

Entire Net-Work

Administration

Version 7.7.1

November 2017

This document applies to Entire Net-Work Version 7.7.1 and all subsequent releases.

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

Copyright © 2017 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA, Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <http://softwareag.com/licenses>.

Use of this software is subject to adherence to Software AG's licensing conditions and terms. These terms are part of the product documentation, located at <http://softwareag.com/licenses/> and/or in the root installation directory of the licensed product(s).

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at <http://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with Software AG.

Document ID: WCPOS-OWCPOSADMIN-771-20171109

Table of Contents

Preface	v
1 About the System Management Hub	1
Commands for Managing the System Management Hub Service in UNIX	2
Accessing the System Management Hub	2
Leaving the System Management Hub	3
Using the Refresh Button in the System Management Hub	4
Getting Help	4
2 Port Number Reference	5
Port Overview and General Assignments	6
Changing the Adabas Directory Server Port Number	7
About System Management Hub Ports	9
3 Directing Log Files to a Shared Server	11
Step 1. Specify the Log File Locations	12
Step 2. Configure the Entire Net-Work and Entire Net-Work Client Windows Services	12
4 Entire Net-Work Configuration Parameters	15
Configuration Parameter List	16
Configuration API Functions	19
5 Entire Net-Work Service Function Utility (wcpadmin)	23
Syntax	24
Special Character Use	28
Creating a Kernel Using wcpadmin	28
General Examples	29
6 Entire Net-Work Utility Functions for the Directory Server (checkadi and setadi)	37
The checkadi Utility	38
The setadi Utility	39
Index	41

Preface

This document describes how to perform the system administration tasks necessary for Entire Net-Work operations using the System Management Hub (SMH).

It is intended for system administrators in your enterprise.

This document is organized as follows:

<i>About the System Management Hub</i>	Introduces you to the System Management Hub and explains how to access it and leave it.
<i>Entire Net-Work Client Administration</i>	For information about Entire Net-Work Client administration tasks, read <i>Entire Net-Work Client Installation and Administration Guide</i> .
<i>Entire Net-Work Server Administration</i>	For information about Entire Net-Work Server administration, read <i>Entire Net-Work Server LUW Installation and Administration Guide</i> .
<i>Port Number Reference</i>	Describes the ports that are needed by Entire Net-Work to perform its processing and how they can be assigned.
<i>Directing Log Files to a Shared Server</i>	Describes the process of directing your Entire Net-Work log files to a shared server.
<i>Entire Net-Work Configuration Parameters</i>	Lists the Entire Net-Work configuration parameters and describes batch and application program tools available for you to set them.
<i>Entire Net-Work Service Function Utility (wcpadmin)</i>	Describes how to use the Entire Net-Work Service Function Utility (wcpadmin) to perform some of the Entire Net-Work service functions in batch mode.
<i>Entire Net-Work Directory Server Utility Functions (checkadi and setadi)</i>	Describes how to use the Entire Net-Work checkadi and setadi utility functions to check for the existence of a Adabas Directory Server and to set Directory Server access parameters for Entire Net-Work and Entire Net-Work Client.

1 About the System Management Hub

- Commands for Managing the System Management Hub Service in UNIX 2
- Accessing the System Management Hub 2
- Leaving the System Management Hub 3
- Using the Refresh Button in the System Management Hub 4
- Getting Help 4

The System Management Hub (SMH) is a Web-based graphical user interface (GUI) you can use to perform administrative tasks for some Software AG products, including Adabas Directory Server, Adabas Administration Services, and Entire Net-Work. It runs in a standard Web browser.

Before you start using the System Management Hub, you must set up an administrative user for the product. To do so, consult the *Add Administrator* section of the System Management Hub documentation, available on Empower.

This chapter provides a high-level overview of the System Management Hub.

Commands for Managing the System Management Hub Service in UNIX

After you set the environment for running Software AG products, as outlined in the previous section, you can issue the following commands to start, stop, restart, pause, and resume the System Management Hub on UNIX.



Note: Before issuing these commands, open a terminal window, and change to the `$SAG/InstanceManager` directory.

Command	Description
<code>argsrvs.bsh start</code>	Starts the System Management Hub.
<code>argsrvs.bsh stop</code>	Stops the System Management Hub.
<code>argsrvs.bsh restart</code>	Restarts the System Management Hub.
<code>argsrvs.bsh pause</code>	Pauses the System Management Hub.
<code>argsrvs.bsh resume</code>	Resumes the System Management Hub.

Accessing the System Management Hub

➤ To access the System Management Hub:

- 1 Type the following URL into your Web browser:

```
http://smh-mil-node:smh-mil-http-port/smh/login.htm
```

where `smh-mil-node` is the name of the machine where the System Management Hub (SMH) is running (normally this is "localhost") and `smh-mil-http-port` is the port number (the default is 49981) for the SMH MIL (Management Independent Layer) server.

 **Note:** If SMH has been installed on an Apache Web server, replace `smh-mil-http-port` with the port number of the Apache Web server (the default is 80) rather than the SMH MIL server.

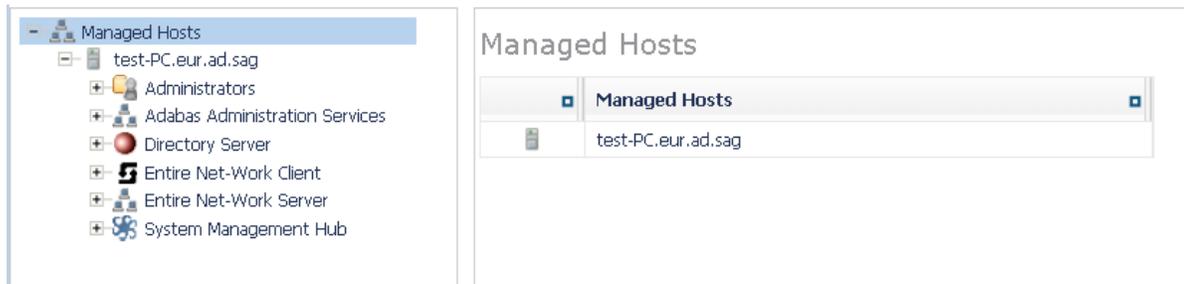
Or:

Select **System Management Hub** on the **Software AG Base Technology** Start Programs submenu (Windows only) and then select **Web Interface** on the resulting submenu.

The login screen for the System Management Hub (SMH) appears.

- 2 Login to the System Management Hub, as described in the section entitled *Internal HTTP Server* under *System Management Hub Web Interface* in *System Management Hub Interfaces and Tools* .

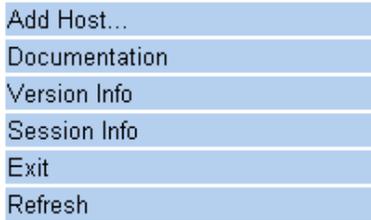
The System Management Hub main panel appears on the **System Management** tab.



Leaving the System Management Hub

➤ To leave the System Management Hub:

- Click the Log Off command at the top of the screen.



Host	
host.address.1	

Or:

Close the Browser window.

The System Management Hub window is closed.

Using the Refresh Button in the System Management Hub

Refresh buttons appear in the command frame of the System Management Hub for many panels. Use the **Refresh** button to update the values of items listed in the detail-view frame.

Getting Help

➤ To get help on an detail-view frame:

- If it is available, click the **Help** button in the detail-view frame of the System Management Hub screen.

The documentation pertaining to that System Management Hub view appears.

For complete information about the System Management Hub, read its documentation, available on Empower.

2 Port Number Reference

▪ Port Overview and General Assignments	6
▪ Changing the Adabas Directory Server Port Number	7
▪ About System Management Hub Ports	9

This chapter describes the ports that are needed by Adabas LUW and Entire Net-Work LUW products to perform its processing and how they can be assigned.

Port Overview and General Assignments

The following table describes the ports that are needed by Entire Net-Work to perform its processing and any default ports assumed by Entire Net-Work. You should consider avoiding the use of these default port numbers for other applications.

Software AG Product Component	Ports Needed	Default Port Number
Adabas Manager Communication Client	One port is needed.	4980
Adabas Directory Server	One port is needed for Entire Net-Work requests to the Directory Server	4952 (IANA port) Note: If older versions of Entire Net-Work (older than 7.3) are in use, this port number may need to be changed to 12731.
Entire Net-Work Administration LUW	One port is needed for System Management Hub (SMH) administration tasks	dynamically assigned
Entire Net-Work Kernel	A port is needed for Kernel access by clients	dynamically assigned
	A port is needed for Kernel access via e-business connections (Entire Net-Work 7 or later)	dynamically assigned
	A port is needed for Kernel access via classic RDA connections (Entire Net-Work 2)	7869
	A port is needed for System Management Hub (SMH) administration of Kernels	dynamically assigned

Software AG has registered port number 4952 with the Internet Assigned Numbers Authority (IANA) for use by the Adabas Directory Server. For more information about Directory Server port number specifications, read *The Directory Server Port Number* in the *Software AG Directory Server Installation and Administration Guide*. For information on changing the Directory Server port number for an Entire Net-Work installation, read [Changing the Adabas Directory Server Port Number](#).

In general, there are no default port numbers assigned to Entire Net-Work Kernels or clients. These are dynamically assigned by Entire Net-Work when the Kernel or client is started, unless you specify a specific port or range of ports to use when you define the Kernel or client. If you set the port number to "0", the Entire Net-Work will dynamically assign a port.

Port numbers are dynamically assigned by Entire Net-Work when the Kernel or client is started, as follows:

- Entire Net-Work searches for the first available port starting from port 49152 through 65535. (The starting search port number, 49152, is the IANA-recommended value from which to start.)
- Once an available port number is found, it is assigned to the Kernel or client in its Adabas Directory Server entry.

While defining Entire Net-Work Kernels, you can also select a specific port or specify a range or list of port numbers that Entire Net-Work should search during the process in which it dynamically assigns a port to the Kernel:

- To specify a specific port number, enter the number in the port number field when you define the Kernel.
- To specify a range of port numbers that Entire Net-Work should search to dynamically assign a port, list the starting and ending ports in the port number field when you define the Kernel, separated by a dash (-). For example, a specification of "9010-9019" would cause Entire Net-Work to search for the first available port between and including port numbers 9010 and 9019.
- To specify a list of port numbers that Entire Net-Work should search to dynamically assign a port, list the port numbers in the port number field when you define the Kernel, separated by commas (.). For example, a specification of "9010,9013,9015,9017,9019" would cause Entire Net-Work to search for the first available port from this list of ports, starting with port 9010 and working from left to right through the list.
- You can, of course, combine search ranges and lists in a port number field. For example, a specification of "9010-9019,10020,10050-10059" would cause Entire Net-Work to search for the first available port first in the 9010-9019 range (inclusive), then port 10020, and finally in the 10050-10059 range (inclusive). The first available port that Entire Net-Work encounters would be used for the Kernel.

If no available port is found in a specified range or list, an error occurs.

For more information about adding Kernels, read *Adding Kernel Configuration Definitions* in the *Entire Net-Work Server LUW Installation and Administration Guide*.

Changing the Adabas Directory Server Port Number

➤ If you need to change the Directory Server port number for your installation, follow these steps:

- 1 Within the settings for Entire Net-Work Client and any client configurations definitions, change all specifications for the Directory Server port number to the new port number you want to use. Directory Server port numbers can be changed for Entire Net-Work Client and the client configurations using the System Management Hub (SMH), as follows:
 1. Start up SMH and access the Entire Net-Work Client SMH administration area. For more information about the Entire Net-Work Client SMH administration area, read *The Entire*

Net-Work Client SMH Administration Area, in the *Entire Net-Work Client Installation and Administration Guide*.

2. Right-click on the name of a client machine listed under **Clients** in the Entire Net-Work Client SMH administration area.
3. Select the **Set Parameters** command from the drop-down menu that appears.

The **Set Client Parameters** panel appears in detail-view. For complete information about this screen, read *Setting Client Parameters*, in the *Entire Net-Work Client Installation and Administration Guide*.

4. On the **Set Client Parameters** panel, change the Directory Server port number to the new port number you want to use in the SAGXTSDSPORT field.
 5. On the **Set Client Parameters** panel, click on **Update all Client Configurations**. A check mark should appear for this option.
 6. Click **OK** to save the settings for the client machine and all of the client configurations associated with it.
- 2 Within the settings for Entire Net-Work Server and any Kernels definitions, change all specifications for the Directory Server port number to the new port number you want to use. These port numbers can be changed using the System Management Hub (SMH), as follows:

1. Start up SMH and access the Entire Net-Work Server SMH administration area. For more information about the Entire Net-Work Server SMH administration area, read *The Entire Net-Work Server SMH Administration Area*, in the *Entire Net-Work Server LUW Installation and Administration Guide*.
2. Right-click on the name of an Entire Net-Work Server listed under **Servers** in the Entire Net-Work Server SMH administration area.
3. Select the **Set Server Parameters** command from the drop-down menu that appears.

The **Server Parameters** panel appears in detail-view. For complete information about this screen, read *Setting Server Parameters*, in the *Entire Net-Work Server LUW Installation and Administration Guide*.

4. On the **Server Parameters** panel, change the Directory Server port number to the new port number you want to use in the SAGXTSDSPORT field.
 5. On the **Server Parameters** panel, click on **Update all Kernels**. A check mark should appear for this option.
 6. Click **OK** to save the settings for the server and all of the Kernels associated with it.
- 3 Shut down the Entire Net-Work Client service or daemon and the Entire Net-Work Server service or daemon, as appropriate. Be sure to shut down every Kernel associated with the server as well.

For information on shutting down the Entire Net-Work Client service or daemon, read *Stopping Entire Net-Work Client* in the *Entire Net-Work Client Installation and Administration Guide*. For information on shutting down the Entire Net-Work Server service or daemon, read *Stopping Entire Net-Work Server* in the *Entire Net-Work Server LUW Installation and Administration Guide*.

- 4 Shut down the Directory Server service or daemon.

For information on shutting down the Directory Server service or daemon, read *Starting and Stopping the Adabas Directory Server*, in the *Software AG Directory Server Installation and Administration Guide*.

- 5 Modify the Directory Server installation, as appropriate for the operating system. When prompted, change the Directory Server port number to the new port number you want to use.
- 6 Start up the Directory Server service or daemon, if it is not automatically started after its installation was modified.

For information on starting up the Directory Server service or daemon, read *Starting and Stopping the Adabas Directory Server*, in the *Software AG Directory Server Installation and Administration Guide*.

- 7 Start up the Entire Net-Work Client service or daemon and the Entire Net-Work Server service or daemon.

For information on starting up the Entire Net-Work Client service or daemon, read *Manually Starting Entire Net-Work Client* in the *Entire Net-Work Client Installation and Administration Guide*. For information on starting up the Entire Net-Work Server service or daemon, read *Manually Starting Entire Net-Work Server* in the *Entire Net-Work Server LUW Installation and Administration Guide*.

About System Management Hub Ports

For information about any System Management Hub installation issues, including port number settings, read *Installing webMethods Products* in Empower.

3 Directing Log Files to a Shared Server

- Step 1. Specify the Log File Locations 12
- Step 2. Configure the Entire Net-Work and Entire Net-Work Client Windows Services 12

If you are using Entire Net-Work 7.3.3 or Entire Net-Work Client 1.3 or later, you can direct your Entire Net-Work log files to a shared server.



Caution: To avoid overwriting log files with the same name, log files for individual servers, Kernels, and clients should be stored in directories with unique names.

The process of directing log files to a shared server involves the steps (note that the second step is only required on Windows) described in this chapter.

Step 1. Specify the Log File Locations

Using the System Management Hub (SMH), specify the fully-qualified path of the directory in which you want to store the log files.



Caution: To avoid overwriting log files with the same name, log files for individual servers, Kernels, and clients should be stored in directories with unique names.

- For information on redirecting server log files, read *Specifying the Server Log File Location*, in the *Entire Net-Work Server LUW Installation and Administration Guide*.
- For information on redirecting Kernel log files, read *Specifying the Kernel Log File Location*, in the *Entire Net-Work Server LUW Installation and Administration Guide*.
- For information on redirecting Entire Net-Work Client log files, read *Specifying the Client Log File Location*, in the *Entire Net-Work Client Installation and Administration Guide*.

Make sure that your network administrator has allowed your local machine access to the directory and server to which you are redirecting the log files. On Windows systems, you must also complete the next step to do this.

Step 2. Configure the Entire Net-Work and Entire Net-Work Client Windows Services

On Windows systems only, you must configure the Entire Net-Work and Entire Net-Work Client services so that the local host can write to the log files on the shared server.

➤ To update the Entire Net-Work and Entire Net-Work Client Windows services appropriately, follow these steps:

- 1 Edit the Windows service definition for the Directory Server and select the **Log On** tab.
- 2 On the **Log On** tab, select the **This account** radio button.

- 3 Enter a user account name that is known to both this host and the file server where the log files are located. This can be a domain account or a local account that is configured on both machines with the same password. The account should have full control access rights to the log file location.
- 4 Click the **OK** button.
- 5 Restart the service.

4 Entire Net-Work Configuration Parameters

- Configuration Parameter List 16
- Configuration API Functions 19

Entire Net-Work configuration settings are stored in a series of parameters. Some of these you can change; some you cannot or should not. You can change these configuration settings using:

- Entire Net-Work System Management Hub (SMH) functions. For more information on this, read *Entire Net-Work Client Administration* (in the *Entire Net-Work Client Installation and Administration Guide*) and *Entire Net-Work Server Administration* (in the *Entire Net-Work Server LUW Installation and Administration Guide*).
- Environment variables with the same names as the configuration parameter.



Caution: Future releases of Entire Net-Work will not support configuration parameter settings in environment variables. For this reason, we recommend that you do not use this method.

- API functions supplied with Entire Net-Work that can also be used in your application programs. Additional information on these functions is provided elsewhere in this chapter.

Configuration Parameter List

The following table describes each of the Entire Net-Work configuration parameters, indicating whether or not you can or should modify its setting, the platforms on which it is available, the default (if any), the name of the SMH parameter that can be used to change it, and the name of the tool that can be used to change it.

Parameter	Description	Can you change it?	Platform Availability	Default	SMH Parameter	Batch Tool
LNKADAESI	whether the external security interface online application should be used instead of a user exit.	yes	Windows	NO	LNKADAESI	AdaSetParameter (set for the LNKADAESI parameter)
LNKADASAF	the library and function names of the user exit that will provide access to the secured Adabas resource via the Adabas SAF Security Kernel (ADASAF).	yes	UNIX and Windows	"lnkxsaf lnkxsaf"	LNKADASAF	AdaSetParameter (set for the LNKADASAF parameter)
LNKTIMEOUT	Global timeout for a response from a remote Adabas call.	yes	UNIX and Windows	1 minute	ADABAS_TIMEOUT	AdaSetTimeout
LNKTRACE	ADALNK and Software AG transport services trace levels.	yes	UNIX and Windows	0 (zero)	LNKTRACE and XTSTRACE	AdaSetTrace

Parameter	Description	Can you change it?	Platform Availability	Default	SMH Parameter	Batch Tool
LNKUEX_0	<p>the file and function names of the user exit that should be called before an Adabas ACB command is sent to the database.</p> <p>Note: LNKUEX_0 handling triggers an undocumented Natural feature; if the user exit is called and returns a non-zero response code, but the Adabas command is an RC command, the RC command is suppressed and a successful return is indicated to the calling program.</p>	yes	UNIX and Windows	—	LNKUEX_0	none
LNKUEX_1	<p>the file and function names of the user exit that should be called after an Adabas ACB command is sent to the database.</p>	yes	UNIX and Windows	—	LNKUEX_1	none
LNKUEX_ACBX_0	<p>the file and function names of the user exit that should be called before an Adabas ACBX command is sent to the database.</p> <p>Note: LNKUEX_ACBX_0 handling triggers an undocumented Natural feature; if the user exit is called and returns a non-zero response code, but the Adabas command is an RC command, the RC command is suppressed and a successful return is indicated to the calling program.</p>	yes	UNIX and Windows	—	LNKUEX_ACBX_0	none
LNKUEX_ACBX_1	<p>the file and function names of the user exit that should be called after an</p>	yes	UNIX and Windows	—	LNKUEX_ACBX_1	none

Parameter	Description	Can you change it?	Platform Availability	Default	SMH Parameter	Batch Tool
	Adabas ACBX command is sent to the database.					
WCLCODEDIR	the directory where the Entire Net-Work Client code is stored If you want to change the value of this configuration parameter, contact your Software AG support representative.	yes, with assistance	UNIX	local directory	none	none
WCLDATADIR	the directory where the Entire Net-Work Client data is stored If you want to change the value of this configuration parameter, contact your Software AG support representative.	yes, with assistance	UNIX	local directory	none	none
WCPCONFIG	Entire Net-Work configuration file name	yes	UNIX and Windows	local <i>xts.config</i>	none	AdaSetParameter (set for the WCPCONFIG parameter)
WCPDIR	the directory where the Entire Net-Work code is stored If you want to change the value of this configuration parameter, contact your Software AG support representative.	yes, with assistance	UNIX and Windows	local directory	none	none
WCPVERS	the Entire Net-Work version If you want to change the value of this configuration parameter, contact your Software AG support representative.	yes, with assistance	UNIX and Windows	—	none	none

Configuration API Functions

This section describes the API functions you can use to set some of the Entire Net-Work configuration parameters.

- [AdaSetParameter API Function](#)
- [AdaSetTimeout API Function](#)
- [AdaSetTrace API Function](#)
- [AdaSetSaf API Function](#)

AdaSetParameter API Function

You can use the AdaSetParameter API function to set values for the following parameters:

- LNKADAESI: Valid values are "YES" (use the external security interface online application) or "NO" (use a user exit). The default is "NO". If LNKADAESI is set to "YES" and a value is given in LNKADASAF, the online application is used (LNKADAESI settings override LNKADASAF).
- LNKADASAF: Specify the library and function names of the user exit that will provide access to the secured Adabas resource via the Adabas SAF Security Kernel (ADASAF). The library and function names should be specified with a space between them, using the following format:

```
library function
```

If no names are specified, the value "lnkxsaf lnkxsaf" is used. (The lnkxsaf library is either *lnkxsaf.dll* or *lnkxsaf.so*).

- WCPCONFIG: Valid values for WCPCONFIG can be any valid file name.

The syntax of the AdaSetParameter API function is:

```
AdaSetParameter("parameter=value")
```

Replace *parameter* with one of the parameter names listed above and *value* with an appropriate value for that parameter.

In the following example, the name of the Entire Net-Work configuration file to be used is set to TEST.CFG:

```
AdaSetParameter("WCPCONFIG=TEST.CFG")
```

AdaSetTimeout API Function

Use the AdaSetTimeout API function to set a global time limit for a response from a remote Adabas call. The syntax of the AdaSetTimeout API function is:

```
AdaSetTimeout(dbid, seconds)
```

Replace *dbid* with a valid Adabas database ID and *seconds* with the number of seconds to use for the global time limit for the specified database.

In the following example, a 60-second timeout period is defined for database 12:

```
AdaSetTimeout(12,60)
```

AdaSetTrace API Function

Use the AdaSetTrace API function to set trace levels for ADALNK and Software AG transport services. The syntax of the AdaSetTrace API function is:

```
AdaSetTrace(level, {TRUE | FALSE})
```

Replace *level* with a valid trace level. Trace levels must be specified in hexadecimal and in the following format:

```
0xllxxxx
```

In the trace level syntax, substitute a hexadecimal value from "00" through "f1" for *ll* to represent the ADALNK trace level. Then substitute a hexadecimal value from "0000" through "FFFE" for *xxxx* to represent the Software AG transport services trace level.

Then specify TRUE or FALSE. If you specify TRUE, the trace level is set globally for all calls; if you specify FALSE, the trace level is set only for the thread in which the call is made.

In the following example, the maximum trace level is specified for both ADALNK and Software AG transport services, but only for the thread in which the call is made:

```
AdaSetTrace(0xf1FFFE, FALSE)
```

In the following example, the maximum trace level is specified for ADALNK calls and no tracing is performed for Software AG transport services. These settings are made globally for all calls:

```
AdaSetTrace(0xf10000,TRUE)
```

AdaSetSaf API Function

Use the AdaSetSaf API function to specify external security interface access information to the user exit that will provide access to the secured Adabas resource via the Adabas SAF Security Kernel (ADASAF). This is the same information you supply using the external security interface online application (read *Accessing z/OS Resources Using the Online Security Application*, in *Entire Net-Work Client Installation and Administration Guide* for more information).

The syntax of the AdaSetSaf API function is:

```
AdaSetSaf(PADASAF_INFO adasaf_info")
```

Replace *adasaf_info* with the name of a data structure that provides information in the following format:

```
CE_CHAR  cUserID[8];          /* UserID          */
CE_CHAR  cPassword[8];       /* Password        */
CE_CHAR  cNewPassword[8];    /* NewPassword     */
```

In the following example, the data structure "mysaf" is used to provide appropriate external security interface access information to secured Adabas resources via ADASAF:

```
AdaSetSaf(PADASAF_INFO mysaf
```


5 Entire Net-Work Service Function Utility (wcpadmin)

- Syntax 24
- Special Character Use 28
- Creating a Kernel Using wcpadmin 28
- General Examples 29

The Entire Net-Work Service Function Utility allows you to perform some of the Entire Net-Work service functions in batch mode.

Syntax

The syntax of the wcpadmin function is:

```
wcpadmin {kernel-name| *} [function]
```

Specify one of the service functions (*function*) using the syntax listed in the following table.



Note: Please review the information on special character use with the wcpadmin function on Windows and UNIX systems. This is described in [Special Character Use](#), elsewhere in this section.

Function Name	Syntax	This function...
*	wcpadmin *	Displays the Entire Net-Work configuration on the machine.
add	wcpadmin kernel-name add node-name url	Allows you to add a connection to the Kernel. The name of the Kernel, the node and the node's associated URL must be specified for this function.
addconnection	wcpadmin kernel-name addconnection {online offline} tgt-kernelurl	Allows you to add a connection URL for a target Kernel (<i>tgt-kernel</i>) to the selected Kernel (<i>kernel-name</i>). You can specify whether the connection is online or offline. For example, wcpadmin krnl addconnection online test4 tcpip://localhost:8000 adds an online connection for Kernel "test4" to Kernel "krnl".
addlisten	wcpadmin kernel-name addlisten {server client classic smh} url	Allows you to add a "listen" to a Kernel definition. Using this function, you can specify the URL (and port) for various incoming connections of the Kernel. You must specify the kind of connection using these keywords: <ul style="list-style-type: none"> ■ Use "server" to define the URL and port number of an incoming Kernel connection. For example, wcpadmin krnl addlisten server tcpip://localhost:8000. ■ Use "client" to define the URL and port number of an incoming client connection. For example, wcpadmin krnl addlisten client tcpip://localhost:8080. ■ Use "classic" to define the URL and port number of an incoming RDA connection. For example,

Function Name	Syntax	This function...
		<p>wcpadmin krnl addlisten classic rda://localhost:8088.</p> <ul style="list-style-type: none"> ■ Use "smh" to define the URL and port number of an incoming System Management Hub (SMH) connection. For example, wcpadmin krnl addlisten smh tcpip://localhost:8050.
connect	wcpadmin <i>kernel-name</i> connect <i>node-name</i>	Allows you to connect a node to the Kernel. The name of the Kernel and the node name must be specified for this function.
create	wcpadmin <i>kernel-name</i> create	<p>Creates a new Kernel definition that is not fully complete. The Kernel configuration file is created and an entry is added to the service. To complete the Kernel definition, you must also call the following subsequent functions:</p> <ol style="list-style-type: none"> 1. Set Kernel parameter values using the setparm function of wcpadmin utility. 2. Add "listen" connections for servers, clients, RDA, and SMH connections to the Kernel using the addlisten function of the wcpadmin utility. 3. Add connections for the Kernel using the addconnection function of the wcpadmin utility.
delete	wcpadmin <i>kernel-name</i> delete	Deletes the Kernel from the service and deletes the Kernel configuration file. This function only works if the Kernel is offline.
disconnect	wcpadmin <i>kernel-name</i> disconnect <i>node-name</i>	Allows you to disconnect a node from the Kernel. The name of the Kernel and the node name must be specified for this function.
getparm	wcpadmin <i>kernel-name</i> getparm <i>parm-name</i>	<p>Allows you to view the value of a specific parameter of the selected Kernel. Possible parameter names that can be specified (<i>parm-name</i>) are:</p> <p>ACCEPTED_DBIDS, ACCEPTED_HOSTS, ACCEPTED_KERNELS, ADABAS_TIMEOUT, AUTOSTART, AUTOSTOP, CHECK_CXT_INTERVAL, CHECK_DBS_INTERVAL, DATE_STAMP, GATEWAY_THREADS, LNKTRACE, LOGSIZE, MAX_CLIENTS, MAX_CPU_THRESHOLD, NODEID, PING_DB_INTERVAL, REJECTED_DBIDS, REJECTED_HOSTS, REJECTED_KERNELS, RELAY_TRAFFIC, SAGXTSDSHOST, SAGXTSDSPORT, STATISTICS_DETAILS, STATISTICS_INTERVAL, TCP_AF, TIMER_TIMEOUT, WCPPARTITION,</p>

Function Name	Syntax	This function...
		WCPTRACE, XTSLOGDIR, XTSTRACE. These parameters are described in <i>Managing Kernels</i> , in the <i>Entire Net-Work Server LUW Installation and Administration Guide</i> .
setparm	wcpadmin kernel-name setparm parm-name parm-value	Allows you to set the value of a specific parameter of the selected Kernel. Possible parameter names that can be specified (<i>parm-name</i>) are: ACCEPTED_DBIDS, ACCEPTED_HOSTS, ACCEPTED_KERNELS, ADABAS_TIMEOUT, AUTOSTART , AUTOSTOP, CHECK_CXT_INTERVAL, CHECK_DBS_INTERVAL, DATE_STAMP, GATEWAY_THREADS, LNKTRACE, LOGSIZE, MAX_CLIENTS, MAX_CPU_THRESHOLD, NODEID, PING_DB_INTERVAL, REJECTED_DBIDS, REJECTED_HOSTS, REJECTED_KERNELS, RELAY_TRAFFIC, SAGXTSDSHOST, SAGXTSDSPORT, STATISTICS_DETAILS, STATISTICS_INTERVAL, TCP_AF, TIMER_TIMEOUT, WCPPARTITION, WCPTRACE, XTSLOGDIR, XTSTRACE. These parameters are described in <i>Managing Kernels</i> , in the <i>Entire Net-Work Server LUW Installation and Administration Guide</i> .
showclients	wcpadmin kernel-name showclients [D R A [<i>id</i>]]	Allows you to view the clients on associated with the Kernel. The name of the Kernel must be supplied for this function. Optionally, supply one of the following: <ul style="list-style-type: none"> ■ D: Specify "D" to view the direct clients of the Kernel. ■ R: Specify "R"to view the relay clients of the Kernel. ■ A: Specify "A" to view the Adabas clients of the Kernel. When you specify "A", you can also specify a client ID (<i>id</i>) to view the details of a specific Adabas client.
showconnections	wcpadmin kernel-name showconnections	Allows you to view the existing connections of the Kernel and the connection status. The name of the Kernel must be specified for this function.
showdb	wcpadmin kernel-name showdb {* <i>dbid</i> }	Allows you to view all the databases on the Kernel or details about a specific database on the Kernel. The name of the Kernel must be supplied for this function. In addition, either an asterisk (*) or a specific database ID (<i>dbid</i>) must be specified.

Function Name	Syntax	This function...
showhosts	wcpadmin <i>kernel-name</i> showhosts [<i>hostname</i>]	Allows you to view the host names of known to the Kernel or statistics about a specific host. The name of the Kernel must be supplied for this function. Optionally, you can specify a specific host name (<i>hostname</i>) to view statistics about a specific host.
showlistens	wcpadmin <i>kernel-name</i> showlistens	Allows you to view the listen ports for the Kernel. The name of the Kernel must be supplied for this function. This function only shows the active listens, not the listens listed in the configuration file. So this function only works if the Kernel is active.
shownodes	wcpadmin <i>kernel-name</i> shownodes [I O]	Allows you to view the Kernel's connections to nodes. The name of the Kernel must be supplied for this function. Optionally, specify "I" to view only incoming connections or "O" to view only outgoing connections. If you specify neither "I" nor "O", all connections are shown.
shownodestats	wcpadmin <i>kernel-name</i> shownodestats <i>node-name</i> {I O}	Allows you to view the Kernel connection statistics for a specific node. The name of the Kernel and the name of the node must be supplied for this function. In addition, you must specify "I" to view the statistics for the incoming connections of a node or "O" to view the statistics for the outgoing connections of a node.
showparms	wcpadmin <i>kernel-name</i> showparms	Allows you to view all parameters in a Kernel configuration file.
showstats	wcpadmin <i>kernel-name</i> showstats	Allows you to view statistics for the Kernel. The name of the Kernel must be supplied for this function.
showstatus	wcpadmin { <i>kernel-name</i> *} showstatus	Allows you to view the status of a specific Kernel or of all Kernels. Specify either: <ul style="list-style-type: none"> ■ The name of a specific Kernel to review the status of that specific Kernel. ■ An asterisk (*) to review the status of all Kernels.
shutdown	wcpadmin { <i>kernel-name</i> *} shutdown	Shuts down (stops) a specific Kernel or all Kernels. Specify either: <ul style="list-style-type: none"> ■ The name of a specific Kernel to shut down. ■ An asterisk (*) to shut down all Kernels.
start	wcpadmin { <i>kernel-name</i> *} start	Starts a specific Kernel or all Kernels. Specify either: <ul style="list-style-type: none"> ■ The name of a specific Kernel to start. ■ An asterisk (*) to start all Kernels.

Function Name	Syntax	This function...
stopclients	wcpadmin <i>kernel-name</i> stopclients {D R A [<i>id</i>]}	Stops the clients of a Kernel based on type (direct clients, relay clients or Adabas clients). The name of the Kernel must be supplied for this function. In addition, supply one of the following: <ul style="list-style-type: none"> ■ D: Specify "D" to stop the direct clients of the Kernel. ■ R: Specify "R" to stop the relay clients of the Kernel. ■ A: Specify "A" to stop the Adabas clients of the Kernel. When you specify "A", you can optionally specify a client ID (<i>id</i>) to stop a specific Adabas client.
stophosts	wcpadmin <i>kernel-name</i> stophosts <i>hostname</i>	Stops all clients of a Kernel coming from a specific host. The name of the Kernel and the name of the host must be supplied for this function.

Special Character Use

In Windows environments, the ampersand [&], pipe [|] and parentheses [()] are special characters that must be preceded by the escape character [^] or passed within quotation marks ["] when they are used in arguments of the wcpadmin function.

In UNIX environments, characters that have special meaning to the shell (such as the greater than [>], less than [<], asterisk [*], question mark [?], pipe [|], and ampersand [&] symbols) are called *metacharacters*. The list of metacharacters varies depending on the UNIX shell in use. When used in arguments of the wcpadmin function, metacharacters should be passed within single quotes or preceded by a forward slash [/]. For example, wcpadmin '*' start or wcpadmin /* start are examples of the correct use of metacharacters in wcpadmin UNIX environment processing.

Creating a Kernel Using wcpadmin

The wcpadmin create function does not create a fully functional Kernel. After kernel creation, other functions must be invoked as well.

➤ To create a fully functional Kernel using the wcpadmin utility, complete the following steps:

- 1 Create the initial Kernel definition using the wcpadmin create function. The Kernel configuration file is created and an entry is added to the service. The following example creates a Kernel definition for the Kernel named "test1".

```
wcpadmin test1 create
```

- Specify Kernel parameter values using the `setparam` function of `wcpadmin` utility. The following examples set Kernel parameters for the "test1" Kernel.

```
wcpadmin test1 setparam XTSTRACEX 0xffffe
wcpadmin test1 setparam LNKTRACEX 0xff
wcpadmin test1 setparam NODEID 1000
wcpadmin test1 setparam MAX_CLIENTS 1000
```

- Add "listen" connections for servers, clients, RDA, and SMH connections to the Kernel using the `addlisten` function of the `wcpadmin` utility. The following examples add server, client, and SMH connections for the "test1" Kernel.

```
wcpadmin test1 addlisten server tcpip://localhost:8000
wcpadmin test1 addlisten client tcpip://localhost:8080
wcpadmin test1 addlisten smh tcpip://localhost:8088
```

- Add connections for the Kernel using the `addconnection` function of the `wcpadmin` utility. The following examples add connections for the "test1" Kernel to "test2" and "test3".

```
wcpadmin test1 addconnection online test2 tcpip://localhost:3000?MANUAL=NO
wcpadmin test1 addconnection online test3 tcpip://localhost:4000?MANUAL=NO
```

Only when you have completed all of these steps have you created a fully functional Kernel definition.

General Examples

Examples for each of the `wcpadmin` functions are given in the following table.

Function Name	Example	Sample Output
*	wcpadmin *	<pre>Software AG Entire Net-Work Server, Copyright ↵ ©) 1997-2011 by Software AG SERVICE=USAXXX2 SAGXTSDSHOST=localhost SAGXTSDSPORT=12731 Kernels Available ===== FRIED XXXXX SECOND V733VIS</pre>

Function Name	Example	Sample Output
		===== WCPadmin exiting ...
add	wcpadmin XXXXX add FRIED TCPIP://localhost:8000	Software AG Entire Net-Work Server, Copyright © ©) 1997-2011 by Software AG Add Connection to FRIED on Kernel=XXXXX WCPadmin exiting ...
connect	wcpadmin XXXXX connect SECOND	Software AG Entire Net-Work Server, Copyright © ©) 1997-2011 by Software AG Connect Node=SECOND on Kernel=XXXXX WCP0628I Connect procedure has been initiated WCPadmin exiting ...
disconnect	wcpadmin XXXXX disconnect SECOND	Software AG Entire Net-Work Server, Copyright © ©) 1997-2011 by Software AG Disconnect Node=SECOND on Kernel=XXXXX WCP0629I Disconnect procedure has been initiated WCPadmin exiting ...
showclients	wcpadmin XXXXX showclients	Software AG Entire Net-Work Server, Copyright © ©) 1997-2011 by Software AG Show number of clients on Kernel=XXXXX Direct Clients 2 Adabas Contexts 2 Relay Clients 0 WCPadmin exiting ...
	wcpadmin XXXXX showclients D	Software AG Entire Net-Work Server, Copyright © ©) 1997-2011 by Software AG Direct Clients ClientId Host Port Dbid Elapsed Time 21 usaxxx2.YYY.ww.zzz 49155 1 0h:1m:3s WCPadmin exiting ...
	wcpadmin XXXXX showclients A	Software AG Entire Net-Work Server, Copyright © ©) 1997-2011 by Software AG Adabas Contexts ContextId Dbid Host Port Elapsed Time 7 1 usaxxx2.YYY.WW.ZZZ 49155 0h:1m:15s WCPadmin exiting ...

Function Name	Example	Sample Output
	wcpadmin XXXXX showclients A 7	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG Adabas context Dbid 1 ClientId 21 NodeName usaxxx2 User usaxxx Pid 0x86868cc4 Host usaxxx2.YYY.WW.ZZZ Port 49155 Requests 16 Replies 15 Bytes In 1468 Bytes Out 1380 Elapsed Time 0h:2m:17s WCPadmin exiting ...
showconnections	wcpadmin XXXXX showconnections	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG Show connections for Kernel=XXXXX TargNode Type Protocol Port A/M Status FRIED E-business TCPIP 8000 Auto Connection defined TEST E-business TCPIP 7000 Manual Connection defined SECOND E-business TCPIP 9000 Manual Disconnected RDHP1 Classic RDA 0 Manual Connected WCPadmin exiting ...
showdb	wcpadmin XXXXX showdb *	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG Show DBIDs for Kernel=XXXXX DBID Node Type Status 168 RDHP1 Classic Active 159 RDHP1 Classic Active 158 RDHP1 Classic Active 140 RDHP1 Classic Active 86 RDHP1 Classic Active 59 RDHP1 Classic Active 1 XXXXX Local Active WCPadmin exiting ...

Function Name	Example	Sample Output
	wcpadmin XXXXX showdb 168	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG Show DBID=168 on Kernel=XXXXX XXXXX Current Since startup Bytes in Requests 0 0 Bytes in Replies 0 0 Number of Requests 0 0 Number of Replies 0 0 Connections 1 WCPadmin exiting ...
showhosts	wcpadmin XXXXX showhosts	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG Host Connections Disconnections Elapsed Time usamih2.YYY.WW.ZZZ 10 10 0h:46m:1s WCPadmin exiting ...
	wcpadmin XXXXX showhosts usaxxx2.YYY.WW.ZZZ	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG Host usamih2.YYY.WW.ZZZ Count 0 Connections 10 Disconnections 10 Requests 122 Replies 102 Bytes In 11225 Bytes Out 9384 Elapsed Time 0h:46m:20s WCPadmin exiting ...
showlistens	wcpadmin XXXXX showlistens	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG Show listens for Kernel=XXXXX Name Protocol Port Status Client Access TCPIP 49155 Running E-Business Access TCPIP 8000 Running SMH Server TCPIP 49154 Running WCPadmin exiting ...

Function Name	Example	Sample Output																														
shownodes	wcpadmin XXXXX shownodes	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG Show nodes on Kernel=XXXXX <table border="1"> <thead> <tr> <th>Target</th> <th>Type</th> <th>0/I</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td>SECOND</td> <td>E-Business</td> <td>I</td> <td>Active</td> </tr> <tr> <td>FRIED</td> <td>E-Business</td> <td>0</td> <td>Inactive</td> </tr> <tr> <td>TEST</td> <td>E-Business</td> <td>0</td> <td>Inactive</td> </tr> <tr> <td>SECOND</td> <td>E-Business</td> <td>0</td> <td>Inactive</td> </tr> <tr> <td>RDHP1</td> <td>Classic</td> <td>0</td> <td>Inactive</td> </tr> </tbody> </table> WCPadmin exiting ...	Target	Type	0/I	Status	SECOND	E-Business	I	Active	FRIED	E-Business	0	Inactive	TEST	E-Business	0	Inactive	SECOND	E-Business	0	Inactive	RDHP1	Classic	0	Inactive						
Target	Type	0/I	Status																													
SECOND	E-Business	I	Active																													
FRIED	E-Business	0	Inactive																													
TEST	E-Business	0	Inactive																													
SECOND	E-Business	0	Inactive																													
RDHP1	Classic	0	Inactive																													
shownodestats	wcpadmin XXXXX shownodestats RDHP1 0	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG XXXXX Current <table border="1"> <thead> <tr> <th colspan="2">Since Startup</th> <th></th> </tr> </thead> <tbody> <tr> <td>Bytes received</td> <td>0</td> <td>128</td> </tr> <tr> <td>Bytes sent</td> <td>0</td> <td>128</td> </tr> <tr> <td>Requests received</td> <td>0</td> <td>3</td> </tr> <tr> <td>Requests sent</td> <td>0</td> <td>2</td> </tr> <tr> <td>Incoming connections</td> <td>0</td> <td>N/A</td> </tr> <tr> <td>Outgoing connections</td> <td>1</td> <td>N/A</td> </tr> <tr> <td>Admin msgs received</td> <td>0</td> <td>3</td> </tr> <tr> <td>Admin msgs sent</td> <td>0</td> <td>2</td> </tr> <tr> <td>Errors</td> <td>0</td> <td>0</td> </tr> </tbody> </table> WCPadmin exiting ...	Since Startup			Bytes received	0	128	Bytes sent	0	128	Requests received	0	3	Requests sent	0	2	Incoming connections	0	N/A	Outgoing connections	1	N/A	Admin msgs received	0	3	Admin msgs sent	0	2	Errors	0	0
Since Startup																																
Bytes received	0	128																														
Bytes sent	0	128																														
Requests received	0	3																														
Requests sent	0	2																														
Incoming connections	0	N/A																														
Outgoing connections	1	N/A																														
Admin msgs received	0	3																														
Admin msgs sent	0	2																														
Errors	0	0																														
showstats	wcpadmin XXXXX showstats	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG Get statistics for Kernel=XXXXX <table border="1"> <thead> <tr> <th colspan="2">Statistics for Kernel XXXXX</th> </tr> </thead> <tbody> <tr> <td>Accumulated Node Count</td> <td>4</td> </tr> <tr> <td>Databases</td> <td>1</td> </tr> <tr> <td>Adabas Calls Processed</td> <td>102</td> </tr> <tr> <td>Classic Nodes</td> <td>1</td> </tr> <tr> <td>Classic Messages</td> <td>0</td> </tr> <tr> <td>e-Business Messages</td> <td>122</td> </tr> <tr> <td>e-Business Contexts</td> <td>10</td> </tr> <tr> <td>Active Clients</td> <td>0</td> </tr> <tr> <td>Relay Messages</td> <td>0</td> </tr> <tr> <td>Memory Allocations</td> <td>64</td> </tr> </tbody> </table>	Statistics for Kernel XXXXX		Accumulated Node Count	4	Databases	1	Adabas Calls Processed	102	Classic Nodes	1	Classic Messages	0	e-Business Messages	122	e-Business Contexts	10	Active Clients	0	Relay Messages	0	Memory Allocations	64								
Statistics for Kernel XXXXX																																
Accumulated Node Count	4																															
Databases	1																															
Adabas Calls Processed	102																															
Classic Nodes	1																															
Classic Messages	0																															
e-Business Messages	122																															
e-Business Contexts	10																															
Active Clients	0																															
Relay Messages	0																															
Memory Allocations	64																															

Function Name	Example	Sample Output
		Memory Frees 20 WCPadmin exiting ...
showstatus	wcpadmin XXXXX showstatus	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG XXXXX Active WCPadmin exiting ...
	wcpadmin * showstatus	Software AG Entire Net-Work Server, Copyright ©) 1997-2010 by Software AG Kernels Status FRIED Inactive XXXXX Active SECOND Inactive V733VIS Inactive ===== WCPadmin exiting ...
shutdown	wcpadmin SECOND shutdown	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG Shutdown Kernel=SECOND for WCP Service=USAXXX2 Shutdown Kernel=SECOND Service=USAXXX2 has been submitted WCPadmin exiting ...
start	wcpadmin SECOND start	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG Start Kernel=SECOND for WCP Service=USAXXX2 WCPadmin exiting ...
stopclients	wcpadmin XXXXX showclients D	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG Direct Clients ClientId Host Port Dbid Elapsed Time 47 usamih2.XXX.wv.zzz 49155 1 0h:0m:46s WCPadmin exiting ...
	wcpadmin XXXXX stopclients D 47	Software AG Entire Net-Work Server, Copyright ©) 1997-2011 by Software AG Stop direct client=47 WCPadmin exiting ...

Function Name	Example	Sample Output
stophosts	wcpadmin XXXXX stophosts usaxxx2.XXX.WW.ZZZ	Software AG Entire Net-Work Server, Copyright ↵ ©) 1997-2011 by Software AG WCPadmin exiting ...

6 Entire Net-Work Utility Functions for the Directory Server (checkadi and setadi)

- The checkadi Utility 38
- The setadi Utility 39

Two Entire Net-Work utility functions with focus on the Adabas Directory Server availability and settings are provided for you to use in batch mode:

- Use the checkadi utility function to check for a Directory Server.
- Use the setadi utility function to set Directory Server access parameters for Entire Net-Work and Entire Net-Work Client.

This chapter describes both of these utilities.

The checkadi Utility

Use the checkadi utility to check for the existence of a Directory Server. The syntax of the checkadi function is:

```
checkadi [host=host-name] [[port=port-value]
```

Use the host or port arguments to check for the existence of a Directory Server on a specific host or port number. You can use both the host and port arguments to more specifically check for a Directory Server on a specific host and port.

Example 1

In the following example, a check is run for a Directory Server on the usaxxx2 host at port 12731:

```
checkadi host=usaxxx2 port=12731
```

The following sample output from such a check might appear:

```
Software AG Entire Net-Work, Copyright ©) 1997-2011 by Software AG
USAGE: checkadi [host=hostname] [port]=portvalue]
argv[1] host=usaxxx2
Check host=usaxxx2
argv[2] port=12731
Check port=12731
Port was set to 12731
Check Host=usaxxx2
Check Port=12731
Server is Active; check if this is a Directory Server
Select Data from Directory Server successful
Bytes ready to read=309
Response=0x010x33 0x760x310x090x720x650x730x700x6f0x6e0x730x650x090x09
Expected=          0x760x310x090x720x650x730x700x6f0x6e0x730x650x090x09
Directory Server is Active
Checkadi ending ...
```

Example 2

In the following example, a check is run to determine where a Directory Server exists:

```
checkadi
```

The following sample output from such a check might appear:

```

Software AG Entire Net-Work, Copyright ©) 1997-2011 by Software AG
USAGE: checkadi [host=hostname] [port]=portvalue]
Resolve SAGXTSDSHOST
Failure Resolve Host Name; use localhost
Port was not set, so we will use the default port=12731
Check Host=usaxxx2.YYY.ww.zzz
Check Port=12731
Server is Active; check if this is a Directory Server
Select Data from Directory Server successful
Bytes ready to read=309
Response=0x010x33 0x760x310x090x720x650x730x700x6f0x6e0x730x650x090x09
Expected=          0x760x310x090x720x650x730x700x6f0x6e0x730x650x090x09
Directory Server is Active
Checkadi ending ...

```

The setadi Utility

Use the setadi utility to set Directory Server access parameters for Entire Net-Work and Entire Net-Work Client. The syntax of the setadi function is:

```
setadi {WCP|WCL} host=host-name port=port-value [XTSTRACE={value|65534}]
```

You must specify either "WCP" (to set the access parameters for Entire Net-Work) or "WCL" (to set access parameters for Entire Net-Work Client). You should also specify the host name and port number parameters. The XTSTRACE parameter is optional; if you do not specify it, a default value of "65534" is used.



Note: While you can use setadi to change the Directory Server used, the changes only affect the configuration of the services and agents. It will not change the Directory Server assigned to any existing Kernels.

Example 1

In the following example, help for setadi is displayed, but no access parameters are set.

```
setadi
```

The following sample output from such a setadi request might appear:

```
Software AG Entire Net-Work, Copyright ©) 1997-2011 by Software AG
Usage: setadi <options...>
The following options are supported:
WCP|WCL
HOST=host name
PORT=port value
XTSTRACE=value (65534)

WCP|WCL - the user selects which product to set, WCP or WCL
```

Example 2

In the following example, an Entire Net-Work entry for host "localhost" at port "12731" is defined. The default XTSTRACE value of "65534" is used.

```
setadi WCP host=localhost port=12731
```

The following sample output from such a setadi request might appear:

```
Software AG Entire Net-Work, Copyright ©) 1997-2010 by Software AG
argv[2] host=localhost
argv[3] port=12731
Check Host=localhost
Check Port=12731
Server is Active; check if this is a Directory Server
Select Data from Directory Server successful
Bytes ready to read=309
Response=0x010x33 0x760x310x090x720x650x730x700x6f0x6e0x730x650x090x09
Expected=          0x760x310x090x720x650x730x700x6f0x6e0x730x650x090x09
Directory Server is Active
CODEPATH=C:\Program Files\Software AG\Entire Net-Work Server\v74\
DATAPATH=C:\Documents and Settings\All Users\Application Data\Software AG\Entire ↵
Net-Work Server\
Changing C:\Documents and Settings\All Users\Application Data\Software AG\Entire ↵
Net-Work Server\service74.config
Changing C:\Documents and Settings\All Users\Application Data\Software AG\Entire ↵
Net-Work Server\agents\xts.config
Configuration file change successful
Setadi exiting ...
```

Index

A

accessing
 System Management Hub, 2
AdaSetParameter API function, 19
AdaSetSaf API function, 21
AdaSetTimeout API function, 20
AdaSetTrace API function, 20
API functions, 19

C

checkadi utility, 38
commands
 for SMH management in UNIX, 2

D

Directory Server
 utility functions for, 37

E

Entire Net-Work
 SMH administration, v
 System Management Hub, 1

H

help
 System Management Hub, 4

L

logging in
 SMH, 3

P

port numbers, 5

R

Refresh button
 System Management Hub, 4

S

setadi utility, 39
shutting down
 System Management Hub, 3
SMH (see System Management Hub)
system administration
 System Management Hub, 1
 UNIX commands for managing SMH, 2
System Management Hub
 about, 1
 accessing, 2
 Entire Net-Work administration, v
 getting help, 4
 logging in, 3
 management commands for UNIX, 2
 Refresh button, 4
 shutting down, 3

U

UNIX
 SMH management commands, 2
utility functions
 checkadi, 38
 for the Directory Server, 37
 setadi, 39

