

Procedure: How to Set the Server Application Path

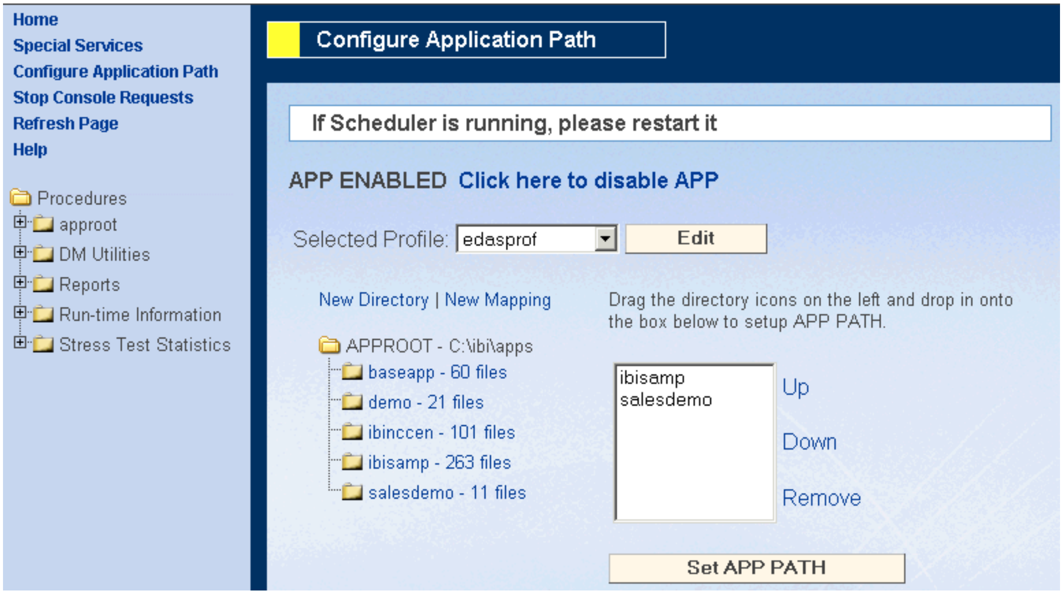
You must add an application directory to the APP PATH for it to be accessible. Follow these steps:

1. Select the profile of the user whose APP PATH you wish to set.
2. In the APPROOT directory tree, drag and drop the directory you want to add from the list of directories to the APP PATH window.

   Note that you must drag the folder icon, not the folder name.
3. Set the directory search order by moving the directories into their search order using the *Up*, *Down*, and *Remove* links.

   You may want to consider moving the ibisamp directory to the bottom of the list to avoid confusion. If you do not plan to use the sample data, you can remove ibisamp from your app path completely.
4. When you have added and arranged all necessary directories, click *Set APP PATH*.

Reference: Options Pane for Configuring the Application Path



If Scheduler is running, please restart it

Drag the directory icons on the left and drop in onto the box below to setup APP PATH.

The Configure Application Path pane contains the following fields/options:

APP ENABLED / APP DISABLED

Specifies whether APP PATH is enabled or disabled.

If you are using DataMigrator, APP PATH must be enabled.

Click here to disable APP / Click here to enable APP

Allows you to enable and disable APP PATH.

Selected Profile

Is the name of the profile file you wish to edit. The edasprof.prf file is selected by default.

You can edit the profile of a specific user, by selecting that user's profile name.

New Directory

Creates a new directory in approot. You can use this option to create a dedicated directory for your application.

New Mapping

Allows you to create a new application name with an alternate mapping to the app default. The newly specified application and its mapping appear in the APROOT directory tree.

APPROOT directory tree

Displays the application directories within the ibi\apps directory.

APP PATH window

Allows you to specify which directories will reside in the APP PATH. Specify a directory by dragging the folder icon from the approot directory tree into the APP PATH window.

Only directories that are in the user's APP PATH are viewable from DataMigrator.

The order of the directories determines what directories are searched first.

Up moves the selected directory up in the list of directories.

Down moves the selected directory down in the list of directories.

Remove removes a directory from approot.

Set APP PATH

Saves the changes made to the application path settings.

Reference: Configuring the Server for E-mail Notification

From the Web Console, you can set up your DataMigrator Server to send an e-mail message that notifies recipients of the successful completion or failure of execution for every flow.

In addition, from the Data Management Console, you can enable the distribution of e-mail messages at any point in a process flow.

However, to enable either variation on the e-mail notifications process, you must first configure an SMTP service. (You need to set up your server for e-mail before it will work anywhere in the flow.)

Procedure: How to Configure SMTP Services

1. In the Data Management Console's navigation pane, right-click the server name and choose *Access Web Console>Home*.
2. From the Web Console Home page, click *Workspace*, then click *Configure*.
3. Click the E-Mail Notification tab at the top of the page and enter the following information:
   - SMTP_HOST defines the IPv4 address of the SMTP e-mail server. The address can be in standard dot notation (for example, *smtpmail.yourcompany.com*).
   - SENDER_EMAIL defines the sender's e-mail address, which appears in the 'From' field in emails (for example, *yourname@yourcompany.com*).
   - SERVER_ADMIN_EMAIL specifies recipients' e-mail addresses (`e-mail address1; e-mail address 2;...`).

- WSM_EMAIL controls distribution of e-mail by Workspace Manager to addresses defined in SERVER_ADMIN_EMAIL when specified events occur (`event1, event2, ...`). For example, you might define stopping and starting the server as events that trigger e-mail distribution.

  The followings predefined events indicate abnormal states resulting from a fatal software error in which the agent process is no longer running. They are provided for diagnostic purposes:

  Crashed indicates an error detected by the operating system.

  Aborted indicates an error detected by the program.

  Once the problem has been investigated, the server administrator can clear such such agents using the kill option.

4. Click *Save*.
5. After configuring SMTP services, go to the Web Console DM Configuration page and select an etl_mail option: off, completion, failure.

Adding a Data Adapter

To access data during the DM process, the data adapters you need must be defined to the server. You need to add a data adapter for every database type you want to access.

Procedure: How to Add and Configure a Data Adapter from the Web Console

1. From the Windows Program menu, choose the *iWay Software>iWay 53 DM Client>Data Management Console*. The Data Management Console opens, with the available servers listed under Servers-Procedures in the navigation pane at the left of the screen.
2. Right-click a server name and select *Connect*.
3. Right-click the server name again. This time select *Access Web Console>Configure Data Adapters*. The Configuring Data Adapters page of the Web Console opens.
4. Expand the Add folder, then expand the appropriate data adapter.
5. Click the Release or Access method for the selected adapter. The Connection Attributes pane opens.
6. Configuring the adapter consists of specifying connection and authentication information for each of the connections you want to establish. Enter the appropriate information and click *Configure*.

For complete documentation on configuring data adapters and for a detailed explanation of the associated connection parameters, refer to the *Data Adapter Administration* manual.

Creating and Testing Metadata

How to:

Create Synonyms in the Web Console

Test Synonyms From the Web Console

Synonyms constitute the metadata that DataMigrator requires to define unique names (or aliases) for each table or view that is accessible from the server. A synonym must be created for every table that DataMigrator accesses.

Once you've created the metadata (synonyms), it's a good idea to test it to make sure that the data is available for DM processing. You can perform this task from the Web Console or directly from the Data Management Console.

Procedure: How to Create Synonyms in the Web Console

1. From the Windows Program menu, choose *iWay Software>iWay 53 DM Client>Data Management Console*. The Data Management Console opens, with the available servers listed under Servers - Procedures in the navigation pane at the left of the screen.
2. Right-click on a server name and select *Connect*.
3. Enter a valid user ID and password. Click *OK*.
4. Right-click on the server name and select *Access Web Console>Manage Metadata*. The Managing Metadata page of the Web Console displays.
5. Expand the Add folder, expand the adapter folder, and then click a connection. The right pane displays table and view selection options:
   - **Select All Tables/Views.** Select this radio button to create synonyms for all tables and views. This value is the default.
   - **Filter by Name, Owner and Table Type.** Select this radio button to filter the tables or views for which to create synonyms. Selecting this option adds the following:

     Owner. Enter a string for filtering the owners' IDs, inserting the wildcard character (%) as needed at the beginning and/or end of the string. For example, enter: ABC% to select tables or views whose owners' IDs begin with the letters ABC; %ABC to select tables or views whose owners' IDs end with the letters ABC; %ABC% to select tables or views whose owners' IDs contain the letters ABC at the beginning, middle, or end.

Table Name. Enter a string for filtering the table or view names, inserting the wildcard character (%) as needed at the beginning and/or end of the string. For example, enter: ABC% to select all tables or views whose names begin with the letters ABC; %ABC to select tables or views whose names end with the letters ABC; %ABC% to select tables or views whose names contain the letters ABC at the beginning, middle, or end.

Table Type. Select one of the following options: Tables (this is the default), Views, Tables/Views.

6. Click the *Select Tables* button. All tables that meet the specified criteria are displayed.

7. If you have tables with identical table names, assign a prefix to distinguish them. For example, if you have identically named human resources and payroll tables, assign the prefix HR to distinguish the synonyms for the human resources tables. Note that the resulting synonym name cannot exceed 64 characters.

   If all tables and views have unique names, leave the field blank.

8. From the SET CONVERSION LONGCHAR drop-down list, select *Alpha*.

9. From the Select Application Directory drop-down list, select a directory. The default value is *baseapp*.

10. Complete your table or view selection:

    - To select all tables or views in the list, click *All*.

    - To select specific tables or views, click the corresponding check boxes.

11. The Default Synonym Name column displays the name that will be assigned to the synonym. To assign a different name, replace the displayed value.

12. Click *Create Synonym*. Synonyms are created and added under the specified application directory.

Procedure: How to Test Synonyms From the Web Console

1. From the Metadata Page, expand the approot directory.

2. Click the data source you wish to test and select *Sample data* from the pop-up menu.

   Sample data is displayed in the right pane.

Working With DataMigrator Flows

You can work with DataMigrator flows using the DataMigrator Procedures directory tree, which contains a list of all the directories available on APP PATH and the DataMigrator flows residing within them. Clicking the name of a DataMigrator flow opens a pop-menu with the following options:

View as Text	Opens the code of the DataMigrator flow.
Run	Runs the DataMigrator flow.
Run Deferred	Adds the DataMigrator flow to the queue of deferred flows and displays a confirmation message in the Web Console.
Copy Delete Move	Copies, deletes or moves the DataMigrator flow. To delete multiple requests, **1.** Click the *page* icon next to each procedure you want to delete. A check marks each of these procedures. **2.** Click the name of one of the marked procedures to open the pop-up menu. **3.** Click *Delete Procedure*. All checked DataMigrator Requests are deleted.
View Log	Displays the flow's log.
View Last Log	Displays information about the last execution of the flow.
View Scheduled Events	Displays information about deferred flows.

Testing Sample Data and Process Flows

Example:

Testing the Sample Data Flow DMVPX

Testing the Sample Data Flow DMVPY

Testing the Sample Process Flow DMVPZ

Once you have verified that all required DM setup tasks have been successfully completed, you can perform three simple tests to ensure that your flows will run properly. We created some sample data and process flow procedures for you to use in testing your setup.

We recommend that you run these tests before you create and attempt to run your own flows.

Example: Testing the Sample Data Flow DMVPX

1. Expand the *ibisamp* directory in the navigation pane.
2. Double-click on the *dmvpx* data flow. The data flow opens in the workspace at the right.

   This data flow extracts the department 'Production' data from the data source named employe2 and creates a new data target, also named dmvpx.
3. Click the *Run* icon on the toolbar and select *Run* from the drop-down menu.
4. When the data flow run has been completed, the following message appears in the Server Message window:

   ```
   (ICM18763) Request ibisamp/dmvpx complete
   ```
5. Right-click *dmvpx* in the Servers - Procedures tree and select *View Last Log*.

   The log, which is displayed in the workspace, indicates that the new table named *dmvpx* has been created and loaded with six rows.
6. Select *Close All* from the Window menu to close the log and the data flow window.

Example: Testing the Sample Data Flow DMVPY

This example assumes that you have already completed *Testing the Sample Data Flow DMVPX*, since it uses a table that you create there.

1. Expand the *ibisamp* directory in the navigation pane.
2. Double-click on the *dmvpy* data flow. The data flow opens in the workspace at the right.

   This data flow extracts the records for the department 'MIS' from the data source named employe2 and adds them to the existing table dmvpx.
3. Click the *Run* icon on the toolbar and select *Run* from the drop-down menu.
4. When the data flow run is complete, the following message appears in the Server Message window:

   ```
   (ICM18763) Request ibisamp/dmvpy complete
   ```
5. Right-click *dmvpy* in the Server - Procedures tree and select *View Last Log*.

   The log in the workspace shows that the target table exists and that six records for the MIS department were added to the target.

Example: Testing the Sample Process Flow DMVPZ

The data flows that ran separately in *Testing the Sample Data Flow DMVPX* and *Testing the Sample Data Flow DMVPY* now run as a combined process flow.

1. Expand the *ibisamp* directory in the navigation pane.
2. Double-click on the *dmvpz* process flow. The process flow opens in the workspace at the right.

   This process flow sets up the flow of *dmvpx* followed by *dmvpy*.
3. Click the *Run* icon on the toolbar and select *Run* from the menu.

   When the process flow runs, the first data flow creates the new table and the second data flow loads it with additional records.
4. When the process flow run is complete, the following message appears in the Server Message window:

   ```
   (ICM18763) Request ibisamp/dmvpz complete
   ```
5. Right-click *dmvpz* in the Servers- Procedures tree and select *View Last Log*.

   The log in the workspace shows that the two data flows ran successfully.

Next Steps: If you have received these results, you can feel confident that your client/server setup is running correctly and you can proceed to the design phase.

CHAPTER 10

Running and Monitoring Your Server

Topics:

- Configuring Workspace Manager
- Monitoring Server Activity
- Migrating Your Server
- Configuring Traces
- Editing Configuration Files
- Viewing and Editing Your License Number

The following section describes how to run and monitor your server.

Configuring Workspace Manager

The Workspace Manager is the component of iWay Server that is responsible for managing all server administrative tasks. iWay Server Administrators use it to monitor server activity, configure and adjust the server's configuration profile, add Users, enable and create Services, define Deferred execution characteristics, and enable and disable e-mail alerts.

The iWay Server Administrator responsible for installation, configuration, security, and maintenance of the server maintains the server available to clients and running at peak efficiency. To use the Workspace Manager, open Internet Explorer and point it to the HTTP port on the HOST machine where the iWay Server is running.

Note: Your iWay Server must be running in order for you to use the Workspace Manager.

Individual keyword references are linked to the corresponding description for each parameter. Refer to one of the keyword links, or click the ? icon next to the parameter on the Workspace Configuration page for a complete list with explanations.

Workspace Manager Configuration

Access to the administrative features of the Web Console can be restricted by a list of users defined in . Users defined in the list may have Server or Application administration level. A Server Administrator has the ability to perform all the administrative tasks available through Web Console operations. If there is more than one Server Administrator defined, the first valid member of the list is used to impersonate FDS and other special services. An Application Administrator is limited to the administrative tasks that do not require changing configuration or restarting the server. All other users (basic users) can only use the Web Console tasks indicated as such in the following table.

Any IDs (beyond the original ID used to configure the server) that are used for server or application administration also require read/write privileges to the respective locations that the IDs are expected to manage. To do this, establish group rights for the locations at the operating system level. To view and run Resource Governor procedures, IDs must be at least at the Application Administrator level.

Web Console Task	**Administration Level**			
	Server Administrator	**Application Administrator**	**Operator**	**Basic user**
Home Page	Yes	Yes	Yes	Yes
Workspace	Yes	In monitor mode only	In monitor mode only	In monitor mode only
Start and stop agents, terminate sessions and connections	Yes	No	Yes	No
Start and stop server	Yes	No	Yes	No
Diagnostics Page	Yes	Display tracing only	Display tracing only	Display tracing only
Version, Log off, Preferences and Help Pages	Yes	Yes	Yes	Yes
Data Adapters Page	Yes	Yes, except Add/Change/Remove connections parameters and Edit configuration files	No	No
Remote Servers Page	Yes	Yes, except Add/Change/Remove connections parameters and Edit configuration files	No	No

Web Console Task	**Administration Level**			
	Server Administrator	**Application Administrator**	**Operator**	**Basic user**
Metadata and Procedures Pages	Yes	Yes, except Configure Application Path	No	No
DataMigrator Page	Yes	Yes, except Configure Application Path	No	No
Create more Server or Application Administrators	Yes	No	No	No

A server configuration requires at least one agent service with the name DEFAULT, defined by a . An agent service is the entity used to define the parameters for a group of data access agents, so that a configuration can manage different groups of data access agents for different purposes. Each data access agent runs for a specific service, and each service may have different values for the settings defined at the service scope. Unless noted otherwise, Workspace Manager settings have global scope.

The settings on the Workspace Configuration page that have service scope are as follows:

The maximum number of data access agents and the number of agents prestarted at server startup for a service are defined by and . The lifetime of a service's agents can be limited through , and/or . Incoming connections for which there is no available data access agent can be put in a queue for the service (configured using and), and after they are connected, their idle time can be limited using .

The of a service defines how data access agents are assigned to connections:

In private deployment, a dedicated application agent is assigned for each connection request. Private deployment retains the behavior of all prior server releases. At connect time, global, as well as user and profiles, are executed. At disconnect time, all temporary files are removed and database connections are closed. The privileges of each application agent depend on the security mode of the server. For more information, see *Server Profiles* in Chapter 1, *Introduction*. With security ON, authentication is processed for every client logging on to the server. With security OFF, user identification and authentication are not required. Requests are processed as the server ID.

In pooled deployment, a predetermined number of application agents can support a large community of users, provided the application is designed to support the small LUW (Logical Unit of Work) concept. These application agents establish their application environments on startup and maintain their environments, including database connections between LUWs, for user connections. This option provides a way to reduce system resource usage and increase for the transaction processing type application. Pooled deployment executes the global server profile and the only. Therefore, each application agent inherits the privileges of one user account. Determining the user account ID that all application agents share depends on what operating system you are using and whether or not you set external security on or off. On the server, each user in the pool shares the user ID of the pooled ID (configured using and).

You can also use the Workspace Configuration page to configure the following settings, which have global scope: , , and .

Deferred Management Configuration

The edaserve.cfg settings corresponding to deferred features are managed separately using the Deferred Management Configuration page. , , and are about scheduling of deferred requests; and are about deferred reports.

Monitoring Server Activity

In this section:

Every server type can be monitored and have its operation parameters changed through the console. These changes can affect the behavior and the performance of the server. The following topics describe the pages used to perform such administrative tasks on a running server.

Clicking *Workspace/Monitor* provides access to the monitoring pages by displaying the first page in the right pane. Use the tabs at the top of the right pane to navigate between monitoring pages.

Server Statistics

The Statistics page displays a list of statistics for the running server as collected by the Workspace Manager.

Note: Click *Refresh Now* to refresh the statistics. The information can be refreshed at predefined intervals by clicking the check box and typing the number of seconds between each refresh.

Click *Reset* to reset Workspace Manager statistics and start collecting from the beginning.

Reference: Server Statistics

Following is a list of the parameters that can appear on the Server statistics page.

Parameter	Explanation
Service	Name of the service configured for Workspace Manager in the edaserve.cfg file.
Conn Total	Count of total connections by all sessions since the server was started. This is different from the total number of sessions, since a persistent session can connect multiple times for different requests.
Conn Completed	Number of successfully completed connections.
Sessions Now	Number of sessions that are currently connected to agents.
Sessions Total	Count of total sessions by all users since the server was started. This is not necessarily the same as the total number of agents started since the server was started, because agents are often reused and can also be pre-started and not used.
Res Fails	Total number of connections rejected for lack of available agents. This includes timed out queued connections.
Sec Fails	Total number of connections that failed for security reasons.
FOCUS Errors	Number of completed connections with FOCUS error(s) during request execution.
Avg Rsp Time	Average number of seconds for request execution.
Queued Now	Number of currently queued connections (not available when queuing is off).

Parameter	Explanation
Queue Timeout	Number of connections that went in the queue and timed out (not available when queuing is off).
Queued Peak	Maximum number of connections that were queued at any given time (not available when queuing is off).
Queue Time	Average number of seconds spent in queue for all connections (not available when queuing is off).
Agents Now	Number of agents that are currently running.
Agents peak	Maximum number of agents that were running at any given time.
Workspace manager process id	Operating system ID number for the current Workspace Manager process.
Started on	Date and time when the current instance of the Workspace Manager was started.
EDATEMP available disk space	Number of kilobytes free on the disk used for the EDATEMP directory, where the edatemp subdirectory and EDAPRINT log are stored.
Average time in queue for all connections	Average number of seconds spent in queue for all connections, only available when queuing is turned on (maximum_q greater than zero).
Number of satisfied queued Connections	Total number of connections that were queued and were eventually successfully connected. Connections that were queued but then failed are not counted here.
Average time in queue for satisfied queued connections	Average number of seconds waiting in queue before connecting for connections that were queued and were eventually successfully connected.
Average running proportion	See the following notes.
Effective server admin	First valid member of server_admin_id (configured in edaserve.cfg). Special services or listeners are run in the security context of that user account when required.
Server admin	List of Server Administrators configured in edaserve.cfg.
Application admin	List of Application Administrators configured in edaseve.cfg.

Note:

1. Some of the preceding variables only appear if the corresponding total is not zero, grouped as follows:
    - total number of connections
    - avg time in queue for above (seconds)
    - number of satisfied queued connections
    - avg time in queue for above (seconds)
    - number of queued connections timed out
    - number of connections run
    - avg agent processing time (seconds)
    - avg running time proportion
2. Variables that are averages (avg) are fractional numbers of seconds rounded to the nearest millisecond for display purposes, but actually computed in higher precision, depending on the operating system:
    - Avg Rsp Time is a measure of waiting time - idle time + running (that is, agent processing) time for all connections and divided by number of connections. It is actually the average time from the moment the user clicks to send his request to connect or resume until the moment the answer appears on the browser, when the user suspends or disconnects the session.
    - Agent processing time is the part of the connection duration spent only on running the request.
    - Running time proportion of a connection is the percentage of its running time compared to its duration.

    The accuracy of the three corresponding averages is only limited by the precision of the operating system. In the rare case where a machine is faster than the precision of its time-measuring, accuracy side effects may occur:

    - For an individual connection that has a duration shorter than the precision of the operating system, duration and running time cannot be measured and are both 0, so the server considers it a 100% running time. A high occurrence of these instances produces an overestimated average proportion.
    - If the duration is longer than the precision because of waiting time but the running time is still less than the precision, then a zero is counted toward the average running time and in calculating the proportion, so a 0% running time proportion is recorded. A high occurrence of these instances produces an underestimated proportion.

Monitoring Agents

Reference:

Agent Statistics

Additional Agent Statistics

The Agents page displays the statistics for the current list of agents monitored by the Workspace Manager. From this page, administrators can manage the existing agent processes, which can be monitored and stopped, and they can also pre-start new agents.

To stop an individual agent, click the agent's row to access the contextual pop-up menu for that agent, and click *Kill Agent*. The running agent is terminated, which invalidates any current connection to that agent. If a request is then issued from such a connection, an error message is returned. Stopping an agent is therefore an emergency administrative measure, as it disrupts the application. After an agent is terminated, the Agents page refreshes automatically. The corresponding row of the terminated agent remains, and the State is listed as 'stopping' until the row eventually disappears. For extreme situations, you may use *Kill All Agents* at the top of the page to terminate all running agents.

To pre-start new agents, type in the number of agents that you wish to start, and click *Start*. Use this method to start additional agents even when there are agents already running.

Note: Click *Refresh Now* to refresh the statistics that appear. The information can be refreshed at predefined intervals by clicking the check box and typing the number of seconds between each refresh.

Reference: Agent Statistics

Following is a list of the parameters that can appear on the Agents page.

Note: You can click an agent row to view additional statistics for that agent (see the following table), or to view the trace file for that agent if it is available.

Statistic	Explanation
Tscom ID	Identification number associated with the agent.
User	User ID associated with the server connection.
State	Current state of the agent. Possible values are starting, stopping, idle, in use, aborted, or crashed. *starting* and *stopping* are normal transitory states, which are self-explanatory. *idle* is when the agent is not connected. It has no sessions at all, whether active or suspended. An agent is *in use* as soon as a session connects and until it disconnects. This includes all time spent between a session suspend and resume when the process is not using the CPU but still has resources allocated for at least one session. *aborted* and *crashed* are abnormal states resulting from a fatal software error detected by the program (aborted) or by the operating system (crashed). The agent process is no longer running and these states are provided for diagnostics purposes. The server administrator can clear (using kill) such agents after the problem is investigated.
Session	Session ID uniquely identifying the currently active session.
Query Time	Indicates the last time that a request was made to the agent. This value is used to calculate the time an agent is idle, in cases where was set.
Last Command	Indicates the first eight characters of the last instruction executed by the data access agent.
FOC/Ext IO	First value is the number of FOCUS I/O operations performed by the agent. The second value is the number of External input operations performed by the agent.
CPU Time	Total CPU time used by the agent.
Memory Usage	Amount of memory in KB used by the agent.

Reference: Additional Agent Statistics

The additional statistics that appear when you click the View statistics option on an agent row are the PID, followed by system-specific statistics for the agent process, then additional portable statistics as described in the following table.

Statistic	Explanation
Number of Sessions	Current number of sessions connected to the agent.
Last Command	Indicates the last instruction executed by the agent against the Server.
Last Master File name	Indicates the last Master File used by the agent.
FOCUS I/O	Indicates the number of FOCUS database I/O operations performed by the agent.
External Database Input	Indicates the number of External Database rows retrieved for a TABLE command, or FIXFORM data captures for a MODIFY command performed by the agent.
Number of Transactions or HLI Commands	Indicates the number of transactions performed by the agent.

Monitoring Sessions

The Sessions page displays the statistics for the current list of sessions assigned to data access agents, enabling administrators to monitor and, if necessary, kill individual sessions.

To terminate a session, click the session's row to access the contextual pop-up menu for that session, and click *Kill Session*. The session is forcefully disconnected from its agent. After a session is terminated, the Sessions page refreshes automatically. The corresponding row of the terminated session remains, and the State is listed as 'terminating' until the row eventually disappears.

Note: Click *Refresh Now* to refresh the statistics. The information can be refreshed at predefined intervals by clicking the check box and typing the number of seconds between each refresh.

Reference: Session Statistics

Following is a list of the parameters that can appear on the Sessions page.

Statistic	Explanation
Tscom ID	Identifies the agent to which the session is connected.
Session	Identification number associated with the session.
State	Current state of the session. Possible values are active, suspended, or terminating. *active* is when the session is connected and active (not suspended). *suspended* is when the session is connected but suspended. *terminating* is a transitory state when the session is killed.
User	User ID that connected the session.
Last Query	Indicates the last time that a request was made by the session. This value is used to calculate the time a session is idle, in cases where was set.
Last Command	Indicates the first 8 characters of the last instruction executed by the session within its agent.
Code Page	Indicates the character code page used by the client connected to the session.

Statistic	Explanation
Connected From	For some protocols, indicates the localization of the client who connected the session by displaying its network address. For example, for TCP/IP, the client's IP address appears.
CPU Time	Total CPU time, in seconds, used within the session.

Monitoring Connections

The Connections page displays the statistics for the current list of connections, enabling administrators to monitor and, if necessary, cancel individual connections.

A connection refers to a physical connection between Client and Server. There are two types of connections: active and queued. An active connection is one that is assigned to a session in a data access agent. A queued connection is one for which there are no available agents for the requested service, and the service is configured with a queue. A queued connection waiting for an agent becomes active as soon as an agent is available. If the maximum time to wait in the queue is reached, the connection is automatically cancelled by the Workspace Manager.

To cancel a connection, click the connection's row to access the contextual pop-up menu for that connection, and click *Kill Connection*. For an active connection, its session is forcefully disconnected from its agent. For a queued connection, it is simply cancelled and the client gets the same error that it would get if queueing was off and there were no available agents, or if the queue was full.

Note: Click *Refresh Now* to refresh the statistics that appear. The information can be refreshed at predefined intervals by clicking the check box and typing the number of seconds between each refresh.

Reference: Connection Statistics

Following is a list of the parameters that can appear on the Connections page.

Statistic	Explanation
Type	Identifies the connection type: connect or resume.
Status	Current status of the connection, either active or queued: *Active* indicates that there is a session associated with the connection. *Queued* indicates that there is no session yet because the connection is being queued. This is applicable only when queueing is set to *On*.
Session	Identifies the session associated with the connection, if any.

Statistic	Explanation
User	User ID that connected the session.
Authentication	Indicates the type of authentication used to connect: trusted, explicit, or IWA.
Requester ID	Unique identifier for the network connection created by a listener.
Time In	Indicates the time at which the connection was activated or queued. This is the time value used to decide when a queued connection times out.
Connected From	Indicates the network address of the client that connected.

Deferred Statistics

The Deferred Statistics page displays a list of statistics for deferred management. It also provides configuration of specific deferred execution parameters. Clicking *Configuration* allows for changes to the configuration of the following Deferred execution mode settings:

dfm_dir
dfm_int_min
dfn_int_max
dfm_maxage
dfm_maxoutput

Reference: Deferred Management Statistics

The statistics that appear are as follows:

- DFM_Dir Available Disk space is the amount of storage space available in the DFM_DIR directory. This allocation differs according to operating system. For Windows, it is the size of the drive where the server was configured.
- Number of Requests Done Since Startup is the total number of deferred requests that were executed since the server was started.
- Number of Response Ready is the total number of deferred reports stored in the DFM_DIR directory.

Deferred List

Reference:

Deferred FOCUS Reports

Extensions for Deferred Files

The Deferred List page is used to view the current status of all deferred reports, stop queuing or executing reports, or remove any report from the server.

Reference: Deferred FOCUS Reports

The Summary of Deferred Status is as follows.

Status	Description
No Error	Done as Requested.
Ready	Response file is ready to be picked up.
Deleted	Response file was deleted as requested.
Queuing	Request is waiting to be executed.
Bad Defer ID	Invalid ID or no request associated with that ID.
Bad User ID	Response File does not belong to that ID.
System Error	System failure, for example, out of memory or disk space.
Executing	Request is being executed.
Stopped	Request was stopped as requested.
Agent Crashed	Agent crashed during execution. Incomplete response file.
Connect Failed	Failed to connect to server. Communication error.
Unknown	Failed to determine request status. Try again.

Reference: Extensions for Deferred Files

Following is a list of possible extensions for the deferred files listed in the DFM_DIR Directory.

Extension	Description
RQD	Data file, contains user ID, optional flags, and so on.
RQP	Request file, indicates request is being executed.
RQF	Completed request file, ready to be executed.
RQI	Incomplete request file, request is being received.
RPF	Complete response file, ready to be retrieved.
RPI	Incomplete response file.
DEL	Request is deleted.

Monitoring Listeners and Special Services

Reference:

Listeners Statistics

Special Services Statistics

Clicking *Listeners* opens the Listeners page, which presents a tree view of listeners and special services in the left pane, and selected object statistics in the right pane. This enables administrators to view statistics of the listeners running in the Workspace Manager address space, and to quiesce and enable some listeners.

Note: Click *Refresh Now* to refresh the statistics that appear. The information can be refreshed at predefined intervals by clicking the check box and typing the number of seconds between each refresh.

Reference: Listeners Statistics

The statistics that appear for the Listeners list are:

Statistic	Description
Protocol	Identifies the type of the listener by displaying its network protocol (for example, HTTP, TCP, SNA/LU6.2, ...). **Note:** OpenVMS and some UNIX platforms do not support SNA/LU62.
Status	Status of the listener. Possible values include: Active, Not Active, and Stopped.
PID/Job #	Process identification number, or Job # on OS/400, associated with each listener running in a Workspace Manager address space. This job # was previously called PID on OS/400.
Port Number/Name	Port number or name associated with each listener.

Reference: Special Services Statistics

The statistics that appear for the Special Services list are:

Statistic	Description
Type	Displays the type of the special service (for example, FDS, DFM, ...).
Status	Status of the special service. Possible values include: Active, Not Active, and Stopped.
PID/Job #	Process identification number, or Job # on OS/400, associated with each special service running in Workspace Manager address space. This job # was previously called PID on OS/400.
Port Number	Port number associated with the special service, if applicable (for example, FDS).

Migrating Your Server

It is recommended that you migrate configuration information from previous releases after you verify the proper basic installation of the new release. To migrate from a previous release, type the full path of the configuration instance directory (EDACONF) on the Migrate page.

Configuring Traces

The Traces Configuration page enables you to set the size limit of the trace files and select tracing options for various components of the server. Any selection you make is written into IBITRACE.FEX and takes effect on the next server run.

The trace size limit specifies the maximum number of lines in the trace files. After the limit is reached, trace data is dumped to an alternate file with the extension TRB. Logging continues into the trace file with extension TRC.

There are an extensive number of components that can be traced, and it is recommended to change from the default setting to custom at the suggestion of Customer Support.

Component	**Description**
Default Components	• R1H • QOPSYS • PRH • NWH • NLS
All Components	All traceable server components.
Typical Components	• CEH • NWH2 • PRH • SQLAGGR • STMTRACE
Custom Components	Select components.

Traces can also be used to capture the server input and create a script file. To enable this type of tracing, select custom components and choose the NWHSIM trace component only.

The IBITRACE settings have:

```
SET TRACEON=NWHSIM
```

Editing Configuration Files

Manually editing configuration files requires extensive knowledge of the inner workings of the server. All files except VERSION.CFG are available for editing on the Edit Configuration Related Files page of the Web Console. VERSION.CFG is created by the installation/configuration process and is available for viewing only from this page.

Viewing and Editing Your License Number

You can view and change the current using the License Management page.

The license code is encoded to determine the configuration, additional products available in the configuration, the number of CPUs on the server, and the number of user seats the configuration supports. For more information on how to obtain a new license code, contact Customer Support or the local Information Builders sales office.

CHAPTER 11

Managing Listeners and Special Services

Topic:

The following section describes how to manage listeners and special services.

Using the Listeners and Special Services Configuration Page

Clicking *Listeners* in the navigation pane opens the Listeners and Special Services Configuration page, which presents a list of configured communication nodes. You can modify communication parameters of existing nodes, remove existing nodes, or add new nodes.

Note: You cannot delete the HTTP listener node. Otherwise, you cannot start the Administration Console after server restart.

There are three major components that must be set to establish communication between a client and a server: the Client, the Protocol, and the Server.

A communication configuration file is broken down into blocks called NODEs. Each NODE is identified by name and represents one client, one server (a listener node), or one cluster and identified by the keyword CLASS. Each block also describes all parameters necessary for communication, security, and so on.

Note: The client and agent must communicate using the same protocol.

CHAPTER 12

Troubleshooting

Topics:

- Viewing Version Information
- Analyzing Server Activity
- Analyzing FOCUS Database Server Activity
- Enabling and Viewing Trace Files
- Generating a Server Trace and Running Saved Diagnostics on OS/390 and z/OS
- Generating a System Dump on OS/390 and z/OS
- Freeing Data Sets Allocated to the Server on OS/390 and z/OS
- Correcting Orphaned Shared Memory Segments on OS/390 and z/OS
- Understanding a U4039 Abend on OS/390 and z/OS
- Correcting INSUFFICIENT AUTHORITY TO GETPSENT Messages on OS/390 and z/OS
- Retrieving IBISNAP Output
- Recording and Reproducing User Actions
- Troubleshooting the Console
- Workspace Manager Safe Mode
- Server Processes

The Web Console enables administrators to access several diagnostics tools that can be used to visualize different internal information. The following section describes the pages used to access such information to perform problem analysis tasks.

Viewing Version Information

The Version information page displays the main identification parameters of the server configuration that was in effect when the server was started. The parameters and their description are as follows:

Parameter	**Explanation**
Configuration Date	Indicates the configuration date of the Server.
Build date, gen number, release, source date	Identifies the release and build dates of a running instance of the Server.
Host Name	Name of the machine where the server is installed.
Server Name	As defined in the configuration file. **Note:** Used on Windows platforms as the system service name for the Server.

Analyzing Server Activity

In this section:

Controlling EDAPRINT History

The EDAPRINT page of the diagnostics section of the Web Console enables you to view the current or prior server activity log (edaprint.log) from either a management perspective (Session and Connection activity), or a raw version of the log with or without filtering for certain components and IDs. The activity log chronologically records all server activity since the Workspace Manager was started; it contains basic activity information as well as server start up information and IBISNAP and shutdown information. IBISNAP is a snapshot of the server environment showing various usage statistics, listener, and agent status. If an abend occurs, the snapshot contains additional debugger stack information that aids Solution Programming staff in determining the problem.

Click *View Using Selected Options* from the EDAPRINT page, with the *Unfiltered Log* radio button selected, to display the raw EDAPRINT log file in a new window.

If the log display selection has filtering, the output contains only selected lines from the EDAPRINT log file.

Management reporting from the EDAPRINT page contains several selections that range from detailed, summary, or a graph of session or connection activity.

You can also select a connection summary report, as well as a graphic representation.

A sample user-defined report is also available. To use this feature, click the EDIT consusr.fex, make changes as required, and click *Save*. Return to the EDAPRINT pane, click the *User Defined Report* selection, and then *View Using Selected Options*.

Controlling EDAPRINT History

The size of the server activity log and the number of files archived in the EDAPRINT history are controlled by and , respectively.

Limit the size of each log file by setting the maximum number of lines to a value other than zero. There is no minimum, but the recommended value is at least 1000 lines. If the limit is too low and the system is too fast (such that two archived files can have the same file system time stamp to the same second), whichever of the archived files is deleted first is undefined.

The first log file for each server start always contains more lines than the maximum because the count begins only after the edaplog process is created, thus it is larger by the size of the startup information. Each time the limit is reached, the current log file is archived and the new file starts with an additional identification line. This line indicates the rank of the file in the series of continuation files, and a reference to the archived name of the first file in the series (the one with the startup information), in a format similar to the following:

```
continuation #n of edapriNN
```

The number of files kept archived is determined by setting the edaprint_history parameter to the maximum number of edapriNN.log files. Each time the server is restarted or the current log reaches maximum size, the previous edaprint file is moved to edapriNN.log, and the maximum number of files is enforced by deleting the oldest files. The effective minimum number of files kept may be as high as 2 in addition to the current edaprint.log, even if edaprint_history was set to 0 or 1, because when files are split to respect edaprint_max_lines, the most recent file and the most recent file containing startup information are always preserved.

Analyzing FOCUS Database Server Activity

The HLIPRINT page of the diagnostics section of the Web Console enables you to view the HLIPRINT log file that logs information from the FOCUS Database Server (FDS) Service.

Click *Show HLIPRINT* to display the contents of the HLIPRINT log file in a new window.

Enabling and Viewing Trace Files

How to:

View a Trace File

The Traces page of the diagnostics section of the Web Console enables you to view traces and turn them dynamically on or off for a running server. It is only available to the server administrators. It does not appear for an end-user ID.

If tracing is set to off and it was never turned on, the page shows that no traces are available and enables you to turn traces on.

If you turned on tracing at server startup (for example, by using Workspace Start with Traces from the Diagnostics menu on a Windows system), or you click *Enable Traces*, the page displays the available traces in drop-down boxes. Which traces are available depends on what requests were made against the server.

Click *Enable Traces* to ask for confirmation before starting traces. This is in part to remind you that a dynamic trace is not the same as turning traces on at server startup. A dynamic trace is usually not sufficient for following a problem through with customer support, but it may suffice for other purposes, such as seeing how something is parsed.

To turn dynamic traces on or off for one user, SET TRACEUSER=ON or SET TRACEUSER=OFF must be present in the user profile.

SET TRACEUSER=tracename command can be used to turn dynamic traces on and set the trace name to tracename, where tracename includes the full path to the trace.

Note: The default for tracing is to trace all components when tracing is turned on. However, the trace settings file (ibitrace.fex) may have been altered. Therefore, components that you expect to be traced may not be traced. Click *Configuration* to access the Traces Configuration page to view and set a trace setting.

Procedure: How to View a Trace File

1. Select the radio button next to the Trace that you would like to view (Agent, WSM, Listener, or Installation Log).
2. Click *Show the File*.

   The trace file appears in a new browser window. If you selected Filtering and nothing matches the requirements, the new window contains the No Trace for Given Component List message.
3. Select the *Errors* radio button to filter existing trace files for errors before viewing the file.

Understanding a U4039 Abend on OS/390 and z/OS

In some instances, the server may abend with a U4039 code. This is a generic abend.

Solution: Determine what caused the abend by checking the **edaprint.log** file, SYSOUT DDname and the MVS system log. If you suspect this is a bug, follow the instructions in *Generating a Server Trace and Running Saved Diagnostics on OS/390 and z/OS* on page 12-5 and *Generating a System Dump on OS/390 and z/OS* on page 12-5 in order to collect information to send to iWay Software.

Generating a Server Trace and Running Saved Diagnostics on OS/390 and z/OS

To generate a server trace:

1. Turn tracing on from the Web Console or by running the ITRCON JCL member.
2. Reproduce the problem.
3. Submit the ISAVEDIA member to produce the diagnostic information.

A directory called sd*nnnnnn* is created under your configuration directory (for example, /ibi/server53/ffs/sd123456). Diagnostic information is placed in this directory. Verify that you have access to this directory.

Do not change anything in the EDAENV DD statement; changes could prevent the correct information from being copied to your directory.

Generating a System Dump on OS/390 and z/OS

To generate a system dump:

1. Allocate DDNAME SYSMDUMP pointing to the data set with the following DCB parameters:

   ```
   RECFM=FB,LRECL=4160,BLKSIZE=4160.
   ```

2. To get the first dump, add the parameter FREE=CLOSE to your DD statement. The DD statement should appear as follows:

   ```
   //SYSMDUMP DD DISP=SHR,DSN=MYID.EDAPTH.SYSMDUMP,FREE=CLOSE
   ```

3. To get the last dump, the statement should appear as follows:

   ```
   //SYSMDUMP DD DISP=SHR,DSN=MYID.EDAPTH.SYSMDUMP
   ```

   Only two IDs must have privileges to write into this data set: ISERVER and IADMIN. General server users DO NOT need read or write access to the SYSMDUMP data set.

4. To prevent abendaid from intercepting the dump, add:

   ```
   //ABNLIGNR DD DUMMY
   ```

5. To prevent Language Environment from intercepting the dump, specify:

   ```
   EDADUMPOPT=UAIMM in EDAENV DD
   ```

   This enables you to get more accurate information reflecting the moment the abend actually occurs.

6. Save the entire job output for the server (including JES logs).

Freeing Data Sets Allocated to the Server on OS/390 and z/OS

How to:

Free a Dataset From the MVS System Console

Example:

Freeing the Allocated Data Set

An MVS operator can issue modify commands to free DDNAMES (or dsnames) allocated to the IWAY server. Both global allocations (made at the server ISTART JCL) and local ones (DYNAM ALLOC commands issue by the user tasks) can be freed. This procedure is useful if you need to free an allocation to run a batch utility overnight, without restarting the server.

Syntax: How to Free a Dataset From the MVS System Console

To free a single DDNAME:

```
F <iway_server_jobname/started task>,DYNAM FREE FI <ddname>
```

To free a single dsname (all occurrences in the server):

```
F <iway_server_jobname/started task>,DYNAM FREE DA <dsname>
```

To free multiple DDNAMEs, passing a pattern (free all ddnames staring with AB:

```
F <iway_server_jobname/started task>,DYNAM FREE FI AB*
```

To free a multiple dsname (all occurrences in the server), passing a pattern (free all allocations of data sets starting with IWAY.VSAM*:

```
F <iway_server_jobname/started task>,DYNAM FREE DA IWAY.VSAM*
```

A message will be issued in the iway_server JESMSGLG indicating if the command was process successfully or not, as follows:

success:

```
@DYNAM COMMAND SUCCESFULLY PROCESSED Rc=0
```

failure:

```
@DYNAM ERROR: IKJ56225I DATA SET IWAY.TEST ALREADY IN USE, TRY LATER
```

Once dataset is freed and the batch utility runs, the operator can re-allocate the dataset to the server, using the same modify command, but this time issuing DYNAM ALLOC, instead of DYNAM FREE.

Example: Freeing the Allocated Data Set

Suppose ISTART JCL (jobname IWAYP) has the following allocation:

```
//VSAMFILE DD DISP=SHR,DSN=VSAM.FILEA.CLUSTER
```

The operator can free this file using the command (from MVS console):

```
F IWAYP,DYNAM FREE VSAMFILE
```

Later, after the batch utility runs, the operator can re-alloacate the file to the server by issuing the command:

```
F IWAYP,DYNAM ALLOC F VSAMFILE DA VSAM.FILEA.CLUSTER SHR REU
```

All valid DYNAM ALLOC and DYNAM FREE syntaxes are supported. For more information on DYNAM command, please refer to the appropriate WebFOCUS manual.

Correcting Orphaned Shared Memory Segments on OS/390 and z/OS

When the server is restarted after an abend, an orphaned shared memory segment is created. The accompanying error message reads:

```
Shared memory remains but WSM process is gone
```

1. To display the orphaned memory objects, issue the command:

   ```
   IPCS M -X
   ```

2. To removed the orphaned memory objects, issue the command:

   ```
   IPCRM M
   ```

   IPL will eliminate the orphaned shared memory segment.

3. Add the following EDASHARE pointer (with the client's directory path) to the ISTART EDAENV DD:

```
000050 //EDAENV      DD      *
000051 TZ=EST5EDT
   000053 EDASHARE=/ibi/srv52/ffs/web
```

where:

web

Is a newly created directory, but can be any name.

This will point the server to a new shared memory segment. This syntax should also be added to the ISTOP jcl.

Correcting INSUFFICIENT AUTHORITY TO GETPSENT Messages on OS/390 and z/OS

INSUFFICIENT AUTHORITY TO GETPSENT messages may appear in JESLOG. For more information on causes of these messages, see IBM's APAR II11813.

The APAR recommends issuing one of the following RACF commands to work around this problem:

```
SETROPTS LOGOPTIONS (NEVER(PROCACT))
```

```
SETOPTS LOGOPTIONS (DEFAULT(PROCACT))
```

However, when a non-superuser in the OMVS shell issues the command ps -ef, the following security message is repeated in SYSLOG:

```
ICH408I USER(default) GROUP(dgltgrp) NAME(bpxdefaultuser) 060
```

```
CL(PROCACT ) INSUFFICIENT AUTHORITY TO GETPSENT
```

This does not indicate an error; it is an informational message issued because of RACF LOGOPTIONS settings. The ps -ef command is a request to show all processes that the requester is authorized to see, but a non-superuser is allowed to see only his or her own processes.

Retrieving IBISNAP Output

When an agent crashes, an IBISNAP is produced (the file name starts with CEEDUMP and is followed by a system generated timestamp). IBISNAP is moved automatically into edaprint.log by the edapth process as soon as it identifies that the agent has crashed (by default, this check runs once every minute). You can also be force this move to occur by clicking the *Writing IBISNAP information to current log* button on the Web Console Diagnostics / EDAPRINT screen. The savediag utility will also preserve this file (if it has not been already moved into edaprint.log).

Recording and Reproducing User Actions

In this section:

Playback Files

Reference:

General Playback Parameters

Recording Hints

The Recording and Playback features enable the server to record the exact sequence of user actions applied through a browser, and then reproduce it, by playback, as a single user or multiple users under the same or different conditions. Files generated by record and playback—known as HTI script (HTTP internal script)—along with server traces are used for problem analyzing and testing.

Note:

- Recording and Playback are available only for Common Gateway Interface (CGI) connections.
- Click *Refresh Now* to refresh the statistics. The information can be refreshed at predefined intervals by clicking the check box and typing the number of seconds between each refresh.

After recording begins, a script file is created. This file is located in the catalog directory of the server. The file name can be edited before recording starts. Check *Append to Existing* to append any recording to the existing HTI script.

All commands issued through a CGI connection are recorded. These commands can be run using the Procedures pane on the Web console.

Click *Stop* to stop the recording.

Use the drop-down list to select the Script Name that was created. This Script can be edited by using Edit Script. Before starting the playback, click *Playback*, and select the parameters to be used.

Reference: General Playback Parameters

Following is a list of general playback parameters that are available.

Parameters	Description
Number Of Threads	A positive number of threads to be used when playing the script. Each thread represents a single client (user).

Parameters	**Description**
Interval Parameter	Specifies timing in a multiple-client (multi-threaded) playback. Possible formats of this value are *m* or *m, n, k*, where *m* is the number of seconds between each client startup, and after each *n* started clients, an interval of *k* seconds is used instead.
Compare Option	Determines if data received by each client (thread) is identical. Uses binary comparison mode and writes comparison result message to playback log file.
Immediate Processing	Ignores every SLEEP and WAIT statement in the HTI script. Issues HTTP requests without any delay between them.
Size Statistics	Displays general output file size statistics for multiple-client playback in the log. Useful when analyzing results of a playback with a large thread number.
Traces	Sets the Playback utility traces on or off. A trace file is then created for each thread.
Result	By default, stores all data received by a thread in a single file.
Keep alive	Causes every thread to repeat a request for a specified time.
Redirect to ODIN node	Sends a request to the node defined in the communications configuration file, odin.cfg.
Redirect to HTTP	Sends a request to the Web server (HTTP listener), for example, www.myhost.com:8101.
Submit as deferred	Sends a request as deferred.

Reference: Recording Hints

When recording is on, the HTTP listener records every CGI request. Ensure that no one else is submitting requests to the same listener.

It is recommended that you open a separate instance of a browser before recording, and that you deal with actual operations to be recorded and switch recording on or off in two different windows.

Do not try to record script from the middle of a persistent session. Always try using user ID and password, then recording the first request.

Playback Files

Playback log files are generated during script playback. Each file contains general information about playback processing, such as thread startup and termination, connection errors, and comparison results.

A log file name is built using script file name and extension log. Two additional log files (stdout and stdlog), if not empty, contain a description of errors in the event of playback failure caused by critical errors.

Playback hto files are generated after script playback. Every file contains data received by a thread.

An hto file name is built using script file name, thread number, and extension hto.

Troubleshooting the Console

If the web browser has problems accessing the Web Console, you may see the following messages.

Error Message	Troubleshooting Tips
Internet Explorer cannot open the Internet site http://address:http_service. A connection with the server could not be established.	Ensure that the Workspace Manager is running or else contact the local system administrator.
There was no response. The server could be down or is not responding. If you are unable to connect again later, contact the server's administrator.	Ensure that the Workspace Manager is running or else contact the local system administrator.

Workspace Manager Safe Mode

If the configuration files contain non-fatal errors that prevent the server from coming up in a consistent state, the Workspace Manager starts in *safe mode*. In this mode only the Web Console is operational. This enables the administrator to check for errors using the Diagnostics EDAPRINT page of the Web Console (the errors that trigger the safe mode would be visible in the edaprint log), correct the problem using the Workspace Configuration page(s) and then restart the server.

Server Processes

In this section:

The server uses a series of processes to accomplish various tasks. Some processes have specialized administrative functions, such as workspace management and logging, while others do end-users tasks. System administrators may want to know what processes the server creates and how many of each. The administrator may want this information for machine sizing or simply to know what processes to consider or ignore when a problem occurs.

Note that each operating environment uses operating system specific commands to view processes. The processes are described generically, not how they would appear in a system listing.

Server Work Space Manager Daemon

The server workspace is control by a daemon process know as PTH. There is one process per server.

Deferred Listener Daemon

The server has several types of listeners, the DFM daemon is used for deferred requests. There is one process per server.

Server Log Daemon

The server log (edaprint.log) is now written via a single daemon process. Previous versions wrote to the log independently from the various processes that might write to the log. In very rare and specific cases, the prior methods were problematic, hence, the change to the use of a daemon. As a result, the starting a server brings up this additional server log daemon (edaplog).

Server Check Up Daemon

The server has always had a process that wakes up and checks various states and takes actions. An example of this is stopping agents that are inactive past the idle limit. In the past this process has been part of the Workspace Manager function, but is now a separate daemon that runs independently. As a result, starting a server brings up this additional server check up daemon (edachkup).

HTTP Listener Daemon

The server has several types of listeners. The HTTP daemon is used for requests using the HTTP protocol. There is one process per server.

TCP Listener Daemon

The server has several types of listeners. The TCP daemon is used for requests using the TCP protocol. There is one process per server.

FDS Listener Daemon

The server has several types of listeners. The FDS daemon (HLISNK) is used for FOCUS Database requests from agent processes. There is one process per server.

Java Listener Daemon

The server has several types of listeners. The JSCOM daemon is used for requests using Java. There is one process per server.

Agent Daemons

The actual worker processes for request are know as agents or tscom3 processes (tscom3 is the actual program name). There is one process per configured agent that is active. Agent processes normally are reused until a recycle point or until they become inactive for a set period. Thus, for a give server, the number of processes associated with agents will change over time.

Index

Reader Comments

In an ongoing effort to produce effective documentation, the Documentation Services staff at Information Builders welcomes any opinion you can offer regarding this manual.

Please use this form to relay suggestions for improving this publication or to alert us to corrections. Identify specific pages where applicable. You can contact us through the following methods:

Mail: Documentation Services - Customer Support
Information Builders, Inc.
Two Penn Plaza
New York, NY 10121-2898

Fax: (212) 967-0460

E-mail: books_info@ibi.com

Web form: http://www.informationbuilders.com/bookstore/derf.html

Name:____

Company:____

Address:____

Telephone:_________________________________Date:___________________________________

E-mail:____

Comments:

Reader Comments

Information Builders, Two Penn Plaza, New York, NY 10121-2898 **(212) 736-4433**