

# **Entire Net-Work**

## **Administration**

Version 7.9.2

October 2022

This document applies to Entire Net-Work Version 7.9.2 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 © 2022 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-792-20221010**

## Table of Contents

Preface .....	v
1 About this Documentation .....	1
Document Conventions .....	2
Online Information and Support .....	2
Data Protection .....	3
2 About the Adabas Manager .....	5
3 Port Number Reference .....	7
Port Overview and General Assignments .....	8
Changing the Adabas Directory Server Port Number .....	9
4 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
5 Entire Net-Work Configuration Parameters .....	15
6 Entire Net-Work Service Function Utility (wcpadmin) .....	17
Syntax .....	18
Special Character Use .....	22
Creating a Kernel Using wcpadmin .....	22
General Examples .....	23
7 Entire Net-Work Utility Functions for the Directory Server (checkadi and setadi) .....	31
The checkadi Utility .....	32
The setadi Utility .....	33
Index .....	35



---

## Preface

---

This document describes how to perform the system administration tasks necessary for Entire Net-Work operations using the Adabas Manager (AMN).

It is intended for system administrators in your enterprise.

This document is organized as follows:

<i>About the Adabas Manager</i>	Introduces you to the Adabas Manager 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.
<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 this Documentation

---

■ Document Conventions .....	2
■ Online Information and Support .....	2
■ Data Protection .....	3

## Document Conventions

---

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

## Online Information and Support

---

### Product Documentation

You can find the product documentation on our documentation website at <https://documentation.softwareag.com>.

In addition, you can also access the cloud product documentation via <https://www.software-ag.cloud>. Navigate to the desired product and then, depending on your solution, go to “Developer Center”, “User Center” or “Documentation”.

### Product Training

You can find helpful product training material on our Learning Portal at <https://knowledge.softwareag.com>.



## Tech Community

You can collaborate with Software AG experts on our Tech Community website at <https://tech-community.softwareag.com>. From here you can, for example:

- Browse through our vast knowledge base.
- Ask questions and find answers in our discussion forums.
- Get the latest Software AG news and announcements.
- Explore our communities.
- Go to our public GitHub and Docker repositories at <https://github.com/softwareag> and <https://hub.docker.com/publishers/softwareag> and discover additional Software AG resources.

## Product Support

Support for Software AG products is provided to licensed customers via our Empower Portal at <https://empower.softwareag.com>. Many services on this portal require that you have an account. If you do not yet have one, you can request it at <https://empower.softwareag.com/register>. Once you have an account, you can, for example:

- Download products, updates and fixes.
- Search the Knowledge Center for technical information and tips.
- Subscribe to early warnings and critical alerts.
- Open and update support incidents.
- Add product feature requests.

## Data Protection

---

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



## 2 About the Adabas Manager

---

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

Before you start using the Adabas Manager, you must set up an administrative user for the product. To do so, consult the *Using Adabas Manager* section of the *Adabas Manager* documentation, available on Empower.



# 3

## Port Number Reference

---

■ Port Overview and General Assignments .....	8
■ Changing the Adabas Directory Server Port Number .....	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 Communicator	One port is needed.	4980
Adabas Directory Server	One port is needed for Entire Net-Work requests to the Directory Server	4952 (IANA port)  <b>Note:</b> If older versions of Entire Net-Work (older than 7.3) are in use, this port number may need to be changed to 12731.
Adabas Manager	One port is needed for Adabas Manager 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 connections (Entire Net-Work 7 or later)	dynamically assigned
	A port is needed for Kernel access via RDA connections (Entire Net-Work 2)	7869
	A port is needed for Adabas Manager 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 Adabas Manager by changing the SAGXTSDSPORT parameter. See the *Adabas Manager* documentation for details.
- 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 Adabas Manager by changing the SAGXTSDSPORT parameter. See the *Adabas Manager* documentation for details.

- 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*.




# 4

## 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 Adabas Manager (AMN), 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.



# 5

## Entire Net-Work Configuration Parameters

---

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 Adabas Manager (AMN) functions. For more information on this, refer to the section *Entire Net-Work Administration* in the *Adabas Manager* documentation.



## 6 Entire Net-Work Service Function Utility (wcpadmin)

---

■ Syntax .....	18
■ Special Character Use .....	22
■ Creating a Kernel Using wcpadmin .....	22
■ General Examples .....	23

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 Linux 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	<p>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:</p> <ul style="list-style-type: none"> <li>■ Use "server" to define the URL and port number of an incoming Kernel connection. For example, wcpadmin krnl addlisten server tcpip://localhost:8000.</li> <li>■ Use "client" to define the URL and port number of an incoming client connection. For example, wcpadmin krnl addlisten client tcpip://localhost:8080.</li> <li>■ Use "classic" to define the URL and port number of an incoming RDA connection. For example,</li> </ul>



Function Name	Syntax	This function...
		<p>wcpadmin krnl addlisten classic rda://localhost:8088.</p> <p>■ 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.</p>
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"> <li>1. Set Kernel parameter values using the setparm function of wcpadmin utility.</li> <li>2. Add "listen" connections for servers, clients, RDA, and SMH connections to the Kernel using the addlisten function of the wcpadmin utility.</li> <li>3. Add connections for the Kernel using the addconnection function of the wcpadmin utility.</li> </ol>
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...
		WCPTTRACE, 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, WCPTTRACE, 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:  ■ 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"> <li>■ The name of a specific Kernel to review the status of that specific Kernel.</li> <li>■ An asterisk (*) to review the status of all Kernels.</li> </ul>
shutdown	wcpadmin { <i>kernel-name</i>  *} shutdown	Shuts down (stops) a specific Kernel or all Kernels. Specify either: <ul style="list-style-type: none"> <li>■ The name of a specific Kernel to shut down.</li> <li>■ An asterisk (*) to shut down all Kernels.</li> </ul>
start	wcpadmin { <i>kernel-name</i>  *} start	Starts a specific Kernel or all Kernels. Specify either: <ul style="list-style-type: none"> <li>■ The name of a specific Kernel to start.</li> <li>■ An asterisk (*) to start all Kernels.</li> </ul>

Function Name	Syntax	This function...
stopclients	<code>wcpadmin kernel-name stopclients {D R A [<i>id</i>]}</code>	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"> <li>■ D: Specify "D" to stop the direct clients of the Kernel.</li> <li>■ R: Specify "R" to stop the relay clients of the Kernel.</li> <li>■ 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.</li> </ul>
stophosts	<code>wcpadmin kernel-name stophosts hostname</code>	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 Linux 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 Linux 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 Linux 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 0xfffe
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
		<pre>===== WCPadmin exiting ...</pre>
add	wcpadmin XXXXX add FRIED TCPIP://localhost:8000	<pre>Software AG Entire Net-Work Server, Copyright ↵ ©) 1997-2011 by Software AG Add Connection to FRIED on Kernel=XXXXX WCPadmin exiting ...</pre>
connect	wcpadmin XXXXX connect SECOND	<pre>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 ...</pre>
disconnect	wcpadmin XXXXX disconnect SECOND	<pre>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 ...</pre>
showclients	wcpadmin XXXXX showclients	<pre>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 ...</pre>
	wcpadmin XXXXX showclients D	<pre>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 ...</pre>
	wcpadmin XXXXX showclients A	<pre>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 ...</pre>

Function Name	Example	Sample Output
	wcpadmin XXXXX showclients A 7	<pre> 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 ... </pre>
showconnections	wcpadmin XXXXX showconnections	<pre> 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 ... </pre>
showdb	wcpadmin XXXXX showdb *	<pre> 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 ... </pre>

Function Name	Example	Sample Output
	wcpadmin XXXXX showdb 168	<pre>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 ...</pre>
showhosts	wcpadmin XXXXX showhosts	<pre>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 ...</pre>
	wcpadmin XXXXX showhosts usaxxx2.YYY.WW.ZZZ	<pre>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 ...</pre>
showlistens	wcpadmin XXXXX showlistens	<pre>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 ...</pre>



Function Name	Example	Sample Output
shownodes	wcpadmin XXXXX shownodes	<pre> Software AG Entire Net-Work Server, Copyright ↵ ©) 1997-2011 by Software AG Show nodes on Kernel=XXXXX 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 WCPadmin exiting ... </pre>
shownodestats	wcpadmin XXXXX shownodestats RDHP1 0	<pre> Software AG Entire Net-Work Server, Copyright ↵ ©) 1997-2011 by Software AG XXXXX   Current ↵       Since Startup Bytes received                               128 ↵       0 Bytes sent                                   128 ↵       0 Requests received                           3 ↵       0 Requests sent                               2 ↵       0 Incoming connections                        N/A ↵       0 Outgoing connections                        N/A ↵       1 Admin msgs received                         3 ↵       0 Admin msgs sent                             2 ↵       0 Errors                                      0 ↵       0 WCPadmin exiting ... </pre>
showstats	wcpadmin XXXXX showstats	<pre> Software AG Entire Net-Work Server, Copyright ↵ ©) 1997-2011 by Software AG Get statistics for Kernel=XXXXX 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 </pre>

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 =====
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.ww.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 ...



# 7

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

---

■ The checkadi Utility .....	32
■ The setadi Utility .....	33

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 ©) 1997-2019 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 ©) 1997-2019 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 ©) 1997-2019 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 ©) 1997-2019 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

Adabas Manager  
  about, 5  
AMN (see Adabas Manager)

## C

checkadi utility, 32

## D

Directory Server  
  utility functions for, 31

## E

Entire Net-Work  
  Adabas Manager, 5  
  SMH administration, v

## P

port numbers, 7

## S

setadi utility, 33  
system administration  
  Adabas Manager, 5  
System Management Hub  
  Entire Net-Work administration, v

## U

utility functions  
  checkadi, 32  
  for the Directory Server, 31  
  setadi, 33

