

openUTM WinAdmin V6.4



Graphic Administration Interface for openUTM

1 Contents

1	Contents.....	2
2	Summary of contents.....	8
3	WinAdmin quick start.....	9
4	Installation.....	12
4.1	Installation under Windows systems.....	12
4.2	Installation under Unix/Linux systems.....	13
4.3	Installation in systems without graphical user interface.....	13
5	What's new.....	14
5.1	... in Version 6.4?.....	14
5.2	... in Version 6.3?.....	14
5.3	... in Version 6.2?.....	14
5.4	... in Version 6.1?.....	14
6	Incompatibilities.....	17
7	General.....	18
7.1	System platforms.....	18
7.2	Resources required.....	18
7.3	Programming language.....	18
7.4	Linking to UTM applications.....	18
7.5	Supported UTM versions.....	20
7.6	Language.....	20
8	Starting and stopping WinAdmin.....	21
9	Configuration database.....	22
9.1	Selecting and creating a configuration database.....	22
9.2	Backup copies.....	23
9.3	Troubleshooting.....	23
10	The program interface.....	24
11	Communication requests.....	25
11.1	List of communication requests.....	26
12	WinAdmin objects.....	28
12.1	WinAdmin users.....	29
12.1.1	Displaying the WinAdmin user list.....	29
12.1.2	Generating a new WinAdmin user.....	30
12.1.3	Deleting a WinAdmin user.....	30
12.2	Hosts.....	30
12.2.1	The "Specify Hostname/IP-Address" dialog.....	32
12.2.2	Displaying the host list.....	32

12.2.3	Generating a new host	32
12.2.4	Deleting a host	33
12.3	UTM applications	33
12.3.1	Displaying the application list	34
12.3.2	Generating a new application	35
12.3.3	Deleting an application	35
12.4	UPIC connections	36
12.4.1	Displaying the UPIC connection list	37
12.4.2	Creating a new UPIC connection	37
12.4.3	Deleting an UPIC connection	38
12.5	Collections	38
12.5.1	Displaying the collection list	39
12.5.2	Generating a new collection	40
12.5.3	Deleting a collection	40
12.5.4	Opening a collection	40
12.5.5	Closing a collection	41
13	The navigation window	42
13.1	Layout of the Application and Cluster subtrees	45
13.1.1	Adapting the layout (user-defined layout)	46
14	Object lists	47
14.1	Displaying object lists with WinAdmin objects	47
14.2	Displaying object lists with UTM objects	47
14.3	Layout of an object list window	48
14.4	Filtering list objects	50
14.4.1	Entering a filter text	50
14.4.2	Column filters	52
14.4.3	Display filters	53
14.5	The lower button bar	56
14.6	The property dialog in the object list	57
14.6.1	The 'Content' sheet	57
14.6.2	The 'Display Filters' sheet	58
14.6.3	The 'Columns' sheet	58
14.6.4	The 'Column Sets' sheet	58
14.7	User-defined layout	61
14.8	Sorting objects	61
14.9	Context menus for the list objects	62
14.10	Data refresh	63
14.11	Linked objects	63
14.12	Definition of partial lists	64
14.12.1	Adding a new partial list	65
14.12.2	Removing a partial list	65
15	Property dialogs	66
15.1	Displaying the property dialog for an object	66
15.2	Displaying multiple objects in a property dialog	66

15.3	Modifying object properties	67
16	Creating and deleting objects	68
17	Functions of the Application and Cluster menu	69
18	Object selections	75
19	Support for the UTM cluster concept	76
19.1	UTM clusters and node applications in the navigation tree	76
19.2	Cluster definition in WinAdmin	77
19.2.1	UPIC users in the UTM cluster	77
19.3	Deactivating cluster nodes	78
19.4	Context menu of a UTM cluster	79
19.5	Context menu of node applications in the UTM cluster	79
19.6	Properties sheet for UTM clusters and node applications	79
19.7	List of UTM cluster nodes	80
19.8	Object lists of a cluster	80
20	Support for UTM access control	81
20.1	Accessing UTM services – keys and locks	81
20.1.1	"Services"	82
20.1.2	"Users"	82
20.1.3	"Couplers"	82
20.2	The "new" concept – roles and access lists	83
20.3	Roles	84
20.3.1	Roles in object lists	86
20.3.2	Roles in property dialogs	86
20.4	The Security view	87
20.4.1	'Security Resources' subtree	87
20.4.2	'Objects' That May Be Privileged' subtree	88
20.4.3	'Objects' That May Be Protected' subtree	89
20.4.4	"Local Services" subtree	89
20.4.5	"Remote Services" subtree	90
20.4.6	'Queues' subtree	90
21	WinAdmin message destination - message collectors	91
21.1	Actions on the UTM side	91
21.2	Actions on the WinAdmin side	92
21.3	Properties of message collectors	92
21.4	Message collectors in a cluster	94
21.5	The grouped messages list	94
21.6	Tips and notes on configuration	95
22	Transaction-oriented administration jobs	97
22.1	Combining jobs in a transaction	97
22.2	Displaying open administration jobs	98
22.3	Terminating open administration jobs	98

23	Administering message queues.....	100
24	Administering printers and print jobs.....	104
24.1	Printer control Lterms.....	104
24.2	Administering printers.....	105
24.3	Administering print jobs.....	106
25	Service-controlled queues in WinAdmin.....	108
25.1	The queue list.....	108
25.2	Creating and deleting objects.....	110
25.3	Displaying a queue's message queue.....	110
25.4	Queues in the Security view.....	110
26	Statistics.....	111
26.1	Statistics window.....	111
26.1.1	Collector list.....	112
26.1.2	Statistics graph.....	115
26.2	Statistic collectors.....	116
26.2.1	Creating statistic collectors.....	116
26.2.2	Deleting statistic collectors.....	117
26.2.3	Statistic collectors in clusters.....	117
26.2.4	Context menu of statistic collectors.....	117
26.3	Threshold alert.....	118
26.4	Statistic values - saving, deleting, exporting.....	120
26.5	Statistic types supported.....	120
27	Security - users and authorizations.....	127
27.1	Passwords and overall privileges.....	127
27.2	Privileges for individual applications and object types.....	128
28	File transfer functions in WinAdmin.....	129
29	Other functions.....	135
29.1	Dead Letter Queue Functions.....	135
29.1.1	Message Queue Extensions.....	135
29.2	Command mode.....	136
29.3	Message Dump.....	138
29.4	Actions.....	139
29.4.1	Save Object Properties in File.....	141
29.5	Importing the BeanConnect container.properties file.....	143
29.6	Get File command.....	144
29.7	Send File command.....	145
29.8	Show File command.....	145
29.8.1	Selecting text.....	147
29.8.2	Copying text to the clipboard.....	147
29.8.3	Highlighting identical words.....	147
29.8.4	Marking text or lines.....	148
29.8.5	Jumping between markers.....	148

29.8.6	Jumping to a specific line	149
29.8.7	Deleting individual markers, the selection or the defined line range.....	149
29.8.8	Deleting all markers and the selection.....	149
29.8.9	Fast searches for text locations	149
29.8.10	Moving to the start/end of the text	151
29.8.11	Defining line ranges.....	152
29.8.12	Hiding a line range.....	152
29.8.13	Showing a hidden line range again	152
29.8.14	Selecting a line range.....	152
29.8.15	Selecting the character font	153
29.8.16	Increasing and decreasing the font size.....	153
29.8.17	Online help.....	153
29.8.18	Switching line wrapping on/off	153
29.8.19	Showing line and column numbers	154
29.8.20	Displaying the property dialog.....	154
29.8.21	Text commands.....	154
29.9	Copying UTM objects to the clipboard	156
29.10	The actions dialogs	156
29.11	Tooltips in lists and property dialogs	159
29.12	Refreshing data	161
29.13	Trace function.....	161
29.14	Specifying the WinAdmin user at the start.....	162
29.14.1	Creating a WinAdmin program link (Windows systems).....	163
30	Settings	164
30.1	The 'General' sheet	164
30.2	The 'File Transfer' sheet.....	166
30.3	The 'Logging / Traces' sheet.....	166
31	Online help.....	168
32	Appendix.....	169
32.1	The menu bar	169
32.1.1	File menu	169
32.1.2	View menu	170
32.1.3	Extras menu.....	170
32.1.4	Window menu.....	171
32.1.5	Help menu.....	171
32.2	The status bar.....	171
32.3	Complex filter masks.....	172
32.4	Displayed icons	173
32.4.1	Object icons	173
32.4.2	Other icons.....	176
32.5	Shortcuts.....	179
32.6	Graphical interface terminology.....	181
33	Index.....	188

2 Summary of contents

This document describes the functionality of Version V6.4 of the openUTM WinAdmin program.

The following chapter is intended for users in a hurry, who are already familiar with the installation and operation of Windows programs and want to get on with using WinAdmin as quickly as possible. The necessary steps are briefly described in this chapter.

Chapter 4 describes the installation of WinAdmin in detail.

Chapter 5 briefly lists the most important functions that have been added over and above V6.0.

Chapter 6 describes incompatibilities with WinAdmin V6.3.

This is followed by a general description of the program and the preconditions for the use of WinAdmin.

The capabilities and operation of the program are displayed in greater detail from chapter 0 onwards.

3 WinAdmin quick start

For users in a hurry, the chapter briefly summarizes the steps that are necessary to work with WinAdmin.

- Adapt the generation of the existing UTM application(s)
To administer UTM applications with WinAdmin, you must insert the following statements in the KDCDEF input:

PROGRAM KDCWADMI, COMP=ILCS (BS2000 system)

or

PROGRAM KDCWADMI, COMP=C (Unix, Linux and Windows systems)

TAC KDCWADMI, PROGRAM=KDCWADMI, CALL=BOTH, ADMIN=Y

The program unit is already supplied with openUTM as of Version V4.0 and need not be linked explicitly.

BCAMAPPL BCUPICR, T-PROT=RFC1006

**TPOOL LTERM=TPUPIC, NUMBER=10, PRONAM=*ANY, PTYPE=UPIC-R,
BCAMAPPL=BCUPICR**

USER WINADMIN, PASS=C'XYZ', PERMIT=ADMIN, RESTART=NO

Now start a KDCDEF generation run.

See Chapter "Linking to UTM applications" on page 18.

- Install WinAdmin on the WinAdmin PC
See Chapter "Installation" on page 12
- Start WinAdmin
Click the WinAdmin icon in the generated program group.
You must now set up a configuration database (only when you first start WinAdmin). You are advised to choose a Microsoft Access database (*.mdb). In the dialog box, simply enter the name of a non-existent file (e.g. 'ConfigDb') and click 'Open'.
- Generate a host object
This makes known to WinAdmin the host on which the UTM application that is to be administered is running (must be done once for each host)
Choose 'Hosts' in the File menu.
Choose 'New Host...' in the Edit menu.
Complete the fields:
Name: Freely definable host name (only has local significance in WinAdmin)
Netname: Name under which the host is known in the network (mandatory field)
Operating System: Please choose the correct system.
The other fields are optional.
Click OK to close the dialog box.

■ Generate an application object

This reports the UTM application that is to be administered to WinAdmin (must be done once for each application)

Choose 'Applications' in the File menu.

Choose 'New Application...' in the Edit menu.

Complete the fields:

Name: Freely definable name of the UTM application (only has local significance in WinAdmin)

Host: Select the host on which the application runs (object has already been entered)

Version: Select the UTM version(automatically updated on the first successful contact)

Port: Port for UPIC communications (in BS2000 system this is usually 102)

Selector Format: Selector name format for UPIC communications (in BS2000 system Transdata is ok)

Click OK to close the dialog box.

■ Create an UPIC connection

This allows WinAdmin to communicate with the UTM application that is to be administered (must be done once for each application)

Choose 'UPIC Connections' in the File menu.

Click the 'Add' button below the (still empty) connection list.

Complete the fields:

UTM Application: The UTM applications to which the connection is to be established (object has already been entered)

Bcamappl: BCUPICR (if above values are used)

User: WINADMIN (if above values are used)

Password: XYZ (if above values are used)

Click OK to close the dialog box.

■ Create collection

Choose 'New Collection...' in the File menu.

Find the file name (e.g. 'Collection') and click 'Save'.

Select the newly entered application from the list of 'Available Applications' and click 'Add'.

Click OK to close the dialog box.

■ Open collection

Choose 'Open Collection...' in the File menu.

In the list, double-click the application collection that you have just created (or alternatively, click the collection with the name '<All UTM Applications>').

■ Check that the application is available

In the navigation tree that is now displayed, choose 'Applications' followed by the host name to move to the application name and right-click it with the mouse. Choose 'Check Availability' in the context menu.

■ View the UTM objects

Expand the subtree under the application name in the navigation tree.

Double-click the required object type. A list of all objects of this type is displayed.

■ You can call the Online Help at any time by pressing F1.

- Tip

It is always a good idea to right-click when pointing at objects since there is usually a context menu for these objects.

4 Installation

Before installation, a Java JRE or JDK of version Java SE 6 Update 25 or higher must be installed on the destination computer. This can be downloaded free of charge from the Oracle Web site (<http://www.oracle.com/technetwork/java/javase/downloads/index.html>). If the destination computer has a 64-bit architecture and is running a 64-bit operating system then it is advisable to install a 64-bit version of Java. It is also possible to install multiple versions of Java on the destination computer, i.e. if you want, you can install a 64-bit version of Java alongside an existing 32-bit version. Depending on the employed Java version, WinAdmin runs as either a 64-bit application or a 32-bit application. The 64-bit version has the advantage of making it possible to use a larger address space. This can be of value in the case of applications with very large numbers of objects.

A WISE installation is available for Windows systems and an IzPack installation is available under Unix/Linux systems.

4.1 Installation under Windows systems

You should close all open applications and windows on your system before starting to install openUTM WinAdmin.

Unzip file *WinAdmin64A00.zip*

Call the file *WinAdmin.msi* to start installing WinAdmin with InstallAware Wizard.

Below, we describe the various sheets of the installation dialog together with the options that you have when installing WinAdmin.

- 'Welcome'
Welcome page of installation wizard in English. WinAdmin will be installed with languages German and English.
Click 'Next' to open the next sheet:
- 'ReadMe Text'
In this sheet you can view the openUTM WinAdmin V6.4 Release Notice.
Click 'Next' to open the next sheet:
- 'Destination Directory'
Here you can specify the directory in which WinAdmin is to be installed. By default this is "C:\openUTMWinAdmin". You can click 'Browse' to choose a different directory.
For Windows Vista or Windows 7 you should not install WinAdmin in the <program files> standard directory because WinAdmin must write or update files within the installation directory!
Click 'Next' to open the next sheet:
- 'Installed version found'
This sheet opens if an installed WinAdmin was found. You have two options:
 - Overwrite existing installation
 - Uninstall existing installation

- 'JRE Directory'

Here you must specify the Java directory in which the Java JRE or JDK which WinAdmin is to work with is installed. This must be installed on the destination computer **before** WinAdmin is installed (see above)!

When you have specified the Java directory, click 'Next' to open the following sheet:

- 'Program Shortcuts'

Here you define the name of the program group that is to be set up for WinAdmin during installation. You can also decide whether you want to create a link to WinAdmin for all Windows users or for the current user only.

Click 'Next' to open the final confirmation sheet.

4.2 Installation under Unix/Linux systems

An IzPack installation in the form of the JAR file "WinAdmin64A00.jar" is available for installing on Unix/Linux systems.

To install WinAdmin, go to the directory containing the file "WinAdmin64A00.jar" and execute the command "`<javapath>/bin/java -jar WinAdmin64A00.jar`". Then follow the instructions issued by the installation program.

Here, "`<javapath>`" stands for the directory in which the Java JDK is installed on the destination computer.

If you install WinAdmin via a graphical user interface the environment variable DISPLAY must be set. Otherwise the DISPLAY variable must not be set.

4.3 Installation in systems without graphical user interface

A `WinAdmin-auto.xml` file is mandatory to permit installation on systems without a graphical user interface. A template file will be delivered. Proceed as follows:

1. The automated installation on a Windows platform requires admin rights.
2. Open the file `WinAdmin-auto.xml` with a text editor and enter the required installation information as follows:
 - a. Specify the directory in which WinAdmin is to be installed in the `<INSTALL_PATH>` tag and
 - b. the directory in which the Java JDK is installed in the `<JDK_PATH>` tag.
3. In the relevant system, open a window for command entry, e.g. shell or DOS prompt.
4. Switch to the directory in which the JAR and XML files are located.
5. Unpack the JAR file using the following command

```
„java -jar WinAdmin64A00.jar WinAdmin-auto.xml“
```

This installs WinAdmin automatically.

Attention: Installing in systems without graphical user interface without using `WinAdmin-auto.xml`, this may lead to unexpected behaviors (e.g. null-pointer-exception).

5 What's new

5.1 ... in Version 6.4?

The most important changes compared to Version 6.3 are:

- WinAdmin supports all the new features of UTM V6.4, e.g switching data compression on or off and gives a graphical output of the pagepool statistics.
- WinAdmin supports new statistic collectors for monitoring the average number of saved UTM pages per data compression and for monitoring the pagepool statistics.

5.2 ... in Version 6.3?

The most important modifications compared to version 6.2 are:

- WinAdmin supports all the new features of UTM V6.3 relating to the administration program interface. These include, for example, the writing of KDCDEF statements to library elements on Inverse KDCDEF runs in BS2000 system.
- Introduction of a new property "value lifetime" for statistic collectors in order to limit the number of statistic values stored in the configuration database.

5.3 ...in Version 6.2?

The most important modifications compared to version 6.1 are:

- New property "Node Name" for Cluster Node and Lses
The new property "Node Name" for Cluster Node and Lses objects will be displayed in WinAdmin too.
- Output of new statistic values
Statistic values that are available as of UTM version 6.2 will be displayed in WinAdmin too. They are located in the properties dialog of a UTM application or a cluster on the "Current" sheet and in the properties dialog of a Tacclass..

5.4 ... in Version 6.1?

While the openUTM WinAdmin versions up to and including V6.0 were implemented in MFC (Microsoft Foundation Classes)-based C++, Version 6.1 is a completely new implementation in Java (100% pure Java).

Compared to the earlier versions, this has the great advantage that WinAdmin is not restricted to Microsoft Windows but can also run on Unix and Linux platforms. This means that it can also be employed by users who do not run or want to run any Windows machines in their data centers.

Although care was taken during the implementation of WinAdmin V6.1 to ensure that there were no fundamental changes to the user interface compared to the previous versions, certain changes have nevertheless been made wherever this was sensible and/or necessary.

The most important changes compared to Version 6.0 are:

- **Generation functions**

There are no longer any generation functions in Version 6.1 of WinAdmin. This version only provides the administration functions.

- **Configuration database**

WinAdmin V6.1 uses the H2 database which is fully implemented in Java. Microsoft Access databases or ODBC databases are no longer supported.

- **UPIC connections**

Earlier versions of WinAdmin could only maintain one UPIC connection at any one time. In Version 6.1, WinAdmin can now maintain parallel UPIC connections to all the UTM applications in a collection.

If a collection contains a number of different applications or if UTM clusters are used, this means that you do not constantly have to shut down and re-establish the UPIC connections. This improves performance.

- **Asynchronous data retrieval**

Whereas earlier versions of WinAdmin always retrieved the data synchronously from the UTM applications, WinAdmin V6.1 performs asynchronous data retrieval wherever possible. In practice, this means that the user simply initiates data retrieval in the GUI and the data is then actually fetched in a separate thread running in the background. The user interface is then refreshed when the data is available. This means, for example, that after clicking on a list with a large number of objects, users can now immediately open another list while the first is filled with data in the background.

See also section "Communication requests" on page 25.

- **Scaleable dialog boxes**

Unlike in previous versions in which there were only statically generated dialog boxes and property dialogs, all the dialog boxes and property dialogs used in Version 6.1 are scaleable, i.e. the user can change their size by "dragging" the border of the dialog box.

- **Tooltips in object lists and property dialogs**

UTM object lists and the property dialogs of UTM objects display tooltips which provide more detailed information about cell contents and object properties. This is particularly useful in the case of UTM cluster applications since it means that the values of the individual node applications are available in lists and dialog boxes that apply globally to the cluster.

See also section “Tooltips in lists and property dialogs“ on page 159.
- **Collections in the configuration database**

Collections are no longer administered in files but in the configuration database. This means that they can be opened by all WinAdmin instances that use the same configuration database.
- **UPIC configuration tool**

The UPIC configuration tool available in the previous version is no longer included in the present version since WinAdmin V6.1 is based on JConnect and no longer on UPIC (Windows).
- **More efficient filtering of object lists**

It is now easy to perform the very fast and efficient filtering of object lists,

See section “Filtering list objects“ on page 50.
- **Adapting the layout of the cluster and application subtrees**

The layout of the cluster and application subtrees in the navigation tree can now be adapted to meet the user's individual requirements.

See section "Layout of the Application and Cluster subtrees" on page 45. You can also hide the navigation window by pressing the keyboard shortcut **Ctrl+Shift+F2**. Pressing this keyboard shortcut again redisplayes it in its previous size. Selecting the 'Navigation Tree' command in the View menu has the same effect.
- **Improved behavior of message and statistic collectors in UTM clusters**

If message or statistic collectors are generated, modified or deleted in a UTM cluster, WinAdmin automatically performs the actions for all the currently active node applications in the cluster. This means that message and statistic collectors no longer have to be maintained at node-specific level. This task is now performed by WinAdmin.

See section “Message collectors in a cluster“on page 94, and “Statistic collectors in clusters“ on page 117.
- **Definition of object subtrees**

See section “Definition of partial lists”on page 64.

Alongside these general changes and extensions compared to the previous version, WinAdmin V6.1 naturally also supports all the new features of UTM Version 6.1.

6 Incompatibilities

This section lists the incompatibilities and the functions that are no longer present in WinAdmin V6.4 compared to WinAdmin V6.3.

- none

7 General

The objective of openUTM WinAdmin is to administer individual or multiple UTM applications. The applications can run on several different computers and under different operating systems (BS2000 systems, Unix/Linux systems, Windows systems).

The program runs on a PC and has a user-friendly graphical interface.

7.1 System platforms

openUTM WinAdmin runs under Microsoft Windows operating systems and on Unix and Linux platforms. It is a pure Java implementation.

More information on the supported platforms can be found in the Release Notice.

7.2 Resources required

A full installation of openUTM WinAdmin, including all the libraries and files used, requires approximately 20 MB of disk space.

The main memory requirements depend on the size of the UTM applications to be administered (to be more exact, on the number of objects contained in the applications) and on the number of applications in the individual collections. However, there should be at least 1 GB of main memory in order to guarantee uninterrupted operation.

To run WinAdmin properly, you really need a screen resolution of at least 1024 x 768 pixels. If the resolution is any lower, some of the dialog boxes cannot be displayed in full on the screen.

7.3 Programming language

openUTM WinAdmin is written entirely in Java and needs a JRE (Java Runtime Environment) of version Java SE 6 Update 25 or higher to run. This can be downloaded free of charge from the Oracle Web site (<http://www.oracle.com/technetwork/java/javase/downloads/index.html>) and must be installed on the destination computer **before** WinAdmin is installed.

7.4 Linking to UTM applications

The link to the UTM applications that are to be administered is established by means of the product JConnect that is shipped and installed together with WinAdmin.

UTM applications of Version 5.3 or higher are supported.

The program KDCWADMI and the KDCWADMI Tac have to be generated for UTM applications. The following KDCDEF statements must be included in the generation:

PROGRAM KDCWADMI, COMP=ILCS (in BS2000 system)
 or
 PROGRAM KDCWADMI, COMP=C (in Unix, Linux or Windows systems)
 and
 TAC KDCWADMI, PROGRAM=KDCWADMI, **CALL=BOTH**, ADMIN=Y

The program unit is already supplied with openUTM as of Version V4.0 and need not be linked explicitly. In addition, an access must be generated for the UPIC client WinAdmin, e.g. using the following KDCDEF statements:

```
BCAMAPPL <bcapplname>, T-PROT=RFC1006
TPOOL LTERM=<upict>, NUMBER=10, PRONAM=*ANY, PTYPE=UPIC-R
, BCAMAPPL=<bcapplname>
USER WINADMIN, PASS=C'XYZ', PERMIT=ADMIN, RESTART=NO
```

A password is not mandatory for the WinAdmin user. If a password is assigned, it must be printable and must comply with the character set restrictions prescribed by the UPIC interface.

Listener Port and Selector Format:

The listener port and the selector format have to be entered during the definition of a UTM application in WinAdmin (see section "UTM applications" on page 33) and their values must match the corresponding values of the UTM configuration.

With UTM applications in BS2000 system the listener port is always implicitly 102.

With Unix, Linux and Windows applications the listener port will be set in the BCAMAPPL KDCDEF statement using the parameter LISTENER-PORT=<port>.

With UTM applications in BS2000 the selector format is always implicitly Transdata.

With UNIX and Windows applications the selector format may be set in the BCAMAPPL KDCDEF statement too (parameter TSEL-FORMAT=<format>). Available are the values T (Transdata), E (EBCDIC) and A (ASCII). If this parameter is not specified in the BCAMAPPL statement, UTM takes the value Transdata if the character set of <bcapplname> matches the Transdata format or the value EBCDIC if this is not the case. It is recommended to explicitly specify a selector format using the TSEL-FORMAT parameter of the BCAMAPPL statement.

Note:

If a UPIC sign-on service is generated in the UTM application (SIGNON UPIC=YES), this sign-on service should be deactivated for the BCAMAPPL that will be used to communicate with WinAdmin (BCAMAPPL <bcamapplname>, SIGNON-TAC=*NONE).

If a UPIC sign-on service is assigned to this BCAMAPPL nevertheless, the TAC KDCWADMI must be considered when programming the sign-on service.

Access via LTERM/PTERM:

In the example above a TPOOL KDCDEF statement is used for access of the UPIC client WinAdmin. Alternatively a LTERM and a corresponding PTERM KDCDEF statement could be used to generate an explicit connection:

```
LTERM <upict>
PTERM <upict>, LTERM=<upict>, PRONAM=<upicpro>, BCAMAPPL=<bcapplname>,
PTYPE=UPIC-R
```

<upicpro> stands for the WinAdmin host name. For listener-port and selector-format see the section above. In WinAdmin, the corresponding UPIC connection definition must use the PTERM name <upict>.

Using a LTERM/PTERM connection for WinAdmin you can minimize UTM V6.4 response times under work load if you generate

```
MAX PRIVILEGED-LTERM=<upict>
```

7.5 Supported UTM versions

Version 6.4 of WinAdmin supports the administration of UTM applications of Version 6.0 and higher.

7.6 Language

The output of openUTM WinAdmin is in English or German. It is possible to switch over from one language to the other while working with WinAdmin if both language variants have been installed. However, this switch-over only takes effect in the next WinAdmin session.

8 Starting and stopping WinAdmin

You usually start WinAdmin using the start script “startwinadmin.sh” (Unix/Linux systems) or “startwinadmin.cmd” (Windows systems) that can be found in the “shsc” subdirectory of the WinAdmin installation directory.

On Windows systems, WinAdmin can also be started using the installed program shortcut.

Please note:

On Unix/Linux systems the current directory (pwd command) on which WinAdmin is started is important. If different users work with a WinAdmin installation the following is recommended: A user should start WinAdmin always in a certain directory under his HOME directory to handle his personal program settings. Then a user must set the path of the common configuration database or create an empty database on first WinAdmin run.

To end WinAdmin, either choose the “Shut...” command in the File menu or click the Close button in the title bar of the program window (usually a cross at the right-hand edge of the title bar).

9 Configuration database

WinAdmin operation requires a database, the so-called configuration database. This database contains the administration data for all the UTM applications known to WinAdmin, the hosts on which these applications run, the UPIC connections to the applications and the WinAdmin users and the application collections.

The database also contains the data for (user-specific) options, authorizations, display filters etc.

In earlier versions of WinAdmin, the database was a Microsoft Access database but could also be a database that could be accessed via ODBC. In contrast, as of Version 6.1 the only database type employed is H2. H2 is an SQL database written exclusively in Java that runs not only on Windows but also on Unix and Linux platforms. It is available as freeware (on the Internet under <http://www.h2database.com>). However, the runtime system for the database is shipped and installed together with WinAdmin.

The configuration database must be declared to WinAdmin once. You generally do this the first time you start WinAdmin V6.4. For more information on selecting a configuration database, please refer to section "Selecting and creating a configuration database" on page 22.

If several instances of WinAdmin are used on different computers, they should all use the same configuration database, in order to keep the generation data consistent and not have to administer it in different places. Of course, for this to be possible all the WinAdmin instances must have access to the configuration database.

In order to avoid loss of data caused by disk or program errors, it is essential to make backup copies of the configuration database at regular intervals and keep these on a different computer or on an external storage medium (see also section "Backup copies" on page 23).

9.1 Selecting and creating a configuration database

When WinAdmin is started for the first time, a dialog box is displayed requesting the user to select a configuration database. There are three possibilities:

- Create new configuration database

If this option is chosen, WinAdmin creates a new, empty configuration database. If you choose this option and then click the 'OK' button in the dialog box, you must next select a directory (empty if possible) in which the database is to be created

The database directory must be located on a medium that can be accessed from the computer on which WinAdmin is installed and for which the user ID under which WinAdmin was started has read and write access.

WinAdmin then creates the new configuration database.

- Create new database and import WinAdmin V6.0 database

If this option is chosen, WinAdmin creates a new, empty configuration database as in the option described above and then imports an existing WinAdmin V6.0 database into this. However, the database that is to be imported must be a Microsoft Access database that has been used by WinAdmin V6.0. ODBC databases cannot be imported.

Once you have clicked 'OK' to close the dialog box and – as in the first option – selected a database directory for the configuration database, you must next choose the WinAdmin V6.0

database that is to be imported. To do this, you must select a Microsoft Access database file (*.mdb).

WinAdmin then creates the new configuration database and imports the data from the Microsoft Access database.

- **Select existing configuration database**

If this option is chosen, you must select an H2 database that has already been generated by WinAdmin V6.1 (*.h2.db) and which is to be used by WinAdmin.

9.2 Backup copies

To protect against data loss due to disk or program errors as well as maintain database performance, you are strongly recommended to reorganize and back up the configuration database at regular intervals. Ideally, the backup copies should be stored on another computer or on an external storage medium.

Depending on the database type, WinAdmin offers you different ways of reorganizing the configuration database and/or creating backups.

9.3 Troubleshooting

To avoid problems during the operation of the configuration database or to eliminate problems that have occurred, you should pay special attention to the following points:

- Since WinAdmin itself generates the required database tables and columns when creating the configuration database, the database user, via whom WinAdmin is connected to the database, must possess the privileges necessary to create and modify tables.
- In addition, the database user must possess the right to read, edit, add and delete records for all the tables generated by WinAdmin.
- The database or individual tables generated by WinAdmin must not be write-protected.
- The database may not be opened exclusively by another process or be fully write-protected.

10 The program interface

At the top edge of the application window, you can see the title bar and, directly below this, the menu bar (see also section “Backup copies” on page 169) which contains the generally applicable commands.

At the lower edge, there is a status bar that contains information on the current state of the program (see also section “The status bar” on page 171).

The main central area of the program window is subdivided into three panes. The left-hand pane contains the navigation window together with the navigation tree (see also section “The navigation window” on page 42).

The right-hand area is itself subdivided into two panes. At the bottom is the protocol window in which program messages are output. The remainder of the program window constitutes the work area in which the work area windows (e.g. object lists) are displayed.

11 Communication requests

In contrast to previous versions of the program, as of WinAdmin V6.1, data is requested asynchronously from the UTM applications from within the user interface whenever this is possible. For example, if you open an object list then the list is displayed immediately in the user interface even if the data that ought to be present in the list has not yet been retrieved from the UTM applications. To ensure that the user interface remains available and is not blocked while the data is retrieved, WinAdmin then requests the data in a separate thread which is independent of the user interface (asynchronous). When the data is available, the user interface is notified and refreshed accordingly. The data then becomes visible in the list.

This type of request for data from a UTM application is referred to as a "communication request" below. Until terminated, such a request is "open". The term communication request is not restricted to the retrieval of data from UTM applications but is also used for all other communication jobs (availability checks, creation and deletion of objects, switching between protocol files etc.).


If a communications error occurs during the processing of a communication request (e.g. if the UTM application is no longer available or if it was not possible to establish the connection) then a corresponding message is output in the protocol window.


WinAdmin creates a thread that is used to process communication requests to the application for every UTM application in the open collection. This thread also maintains the UPIC connection to the UTM application. In addition, there is a queue for each of these threads. New communication requests are appended to these queues and are worked through in sequence.


Alongside situations in which the result of a communication request is not required "immediately", there are also situations in WinAdmin in which the result of a communication request is needed "immediately" in order to continue working. This type of request is referred to as a "synchronous" communication request. Synchronous requests have to be processed as quickly as possible and are therefore inserted at the front of the queue. They are therefore processed immediately the request that is currently being processed is terminated.


Users are informed at various locations in the GUI if there are any open communication requests:

- In the status bar

The icon  is displayed in the right-hand part of the status bar. If there are open communication requests then a number indicating the number of these open communication requests is displayed to the right of the icon.
- In object lists

The icon  is also displayed above object lists. Here again, the number of open communication requests for this object list is displayed to the right of the icon. Unlike in the status bar, the icon is only displayed if open communication requests exist. This should be borne in mind, for example, when refreshing a list.
- In property dialogs

The icon  and a corresponding number are displayed at the bottom right of property dialogs if there are any open communication requests involving data that is to be displayed in the property dialog in question. As in the case of object lists, the icon is only displayed if open communication requests exist. This should be borne in mind, for example, when refreshing the data in the sheet.

An overview of the communication requests can be obtained by clicking the  icon in the status bar. This opens a list of communication requests which is described in greater detail in the section below.

11.1 List of communication requests

As mentioned above, you can view this list by clicking the icon  in the status bar.

It contains an entry for every communication request that is currently open. In addition, a certain number of terminated communication requests are also displayed. This number can be defined in the WinAdmin settings dialog (Extras menu, "Settings" command, 'Max. Stored Communication Requests' field of the "General" tab). If the number of terminated requests exceeds the value defined here then the oldest requests are deleted from the list as required.

This list contains detailed information on the communication requests, such as the wait and processing times and the volume of data sent and received. In addition, you are able to interrupt or remove communication requests that are still present in the queue.

The list contains the following columns:

- Requested
Time at which the communication request was issued in the format YYYY-MM-DD hh:mm:ss (year, month, day, hour, minute, second).
- Number
Queue-specific serial number for the communication request.
- Wait time
Time during which the communication request has been waiting in the queue in the format m.ss.mmm (minutes, seconds, milliseconds).
- Started
Time at which the processing of the communication started in the format hh:mm:ss (hour, minute, second).
- Processing.
Processing time for the communication request in the format m.ss.mmm (minutes, seconds, milliseconds).
- Finished
Time at which the processing of the communication was terminated in the format hh:mm:ss (hour, minute, second).
- Status
Current status of the communication request.
Possible values:
 - Created
The request has been created but is still present in the queue.
 - Waiting to be performed
The request is present in the queue.

- Prepared to be performed
The request has been taken from the queue and is being prepared for processing.
 - Performing
The request is currently being processed.
 - Finished
Processing of the request is finished.
 - Cancelled
The request has been cancelled.
- UTM Application
The UTM application that is the recipient of the communication request.
 - Description
Description of the communication request.
 - Communication Steps
The number of communication steps (so far) performed for the processing of the application.
 - Send
Volume of data sent to UTM (in bytes).
 - Received
Volume of data received from UTM (in bytes).
 - Communication Time
Actual processing time for the request in the format m.ss.mmm (minutes, seconds, milliseconds). This is the total of the times for all the communication steps between the sending and reception of the data.
 - Exception
Indicates whether an error has occurred during the communication request.
If an exception has occurred then a tooltip containing the identified exception is displayed if the mouse pointer is left unmoved briefly over the corresponding exception checkbox.
 - Communication Initiator
If the communication request was generated by an object list or property dialog (for example) then this is specified here

You can cancel communication requests that are pending or are currently being processed. Pending requests are removed from the queue and then displayed as "Cancelled". Communication requests that are currently being processed can only be cancelled between two communication steps. It is therefore not possible to cancel started requests that involve only one communication step.

Communication requests that have been terminated can be removed from the list.

To remove terminated communication requests, click the 'Remove' button below the list. The context menu displayed if one or more communication requests are selected in the list also contains commands that can be used to remove or cancel requests. WinAdmin asks you to confirm that you really want to remove or cancel the requests.

12 WinAdmin objects

In order to administer UTM applications using WinAdmin (administer and/or configure), WinAdmin requires various items of information (e.g. the network name of the host on which the UTM applications are running). WinAdmin records this information in various types of object, the WinAdmin objects, and stores it in the configuration database.

Naturally, you can also generate, edit and delete the WinAdmin objects via the WinAdmin graphic user interface.

For each WinAdmin object, there is a property dialog in which you can edit the properties of an existing object or define the properties of a new object that you want to generate. In addition, for each WinAdmin object type there is an object list containing all the currently defined WinAdmin objects. These lists have the same properties as the object lists in which UTM objects are displayed (see section "Object lists" on page 42).

The following types of WinAdmin object exist:

- WinAdmin users
The WinAdmin users provide information about user authorizations, for example whether or not the user who is currently signed on to WinAdmin is authorized to generate a certain type of UTM object in application A.
- Hosts
The hosts describe the computers on which the UTM applications (and the WinAdmin instances) run.
- Applications
These objects describe the properties of the UTM applications that are to be administered using WinAdmin.
- UPIC connections
These objects describe the properties of the UPIC connections between the WinAdmin instances and the running UTM applications. If a UTM application is not simply to be generated, but also administered, by means of WinAdmin then at least one UPIC connection to this application must be defined.
- Collections
Collections make it possible to group together applications that are to be administered together. Only one application collection can be open at any one time in WinAdmin.

The individual WinAdmin object types and their property dialogs are presented in more detail in the following sections.

12.1 WinAdmin users

Objects of "WinAdmin user" type are used to administer the authorizations of the individual WinAdmin users. When WinAdmin starts, the user must sign onto the program. This requires the entry of the WinAdmin user name. If a password is assigned to this WinAdmin user then the user must also specify this before he or she can work with WinAdmin.

There is a predefined user called the "master". This user has all authorizations and cannot be deleted or modified. Depending on the number, tasks and authorizations of the administrators who work with WinAdmin, it may be worthwhile setting up additional "WinAdmin users" and assigning those privileges which are suitable for the tasks they have to perform. See also section "Security - users and authorizations" on page 127. Only a WinAdmin user with the master privilege is authorized to generate, delete or modify a WinAdmin user.

The fields in the property dialog:

- 'Name' edit box
You can select any name for the WinAdmin user. However, it has to be unique within the relevant configuration database. The name can have a maximum length of 28 characters.
- 'Password' edit box
Specifies the user's password. It can have a maximum length of 32 characters. The entry is optional.
- 'Overall Privileges'/Master' dropdown box
Specifies whether the user has overall master privileges or not.
- 'Overall Privileges'/Administration Write' dropdown box
Specifies whether or not the user has overall write privileges for administration. This privilege includes the overall read privilege for administration.
- 'Overall Privileges'/Administration Read' dropdown box
Specifies whether or not the user has overall read privileges for administration.
- 'Remark' edit box
Here you can enter a comment relating to the WinAdmin user

12.1.1 Displaying the WinAdmin user list

You can see a list of all the users defined in the configuration database by

- choosing the 'WinAdmin Users' command in the File menu, or
- clicking the "WinAdmin Users" node under "WinAdmin Objects" in the navigation tree, or
- choosing the 'WinAdmin User List' entry in the context menu of the "WinAdmin Users" node in the navigation tree below "WinAdmin Objects".

12.1.2 Generating a new WinAdmin user

You can define a new WinAdmin user with one of the methods below:

- If the WinAdmin user list is active: choose the 'New WinAdmin User...' command in the Edit menu.
If precisely one host is selected in the user list then the new WinAdmin user is assigned the properties of the selected user.
- If the WinAdmin user list is active: choose the 'New WinAdmin User...' command in the context menu of an already defined user in the list.
The properties of this 'source user' are assigned to the new user.
- In the navigation tree: choose the 'New WinAdmin User...' command in the context menu of the 'WinAdmin Users' node.
- In the navigation tree: choose the 'New WinAdmin User...' command in the context menu of an existing user.
The properties of this 'source user' are assigned to the new user.

12.1.3 Deleting a WinAdmin user

You can delete an existing WinAdmin user with one of the methods below:

- If the WinAdmin user list is active: select the user you want to delete in the list and then choose the 'Delete Selected WinAdmin User...' command in the File menu.
- If the WinAdmin user list is active: choose the 'Delete WinAdmin User...' command in the context menu of the user you want to delete.
- In the navigation tree: choose the 'Delete WinAdmin User...' command in the context menu of an individual WinAdmin user (subtree: "WinAdmin Objects"/"WinAdmin Users").

It is not possible to delete the WinAdmin user with the name "Master".

12.2 Hosts

The WinAdmin objects of type "Host" describe the computers on which the UTM applications to be administered run (the application computers).

The fields in the property dialog:

- 'Name' edit box
Select any name for the host. However, it has to be unique within WinAdmin (to be more precise, within the current configuration database). The name can have a maximum length of 16 characters.

- 'Netname' combo box
You specify the name under which the host will be known in the network here. You can enter the name either in the edit box of the combo box or select the entry "<resolve from IP-Address>" from the combo box's list box. In this case you must enter the host's valid IP address in the combo box 'IP-Address'. WinAdmin will try to resolve the host's netname from the specified IP address when you click 'OK' or 'Apply'.
- '...' button
Clicking this button opens the "Specify Hostname/IP-Address" dialog box which is described in the next section.
- 'UTM Hostname' edit box
This contains the name under which the corresponding host is known in UTM. This is particularly important in the case of cluster node applications. Here, the name corresponds to the 'Hostname' value specified for the cluster node object.
- 'IP-Address' edit box
Here you can specify the IP address of the host. Alternatively, you can determine it using the '...' button (see above).
- 'Operating System' dropdown box
Specifies the operating system used on the host. The available operating systems are BS2000 systems, Unix/Linux systems and Windows systems.
- 'Operating System Remark' edit box
Enter a remark concerning the employed operating system in this edit box (e.g. the precise system version number). Maximum length: 255 characters. The entry is optional.
- 'openFT Available' dropdown box
Specifies whether or not openFT is available on the host. The default value is 'Unknown'. In this case, WinAdmin itself tries to determine what communications system to use for file transfer functions. If possible, you should select 'Yes' or 'No' here. See also section "File transfer functions in WinAdmin" on page 129..
- 'FTP Available' dropdown box
Specifies whether or not FTP is available on the host. The default value is 'Unknown'. In this case, WinAdmin itself tries to determine what communications system to use for file transfer functions. If possible, however, you should select 'Yes' or 'No' here. See also section "File transfer functions in WinAdmin " on page 129.
- 'Remark' edit box
A comment concerning the host can be entered here. Maximum length: 255 characters. The entry is optional.

12.2.1 The “Specify Hostname/IP-Address” dialog

This dialog box allows you to check whether or not a computer can be accessed from the machine on which WinAdmin is running. To do this, enter the netname or IP address of the host in the 'Hostname/IP-Address' box. Even while you are entering this data, WinAdmin attempts to contact the host using the entered portion of the netname or IP address.

The result of this attempt to establish contact is then output in the lower part of the dialog box. The netname ('Hostname') and IP address (if it has been possible to determine both of these) and IP address are output together with the response time.

If you click the 'Ping' button, WinAdmin sends a "Ping" command together with the entry in the 'Hostname/IP-Address' box and displays the ping result in a separate message box.

If you click the 'OK' button to exit the dialog box then host's netname and IP address (if it has been possible to determine them) are entered in the corresponding boxes in the host property dialog from which the dialog box was called.

12.2.2 Displaying the host list

You can see a list of all the hosts defined in the configuration database by

- choosing the 'Hosts' command in the File menu or
- clicking the "Hosts" node under "WinAdmin Objects" in the navigation tree, or
- choosing the 'Show Host List' entry in the context menu of the "Hosts" node in the navigation tree below "WinAdmin Objects".

12.2.3 Generating a new host

You can generate a new host with one of the methods below:

- If the host list is active: click the 'Add' button below the list.
- If the host list is active: choose the 'New Host...' command in the context menu of an already defined host in the list.
- In the navigation tree: choose the 'New Host...' command in the context menu of the 'Hosts' node.
- In the navigation tree: choose the 'New Host...' command in the context menu of an existing host. The properties of this 'source host' are assigned to the new host.

Once a host has been successfully created, WinAdmin asks whether it is also necessary to create a UTM application that is to run on the new host. If this is the case then the dialog box for the creation of a new UTM application is displayed.

12.2.4 Deleting a host

You can delete a host with one of the methods below:

- If the host list is active: select the host you want to delete in the list and then choose the "Remove" command below the list.
- If the host list is active: choose the 'Delete Host...' command in the context menu of the host you want to delete.
- In the navigation tree: choose the 'Delete Host..' command in the context menu of an individual host (subtree: "WinAdmin Objects"/"Hosts").

When you delete a host, all the UTM applications defined on this host are also deleted (see section "Deleting an application" on page 35).

12.3 UTM applications

The WinAdmin objects of type "UTM applications" describe the UTM applications that are to be administered with WinAdmin. A WinAdmin object of type "Host" describing the computer on which the application is to run must be generated before a UTM application can be generated in the configuration database (see previous section "Hosts").

If the application is also to be administered with WinAdmin (as far as WinAdmin is concerned, this is the case if the current WinAdmin user has administration privileges), at least one UPIC connection has to be entered for the UTM application so that WinAdmin can access the application through UPIC (see section "UPIC connections" on page 36).

The fields in the property dialog:

- 'Name' edit box
You can select any name for the application. However, it has to be unique on the selected host. For the sake of clarity, however, you are advised to enter a "genuine" UTM application name here (MAX APPLINAME). The name can have a maximum length of 16 characters.
- 'Host' dropdown box
Specifies the computer on which the application is to run. All hosts known to WinAdmin are listed in the dropdown box. The host may need to be defined first (see section "Hosts" on page 30).
- 'Cluster' text box
If the UTM application forms part of a UTM cluster and WinAdmin has already detected this then the name used in WinAdmin for this cluster is specified here.
This field is not displayed when you create a new application.
- 'UTM Version' dropdown box
Specifies the version of the UTM application. The dropdown box contains all versions known to WinAdmin.
If necessary, WinAdmin updates the version automatically on the first successful contact with the UTM application.

- 'Version Remark' edit box (concerning version)
Enter a remark concerning the version in this edit box (e.g. the precise version name). Maximum length: 64 characters. The entry is optional.
- 'Standard Port' edit box
Specifies the port through which the UPIC connection is to be established (default port number 102). An entry in this field is optional.
The port entered here must match the corresponding value of the UTM configuration (see section "Linking to UTM applications" on page 18).
- 'Selector Format' dropdown box
Specifies the selector format for the UPIC connection. Possible formats are 'ASCII', 'EBCDIC' and 'Transdata'.
The selector format specified here must match the corresponding value of the UTM configuration (see section "Linking to UTM applications" on page 18).
- 'UPIC Connection' dropdown box
Specifies the UPIC connection used for the UTM application. If multiple UPIC connections are defined to a UTM application then you can select one of these here.
This box is not displayed when you create a new application.
- 'Keep Open Transaction when Possible' checkbox
If this option is enabled then, in the case of read only calls from the UTM administration interface, the transaction is kept open if possible (PEND KP in the administration program unit KDCWADMI of the relevant UTM application). This reduces the transaction management overhead in the UTM application.
With UTM applications with a static XA database connection, it is recommended that this option should not be enabled, because otherwise transactions may be terminated abnormally on the database side as a result of timeouts. If the option is not enabled, the administration program unit performs a PEND FI rather than a PEND KP.
- 'Remark' edit box
A comment concerning the UTM application can be entered here. Maximum length: 128 characters. The entry is optional.

12.3.1 Displaying the application list

You can see a list of all the UTM applications defined in the configuration database by

- choosing the 'UTM Applications' command in the File menu or
- clicking the "UTM Applications" node under "WinAdmin Objects" in the navigation tree, or
- choosing the 'Show Application List' entry in the context menu of the "Applications" node in the navigation tree below "WinAdmin Objects".

12.3.2 Generating a new application

You can generate a new application with one of the methods below:

- If the application list is active: click the 'Add' button below the list.
- If the application list is active: choose the 'New UTM Application...' command in the context menu of an already defined application in the list.
- In the navigation tree: choose the 'New UTM Application...' command in the context menu of the 'UTM Applications' node.
- In the navigation tree; select the 'New Application...' command in the context menu of an existing application (either in the subtree "WinAdmin Objects/" UTM Applications" or in the subtree "WinAdmin Objects/"Hosts").
- In the navigation tree: choose the 'UTM New Application...' command in the context menu of an existing host (subtree "WinAdmin Objects/"Hosts")..

After a UTM application has been successfully created, WinAdmin asks whether a UPIC connection should also be created for this application. If this is the case then the dialog box for the creation of a new UPIC connection is displayed.

12.3.3 Deleting an application

You can delete an application with one of the methods below:

- If the application list is active: click the 'Remove' button below the list.
- If the application list is active: choose the 'Delete UTM Application...' command in the context menu of the host you want to delete.
- In the navigation tree: choose the 'Delete UTM Application..' command in the context menu of an individual application (subtree "WinAdmin Objects/" UTM Applications" or "WinAdmin Objects/"Hosts").

If the UTM application is present in the currently open application collection then it cannot be deleted. In this case, either the application must first be removed from the open collection or the collection that is currently open must be closed.

When you delete an application, all the objects belonging to this application are also deleted from the configuration database. In particular, this also includes the UPIC connections, message collectors (including all saved messages) and statistic collectors (including all saved statistics values) defined for this UTM application. In addition, the UTM application is removed from all the application collections in which it is present.

12.4 UPIC connections

The WinAdmin objects of type "UPIC Connection" describe the UPIC connections required by WinAdmin to communicate with the UTM applications if they are to be **administered** via WinAdmin. If the current WinAdmin user has administration privileges, WinAdmin searches for an 'appropriate' UPIC connection to each UTM application in the collection it is opening. If there is no UPIC connection available for an application, a message is issued and the application does not appear in the application collection (see below).

The fields in the property dialog:

- 'UTM Application' combo box
Specifies the UTM application to which the UPIC connection is to be established.
When a new UPIC connection is established, a UTM application can be selected from the dropdown box here. If a UPIC connection is modified then it is not possible to modify the UTM application. It is then simply specified in a text box.
- 'Bcamappl/Appliname' edit box
Specifies the application name which is to be used to establish the UPIC connection. This will usually be a BCAMAPPL in the UTM generation.
- 'Pterm' edit box
If the UPIC connection is to be established via a specific PTERM in the UTM generation then the name of the corresponding Pterm must be specified here.
If the UPIC connection is to be established via a terminal pool (TPOOL) in the UTM generation then you can either leave this edit box empty or leave the edit box's default value "<Tpool-Coupling>" unchanged.
- 'Port' edit box
If nothing is entered here then the default port specified for the UTM application is used for communication.
If a port other than the standard port is to be used then you should specify this here.
- 'Selector Format' dropdown box
You select the selector format here.
Either the standard selector format specified for the UTM application or another format can be used.
- 'User' edit box
If the UPIC security function is to be used then you must specify the name of the UTM user under which the UPIC connection signs on at UTM here. Otherwise, you can leave this field empty. The user name specification is case-sensitive.
- 'User Password' edit box
Here you specify the password of the UTM user under which the UPIC connection signs on at UTM. If the specified user does not have a password or if no UTM user was specified then leave this text box empty.
- 'Connection Timeout (sec)' dropdown box
Here you can decide whether or not to set a timer for the monitoring of the UPIC connection.

If <no timeout> is selected here then no timer is set and WinAdmin waits until the UTM application returns a response.

If the entry <seconds> (WinAdmin default) is selected then the global WinAdmin timeout setting is used. In this case, the value can be modified centrally at one location in the WinAdmin settings (General tab).

Alternatively, you may also select a value other than the WinAdmin default here.

If no response is received from the UTM application in the specified time then WinAdmin terminates the connection and identifies the relevant UTM application as not available.

- 'Encryption' dropdown box

Here you can decide whether or not encryption is to be used on the UPIC connection. When the connection is established, the encryption level to be used is negotiated with UTM.

If no encryption is available then the connection is used without encryption even if the value 'Yes' is selected here.

- 'Remark' edit box

Here you can enter a comment relating to the UPIC connection. Maximum length: 255 characters. This specification is optional.

12.4.1 Displaying the UPIC connection list

You can see a list of all the UPIC connections defined in the configuration database by

- choosing the 'Upic Connections' command in the File menu or
- clicking the "UPIC Connections" node under "WinAdmin Objects" in the navigation tree, or
- choosing the 'Upic Connections List' entry in the context menu of the "Upic Connections" node in the navigation tree below "WinAdmin Objects".

12.4.2 Creating a new UPIC connection

You can define a new UPIC connection with one of the methods below:

- If the UPIC Connection list is active: click the 'Add' button below the list.
- If the Upic Connection list is active: choose the 'New UPIC Connection...' command in the context menu of an already defined connection in the list.
- In the navigation tree: choose the 'New UPIC Connection...' command in the context menu of the UPIC Connections' node.
- In the navigation tree; select the 'New UPIC Connection...' command in the context menu of an existing connection (either in the subtree "WinAdmin Objects/" UPIC Connections" or in the subtree "WinAdmin Objects"/"Hosts").

12.4.3 Deleting an UPIC connection

You can delete an UPIC connection with one of the methods below:

- If the UPIC Connection list is active: select the UPIC connection you want to delete in the list and then choose the 'Remove' button below the list..
- If the UPIC connection list is active: choose the 'Delete UPIC Connection...' command in the context menu of the UPIC connection you want to delete.
- In the navigation tree: choose the 'Delete UPIC Connection...' command in the context menu of an individual UPIC connection (subtree "WinAdmin"/" UPIC Connections" or "WinAdmin Objects"/"UTM Applications" or "WinAdmin Objects"/"Hosts").

12.5 Collections

You have the option of creating so-called "collections" of UTM applications in which applications that are to be administered (i.e. administered and/or generated) jointly are combined into groups. These collections are stored in the form of files by WinAdmin and can then be reopened at any time in subsequent program sessions.

To administer or generate applications you must always first open a collection that contains the relevant applications. A collection can also contain a single UTM application.

Only one collection can be open at any one time in WinAdmin. If you open a second collection, the first is closed automatically (although it is, of course, saved and can be reopened at any stage).

To this end, the File menu of WinAdmin contains the 'New Collection...' and 'Open Collection...' commands; if a collection is open, the commands 'Edit Collection...' and 'Close Collection' are also available. The WinAdmin toolbar also contains buttons for the 'New Collection' and 'Open Collection' functions. The most recently used collections are also listed in the File menu and can therefore be opened again quickly.

WinAdmin always recognizes a generic collection named '<All UTM Applications>' which always contains all the UTM applications that are currently defined in the configuration database. The collection cannot be deleted or modified.

If a collection is open then you can also add applications to this collection or remove applications from it by choosing the 'Add to Collection' (in the "WinAdmin Objects" subtree) or 'Remove from Collection' commands.

The fields of the property dialog:

- 'Name' edit box
You can choose any name you want for the collection, but the name must be unique within the configuration database. The name can be up to 64 characters in length.
- 'Group' combo box
The group name specified here is used to arrange the collection node accordingly in the navigation tree below the "WinAdmin Objects"/"Collections" node. If nothing is entered here then the collection node is entered directly below this node in the navigation tree. If a group name is specified here then an intermediate node with the name of the group is inserted between the "WinAdmin Objects"/"Collections" node and the collection node.

You can also specify more than one intermediate node. In this case, they must be separated from one another using the "/" character. This type of specification resembles a file path in a Unix/Linux system.

All the collection groups that are known in the configuration database are listed in the combo box's dropdown box where they can be selected quickly and easily. You can also enter a group name that is not listed in the dropdown box in the edit box section of the combo box..

The group name can be up to 64 characters in length. This specification is optional.

- 'Remark' edit box

Here you can enter a comment relating to the collection. The comment can be up to 255 characters in length. This specification is optional.

- 'Auto Open' checkbox

Specifies whether this collection is to be opened automatically when the WinAdmin program starts. This option can only be set for one collection in any WinAdmin instance.

WinAdmin saves this setting in the WinAdmin instance's configuration file on the local computer (not in the configuration database).

- 'Show All UTM Applications' checkbox

If this option is activated then all the UTM applications defined in the configuration database are displayed in the list of UTM applications below it. If it is not activated then only those UTM applications that are currently present in the collection are displayed.

- List of UTM applications

This list contains either all the UTM applications in the configuration database or only those UTM applications that are currently present in the collection (depending on whether the option 'Show All UTM Applications' is selected).

You can check or uncheck the boxes in the table's 'In Collection' column to add applications to the collection or remove applications from it. To do this, simply click the corresponding checkbox. If it was previously selected then the application is removed from the collection and if it was not previously selected then it is now added to it. Of course, you can only add applications in this way if all the UTM applications are displayed in the list.

You can also use the 'Add' and 'remove' buttons below the list to add applications to the collection or remove applications from it. To do this, select the applications that you want to add or remove in the list and click the corresponding button.

If a UTM application that is assigned to a cluster in the configuration database is added or removed then the other applications in the database that are assigned to the same cluster are also added to or removed from the collection.

You can enter a text in 'Filter Text' which is then used to restrict the number of applications displayed in the list.

12.5.1 Displaying the collection list

You can display a list of all the collections defined in the configuration database by

- choosing the 'Collections' command in the File menu or
- clicking the "Collections" node under "WinAdmin Objects" in the navigation tree, or
- choosing the 'Show Collection List' in the context menu of the "Collections" node under "WinAdmin Objects" in the navigation tree.

12.5.2 Generating a new collection

You can define a new collection in the following ways:

- Choose the 'New Collection...' command in the File menu.
- If the collection list is active: Click the 'Add' button below the list.
- If the collection list is active: Choose the 'New Collection...' command in the context menu of a collection that is already defined in the list.
- In the navigation tree: Choose the 'New Collection...' command in the context menu of the "Collections" node.
- In the navigation tree: Choose the 'New Collection...' command in the context menu of an existing collection in the "WinAdmin Objects"/"Collections" subtree

12.5.3 Deleting a collection

You can delete an existing collection in the following ways:

- If the collection list is active: Select the collection that is to be deleted in the list and click the 'Remove' button below the list.
- If the collection list is active: Choose the 'Delete Collection...' command in the context menu of the collection that you want to delete.
- In the navigation tree: Choose the 'Delete Collection...' command in the context menu of an individual collection in the "WinAdmin"/"Collections" subtree.

The collection that is currently open cannot be deleted.

12.5.4 Opening a collection

You can open a collection in the following ways:

- Choose the 'Open Collection...' command in the File menu.
This opens a dialog box that displays all the collections currently defined in the configuration database. You can then double-click a collection or select a single collection and click the 'OK' button to open the collection.
- Choose a collection that has already been opened in the File menu.
The last opened collection is listed in the File menu with its name preceded by the number '1'. Other collections that have already been opened may also be displayed below this with larger numbers.
- If the collection list is active: Choose the 'Open Collection' command in the context menu of the collection that you want to open.
- In the navigation tree, click the node for a defined collection below the "WinAdmin"/"Collections" node.

- In the navigation tree, choose the 'Open Collection' command in the context menu of a defined collection below the "WinAdmin"/"Collections" node.

If another collection is already open when you open a collection, WinAdmin asks whether the already opened collection should be closed and the collection you have just chosen opened instead of it.

Automatically opening a collection on program start

An existing collection can be opened automatically on program start for each WinAdmin instance. You can specify this action in the property dialog of the relevant collection or, alternatively, by means of the 'Open Automatically On Program Startup' command in a collection's context menu.

12.5.5 Closing a collection

You can close the currently open collection in the following ways:

- Choose the 'Close Collection' command in the File menu.
- Choose the 'Close Collection' command in the context menu of the node corresponding to the open collection in the navigation tree.

13 The navigation window

The navigation window is located at the left-hand edge of the WinAdmin program window. It allows you to access the majority of the objects and functions provided by WinAdmin.

WinAdmin displays an open collection as a tree structure in a window (in the same way as the Windows Explorer shows individual drives and directories). This structure is referred to below as the "collection tree". The collection tree possess various nodes which may represent, for example, UTM applications or UTM objects. Nodes at the top level are referred to as roots and nodes at the lowest level as leaves. All nodes below a given node or root are referred to as a "subtree".

The collection tree is handled according to the Windows conventions. For example, a '+' sign (or another equivalent symbol depending on the defined Look&Feel) in front of a node means that there is a subtree below that is not currently displayed. This subtree is expanded by clicking the '+' sign (or double-clicking the node). Once the node has been expanded, a '-' sign replaces the '+' in front of the node. If you click the '-' sign, the subtree is closed again.

As you move the mouse pointer over the nodes in the navigation tree then the entry below the pointer is displayed in blue so that you can easily identify the node above which the mouse pointer is positioned at any given time. Many of the nodes are also displayed underscored when the mouse pointer is located over them. This indicates that single-clicking the node triggers an action. If you leave the pointer above this type of node for a short time then a tooltip is displayed. This describes the action that is performed if you single-click the node. This usually consists of opening a list of the objects represented by the node in the navigation tree.

The root of the tree now specifies the current WinAdmin user, i.e. the user name used on sign-on to the current WinAdmin session. On the next level, you find the four nodes "WinAdmin Objects", "Actions", "Display Filters" and "Open Windows". If a collection is open then there is also a fifth node "Collection ..." on the topmost level.

■ "WinAdmin Objects"

The subtree associated with this node allows you to access the following five types of WinAdmin objects: WinAdmin Users, Hosts, UTM applications, UPIC connections and Collections.

• "WinAdmin Users"

The context menu of the "WinAdmin Users" node offers you the commands 'Show WinAdmin User List' to display a list of all WinAdmin users and (provided that you have the master privilege) 'New WinAdmin User...' which you use to create a new user.

Similarly, the 'Just Show Existing Privileges' command is only displayed if the current user has the master privilege. This menu command is used to define whether WinAdmin only displays the privileges that the WinAdmin users actually possess or whether all the privileges should be displayed.

The subtree below this node contains all the WinAdmin users in the configuration database together with the associated privileges, possibly subdivided on application and object type. For information on assigning privileges to WinAdmin users via this subtree, see section "Privileges for individual applications and object types" on page 128.

- "Hosts"

You can find all the hosts defined in the configuration database under this node. The context menu that can be called at this node contains the functions 'Show Host List' to open a list of all hosts and 'New Host...' which you use to create a new host in the configuration database.

In the tree structure, you can see the applications defined for each individual host and, furthermore, the UPIC connections that exist for each of these applications. The context menu for each host contains commands that allow you to generate a new host, create a new application for this host, delete the host or display the host's properties.

- "UTM Applications"

Here you can find all the UTM applications known to the configuration database. The node's context menu allows you to open a list of all the applications or generate a new application in the database. The associated UPIC connections are displayed below each individual application in the tree structure. The context menu for each application contains commands for creating a new application, generating a new UPIC connection for this application, deleting the application from the database, adding the application to or removing it from an open collection and for displaying the properties of the application.

- "UPIC Connections"

The subtree contains nodes for all the UPIC connections present in the configuration database. The context menu of the "UPIC Connections" node allows you to display a list of all the connections or generate a new connection. The context menu of every UPIC connection contains commands for generating a new connection, deleting the connection or displaying the properties of this UPIC connection.

- "Collections"

The subtree contains nodes for all the application collections present in the configuration database. You can use the context menu of the "Collections" node to display the list of all collections or create a new collection. The context menus for all collections contain commands allowing you to open the collection, create a new collection, import a WinAdmin V6.0 collection file, delete the collection or display the properties of the collection.

- "Display Filters"

This subtree allows you to access the currently defined display filters for all the object lists.

It contains one node for each object list. Below this are listed the filter criteria defined for each object list as well as the display filters that are constructed from the filter criteria. The context menu of the "Filter Criteria" node contains the 'Create Criterion...' command which you can use to open the "Edit Filter Criterion" dialog box in order to define a new filter criterion. The context menus of the nodes for the individual display filters contain the commands 'Select Filter' (only available if the corresponding list is open), 'Delete Filter...' and 'Properties'. For the individual filter criteria, the context menu also contains the commands 'New Criterion...', 'Delete Criterion...' and 'Properties'.

The context menu of the 'Display Filters' node contains the 'Just Show Existing Filtertypes' command. You can use this menu command to define whether WinAdmin is to display only the filter types for which at least one filter is currently defined, or all the filter types.

For information on defining and using display filters, see section "Display filters

" on page 53.

- "Open Windows"

Any open windows are displayed below this node. The context menu for each window contains the 'Show Window' (moves the selected window to the foreground) and 'Close Window' commands.

The context menu of the 'Open Windows' node contains the commands 'Close All Windows', 'Maximize All Windows', 'Cascade' and 'Tile Horizontally'.

- "Collection ..."

The navigation tree has as of version 6.1 only the subtree, "Administration" below the node "Collection '<name>'" (the "Generation" subtree was dropped). The "Administration" subtree gives access to WinAdmin administration functions. All functions available through this subtree use data from active UTM applications, and communication is realized via UPIC.

The subtree (may) possess the nodes 'Cluster' and 'Stand-Alone'. The UTM clusters present in the collection are listed under the 'Cluster' node (for more information, see section "UTM clusters and node applications in the navigation tree" on page 76). The standalone applications present in the collection are listed under the 'Stand-Alone' node.

The "Stand-Alone" subtree has the nodes "Applications" and "Objects" and, on the highest level, the leave "Statistics". The subtree under "Applications" provides a classical view of individual UTM applications, while the subtree under "Objects" provides a global view of UTM objects (Pterms, Users, etc.).

The leave "Statistics" and its functionality is described in a subsequent section.

The "Applications" subtree is structured as follows:

Host level

The computers running the various UTM applications of the collection are displayed here. This level contains the nodes "All Hosts", "<Host A>", "<Host B>", etc.

Application level

The individual UTM applications belonging to the corresponding host node are displayed here. This level contains the nodes "All Applications", "<Application A>", "<Application B>", etc.

Object level

The individual UTM object types that belong to the application node are displayed here. These include the leaves "Users", "Pterms" etc. This level is the extremity of this subtree and the leaves are used to access the individual object lists.

A key combination can be used to switch quickly between the four nodes (or five when a collection is opened) on the first level below the root of the navigation tree:

- Alt+ 0 (press and hold down the Alt key and press the 0 key)

Places the input focus in the collection window.

You can also do this by pressing the keyboard shortcut **Shift+F2** or by selecting the 'Focus Navigation Tree' command in the View menu.

- Alt+ 1

Opens and selects the first subtree ('WinAdmin Objects') and simultaneously closes the other subtrees.

- Alt+2

Opens and selects the 'Display Filters' subtree and simultaneously closes the others.

- Alt+3

Opens and selects the 'Open Windows' subtree and simultaneously closes the others.

- Alt+4
Opens and selects the 'Collection...' subtree and simultaneously closes the others.
- Alt+5
Opens and selects the 'Actions' subtree and simultaneously closes the others.

You can change the size of the navigation window by dragging the right-hand border. WinAdmin remembers the size that you define and sets the navigation window to the same size as when you exited the program on your next program session.


You can also hide the navigation window by pressing the keyboard shortcut **Ctrl+Shift+F2**. Pressing this keyboard shortcut again redisplayes it in its previous size. Selecting the 'Navigation Tree' command in the View menu has the same effect.

13.1 Layout of the Application and Cluster subtrees

Unlike earlier versions, WinAdmin as of version 6.1 allows you to modify the layout of the subtree of a UTM application (including node applications) or of a UTM cluster. The relevant settings are stored for each WinAdmin user in the configuration database and are consequently available in subsequent program sessions as well as to other WinAdmin instances that work with the same configuration database.

You can make separate settings for the Cluster subtree and for the Application subtree (standalone and node applications). These settings apply to all the Cluster and/or Application subtrees displayed in WinAdmin.

You make the basic settings in the "Settings" dialog box (see section "The 'General' sheet " on page 164). This dialog box contains the dropdown boxes 'Layout of the Application Subtree' and 'Layout of the Cluster Subtree'. You can choose between the following values here:

- 'Standard'
Corresponds to the familiar layout. All the entries used to access object lists, command mode etc. are arranged directly below the cluster or application node in the navigation tree.
This is the default setting.
- 'Standard With Categories'
If you choose this setting then some of the entries used to access object lists etc. are grouped together into categories which are displayed as subtrees identified by the icon  below the cluster or application node (category nodes). The following standard categories are available:
 - Security
Contains the entries "Security" (only in the Application subtree), "Ksets" and "Roles"
 - Statistics / Messages
Contains the entries "Statistics" and "Message Collectors"
 - UTM-D (LU6.1)
Contains the entries "Cons", "Lpaps" and "Lses".
 - UTM-D (OSI-TP)
Contains the entries "Abstract Syntaxes", "Access Points", "Application Contexts", "Osi Associations" (only in the Application subtree), "Osi Associations (generated)", "Osi Cons", "Osi Lpaps" and "Transfer Syntaxes".

- Printing (only in the Application subtree)
Contains the entries "Print Jobs (to confirm)" and "Printers".


- User Defined


Each user can modify the layout of the Cluster and Application subtrees in the navigation tree as they wish. The section below describes how you do this.

If a user-defined layout has previously been saved then this layout is loaded. Otherwise, the current layout is used as the basis for a user-defined layout.

13.1.1 Adapting the layout (user-defined layout)

You can adapt the layout easily by dragging and dropping the various elements to the required position.

You can move the individual entries in the subtree to a different position by left-clicking the icon corresponding to an entry and holding the mouse button down while dragging it to a different position. When you do this, the future position of the entry is displayed while you are dragging it. When you release the mouse button ("drop"), the entry is moved to the new position in the subtree and WinAdmin saves the current layout as a WinAdmin user-specific layout in the configuration database. If you drag an entry to a category node (with the icon ) then it is inserted as the last entry for this category in the category subtree. You can only move entries within your Cluster or Application subtree.

You can also rename categories. To do this, choose the 'Rename Category...' command in the context menu of a category node ( icon).

You can delete categories from which all the entries have been removed. To do this, choose the 'Remove Category...' command in the context menu of an empty category node.

It is also possible to create new categories. To do this, choose the Layout / 'Add Category' command in the context menu of the relevant cluster or application node. When you have entered the category name (which you can subsequently change at any time as described above), the new category node is displayed as the first node directly below the cluster or application node and can then be "filled" with entries using drag-and-drop.

Note:

You can make these changes to the layout whenever you want, i.e. even if no user-defined layout is currently set. These changes do not affect the standard layouts. Instead, the new layout is saved as a user-defined layout and overwrites any other user-defined layout that may already exist.

If you use drag-and-drop to modify the layout, you should always drag the **icon** corresponding to a node in the navigation tree to a new position and **not the name** of the node.

14 Object lists

(Note: The description of object lists contained in this section refers both to object lists for UTM objects and WinAdmin objects, i.e. WinAdmin users, hosts, UTM applications, UPIC connections and collections.)

UTM objects that can occur repeatedly in an application (e.g. User), as well as WinAdmin objects (WinAdmin users, hosts, UTM applications, UPIC connections and collections) are displayed in list windows in the same way as in the Windows Explorer, for example.

The objects are displayed in tabular form. Each line describes an individual object and each column indicates a property of the object. Some properties are not available for certain objects or applications, for example if a parameter is only relevant in BS2000 system and not for Unix/Linux applications. In such cases, WinAdmin displays the symbol "<.>", which means "not available".

14.1 Displaying object lists with WinAdmin objects

The object lists for the WinAdmin objects (WinAdmin users, hosts, UTM applications, UPIC connections and collections) can be displayed by selecting the relevant menu commands from the File menu:

- 'WinAdmin Users' menu command
Displays the WinAdmin user list.
- 'Hosts' menu command
Displays the hosts list.
- 'Applications' menu command
Displays the application list.
- 'UPIC Connections' menu command
Displays the UPIC connection list.
- 'Collections ' menu command
Displays the Collections list.

Alternatively, you can also open the lists of WinAdmin objects via the navigation tree. Five nodes for the four types of WinAdmin object are displayed below the "WinAdmin Objects" node on the top level of this tree. The context menus of each of these five nodes contain a corresponding 'Show ... List' command which you can use to open the corresponding object list. You can also single-click the named node to open the corresponding object list.

14.2 Displaying object lists with UTM objects

Object lists are displayed by simply activating one of the leaves on the navigation tree. Depending on the subtree in question, the objects of the desired type for a particular UTM application or for all applications on a certain host or in the collection or in a cluster are displayed.

The appropriate leaf on the navigation tree is activated either by clicking on it or by selecting it (using the keyboard or mouse) and then pressing the Enter key. In addition, the context menu of the node that represents the corresponding object list in the navigation tree contains a command that enables you to open the object list.

When you activate a leaf in this way, WinAdmin first checks whether a window with the required list already exists. If so, this is displayed; otherwise, a new window is created. If a number of different lists are open, you can use the keyboard shortcuts **Alt+F9** (previous window) and **Alt+F10** or **Ctrl+F6** (next window) to switch between them. Alternatively, you can use the 'Previous Window' and 'Next Window' commands in the Window menu. In addition, all the open lists are indicated in the Window menu and can be selected by choosing the corresponding menu command.

To save you from searching for an open window which may possibly already be open, you can also access it via the navigation window: All the windows that are currently open are listed under the 'Open Windows' node. You can single-click the window node to select and display the corresponding window.

Administration objects of the Osi Association type are an exception to this rule: in this case, the navigation tree contains two leaves for each of these objects, and two object lists are displayed accordingly, namely a list of generated Osi Associations and a list of established Osi Associations. The properties of the objects can only be displayed for the established Osi Associations. The names that appear in this list correspond to the Association IDs assigned by XAP-TP and cannot be assigned to the names generated in openUTM.

Lists of Queue type objects form another exception (service-controlled queues): Alongside the temporary queues generated by the UTM program units, the list for queue objects also contains all the Tac queues (objects of type Tac with TYPE=Q) as well as all the user queues (objects of type User with QLEV > 0). All the parameters for these objects that are relevant for queues are displayed.

14.3 Layout of an object list window

An object list window is basically subdivided into three parts:

The object list itself is displayed in the center. This consists of a column header and column titles as well as a row for each object to be displayed.

Above and at the left of the object list, you can select the column set that is to be displayed in the list. On the right, you can select an existing display filter or create a new display filter:



The individual items have the following functions:

- (1) 'Column Set' button
Clicking this button opens a context menu containing the item 'Proposal'. If you choose this menu item then a suggested layout for the columns as column sets is displayed and applied.
- (2) 'Column Set' dropdown box

You can select one of the defined column sets in this dropdown box.

- (3) 'Next Column Set' button
Moves to the next defined column set. If the last column set has been reached then this button returns you to the first defined column set.
- (4) 'Display Filter' button
Clicking this button opens a context menu in which you can create new display filters or delete or edit existing ones.

If the context menu contains the items 'Delete Filter...' and 'Properties' then these commands apply to the display filters that are displayed in the 'Display Filters' dropdown box.
- (5) 'Display Filter' dropdown box
Here you can select a defined display filter.
- (6) 'Deactivate Display Filter' button
Clicking this button deactivates the currently selected display filter. The entry <no filter> then appears in the 'Display Filter' dropdown box

Further explanations of items (4), (5) and (6) can be found in the section "Layout of an object list window" on page 48.

Below the object list, there is a line in which you can filter the objects that are to be displayed (see section "Entering a filter text" on page 50) and, below this, a row of buttons which can be used to run list functions (see section "The lower button bar" on page 56).

14.4 Filtering list objects

There are many different ways of filtering the objects that are to be displayed in an object list. Compared to earlier WinAdmin versions, WinAdmin as of version 6.1 provides more ways of filtering list objects. One quick way of performing filtering is to enter a filter text in the 'Filter Text' box below the object list. Another way is to define column filters using the buttons in the column header. A final type of filtering is familiar from older WinAdmin versions, i.e. the definition of complex display filters. These three filtering methods are described in more detail below.

If more than one type of filter is active then the list objects are only displayed in the list if they satisfy **all** the activated filters.

14.4.1 Entering a filter text

Entering a filter text is the easiest and fastest way of performing filtering.

Below the list there is usually a filter bar which you can use to filter the contents of the list. All the list objects are filtered on the basis of the entered filter text and defined options. Only those objects that meet the current filter criteria are then displayed in the list. When you modify the filter text or options, the displayed list contents change immediately to match the new settings. This constitutes a quick way of gaining an overview of lists that contain a large number of objects.

By default, all cell contents displayed for the list objects are taken into account. However, you can also specify in more detail which list columns (and therefore which object properties) should be taken into account during filtering.

An object is displayed in the list even if the content of only one of its cells matches the current filter criteria.

Filter criteria

The filter is primarily determined by the entered filter text.

You can also specify whether or not filtering is to be case-sensitive.

In addition, you can define how the entered filter text is to be interpreted. You can choose between the following filter types:

- Simple Text
A simple search is performed for the entered text.
- Regular Expression
The filter text is interpreted as a regular expression.
- Complex Filter
The filter text is interpreted as a complex filter mask (see → "32.3 Complex filter masks").
In this case, the case-sensitivity setting is ignored since this filter mask option has to be specified in the filter text.

By default, only the text box for the filter text is displayed in the filter bar. In this case, a simple text search is performed and the search is not case-sensitive.

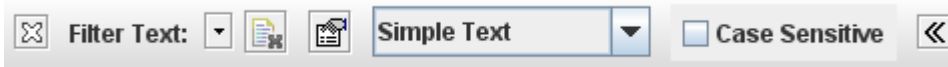
The controls allowing you to select the filter type and specify whether a case-sensitive search is to be performed are only displayed if you request them in order to ensure that the appearance of the filter bar is not too complex when these items are not required.

Layout of the filter bar

As mentioned above, there are two different layouts for the filter bar. In the simple layout, it has the following appearance:



You can use the arrow button **»** at the right-hand end to display the extended filter bar:



If the extended layout is displayed, you can click the arrow-button **«** at the right-hand end to return to the simple filter bar layout.

You can hide the filter bar by clicking the cross **✖** at the left-hand end of the bar. If the object list has the input focus then the filter bar can be displayed again using the keyboard shortcut **Ctrl+F**.

You enter the filter text in the text box labeled 'Filter Text'.

Immediately to the right of this text box there is a small downward-pointing arrow **▼**. If you click this then a context menu may be displayed in certain situations (see → "Context menu").

Clicking the **📄** button next to the arrow clears the text box. When you click this button, the complete list contents are again displayed unfiltered.

The **📊** button next to this opens a dialog box indicating all the currently displayed list columns. Here you can select the list columns that are to be displayed when filtering is performed. The Column context menu for a list column contains the 'Consider When Filtering' command which can also be used to determine whether or not the corresponding list column is to be taken into account during filtering.

You can use the dropdown box to select the filter type (see → "Filter criteria").

In the 'Case Sensitive' checkbox, you can specify whether or not filtering is to be case-sensitive (see → "Filter criteria").

Filter operation

If you enter a filter text in the corresponding box then the content of the list is updated immediately, i.e. on every modification to the filter text. During the filter operation, the background of the text box is displayed bright yellow to indicate that the filter operation has not yet terminated. However, this is usually only visible in lists with a large number of objects.

If no list object matches the current filter criteria then the background of the text box is displayed bright red.

Context menu


The text box for the filter text possesses a context menu that is also displayed when you click the small arrow to the right of the text box.

You can use the commands in the context menu to save the current filter criteria. You can also reload filter criteria that you have stored in the past. The saved filter criteria include the filter text, the filter type and the case-sensitivity setting.

There is also a command that allows you to delete the entered filter text.

The context menu is displayed either if the text box for the filter text is not empty or if stored filter criteria already exist.


14.4.2 Column filters

To activate column filters, single-click the icon  in the corresponding columns of the column header. This displays a dialog box in which you can define the column filter.

In the top part of the dialog box, there are two buttons below the title 'Switch-off Column Filter'. You can use these buttons to deactivate active column filters. If you click the 'Only This Filter' button then only the column filter that is displayed in the current dialog box is deactivated. Clicking the 'All Filters' button deactivates all the active column filters in the list. The dialog box is closed as soon as you click one of these buttons.

Below these, you can see all the values currently present in the selected list column. By clicking the checkboxes corresponding to these entries, you can specify which values are to be permitted in the column and which not. If the box is checked then the corresponding value is permitted. The number of list objects in the list column that possess the corresponding value is indicated in brackets in front of the checkbox. The 'Sort by Used-Frequency' checkbox allows you to specify whether the individual values are to be sorted based on their frequency of occurrence (box checked) or alphabetically (box not checked).

You can use the 'Select All' checkbox to select or deselect all the values.

If you click 'OK' to exit the column filter dialog box then the column filter for the corresponding column is activated. The icon  is then displayed in the column header of this column. If you hold down the **Shift** key when you click on this icon then the column filter for this column is deactivated.

In this way, you can also define and activate multiple column filters. If multiple column filters are active then only those list objects that satisfy all the column filters are displayed.

Column filters are "transient", i.e. if a list is closed then the column filters defined in it are no longer present when the list is opened again and they have to be redefined.

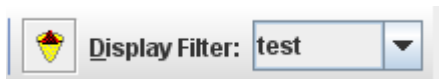
The column filters dialog is also a very good way of gaining an overview of the column values that are present in the list. This is particularly useful if the corresponding column contains a bipolar data type (e.g. 'On'/'Off', 'Yes'/'No' or 'Established'/'Not Established' etc.).


14.4.3 Display filters


In addition to the above-mentioned ways of selecting the objects that are to be displayed in an object list, WinAdmin still allows you to define display filters that can be permanently stored in the configuration database and activated as required. However, WinAdmin as of V6.1 no longer supports name filters. One good substitute method consists of entering a filter text below the object list.

It does, nevertheless, continue to support the complex display filters that can be used to specify which objects should be displayed on the basis of their properties. These filters are able to take account of all the properties of the various object types. As indicated above, the complex display filters defined by users are stored in the configuration database and can be activated whenever they are required.

You can use the following text boxes above the object list to activate, create, modify or delete display filters:



You activate an existing display filter in the 'Display Filter' dropdown box. The dropdown box also contains the entry '<no filter>'. If you select this entry then no display filter is active in the object list. You can also activate '<no filter>' by clicking the icon  to the right of the dropdown box.

The icon  is displayed to the left of the dropdown box. You can click this icon to open the context menu for the display filters. This contains the following commands:

- 'New Filter...'
This command is always present and allows you to define a new complex display filter.
- 'Delete Filter...'
This command allows you to delete the complex display filter that is displayed in the dropdown box (and is currently active). This command is not available if the dropdown box is displaying the entry '<no filter>'.
This command is not available if the dropdown box is displaying the entry '<no filter>'.
- 'Properties'
This command allows you to edit the complex display filter that is displayed in the dropdown box (and is currently active). This command is not available if the dropdown box is displaying the entry '<no filter>'.

The complex display filters are described in more detail below.


Complex display filters

Complex display filters consist of one or more filter criteria (maximum ten). A filter criterion always applies to precisely one object property. If the filter consists of more than one criterion then you can specify whether the individual criteria are to be combined by ANDs or ORs. If you combine them using AND, the object is only displayed if all its object properties fulfill all the specified filter criteria. If you combine them using OR, the object is displayed if at least one of the specified filter criteria is fulfilled.

The individual defined filter criteria are stored in the configuration database and can be combined to form new complex display filters as required. The complex display filters assembled in this way are also stored in the configuration database under the name specified by the user. This makes it

possible for users to gradually construct a collection of extremely powerful complex display filters which can be used to select the objects for display.

You can also access the filter criteria for all the object lists and the complex display filters assembled from them via the 'Display Filter' node on the topmost level of the navigation window (see also "Display Filters" in section "The navigation window" on page 42).

If an object list is active then you can create, edit and delete filters in the menu which is displayed when you click the  icon (see above).

If you select the 'New Filter...' command, the 'Create Filter' dialog box is opened and if you select 'Properties', the 'Edit Filter' dialog box is displayed. Both dialog boxes have the same structure:

The 'Create/Edit Filter' dialog box

This dialog box enables you to create and edit complex display filters.

- 'Name' text box

Here you enter a name for the complex display filter.

- 'Combine Using AND' / 'Combine Using OR' radio buttons

If you have "checked" a number of different filter criteria then you can use these two radio buttons to specify whether these filter criteria are to be combined using AND or OR (see section "Complex display filters

" on page 53). If one filter criterion is checked, then both options will yield the same result.

- Table containing the filter criteria

This table lists all the filter criteria that exist for the relevant object type. It contains two columns:

- In Filter

Specifies whether or not the filter criterion is present in the current complex display filter. You can check or uncheck the box in this column to add the criterion to or remove it from the display filter.

- Name

Contains the name of the filter criterion.

Below the table, you will see the following buttons along with other controls:

- 'Add' button

Clicking this button creates a new filter criterion. It also opens the 'Create Filter Criterion' dialog box (see below) in which you define the new criterion.

- 'Edit' button

If exactly one filter criterion is selected in the table (not to be confused with "checked") then you can click this button to edit the filter criterion. The 'Edit Filter Criterion' dialog box is displayed (see below).

- 'Remove' button

Clicking this button removes the selected filter criterion. The button is only active if the filter criterion is not present in a complex display filter, i.e. more specifically if it is not currently checked".

The 'Create/Edit Filter Criterion' dialog

With this dialog, you can edit an existing filter criterion or define a new one. Description of the individual dialog fields:

■ 'Name' section

The name of the filter criterion is specified in this section. This can either be done automatically (radio button 'automatic') or manually (radio button 'user-defined').

- 'automatic' radio button

If this radio button is selected, WinAdmin creates a generic name for the filter criterion. This generic name describes the properties of the filter criterion and is adjusted automatically when the properties are changed in other parts of the dialog. The current name is displayed in the 'Name' edit box. In this case the edit box is set to "read only".

- 'user-defined' radio button

If this radio button is selected, you must specify a name for the filter criterion in the 'Name' edit box. The name has to be unique among the names of the filter criteria for this object type.

- 'Name' edit box

The name of the filter criterion is either displayed (see radio button 'automatic') or must be entered (see radio button 'user-defined') in this edit box.

■ 'Object Property' section

Filter criteria always apply to precisely one object property of an object type. The dropdown box contains a list of all object properties of the current object type. Click on and select the object property to which the filter criterion is to apply. The object properties in the list are identical with the column headings of the associated object list. The entries in the dropdown list are in alphabetical order.

■ 'Comparison' section

The fields in this section box specify how an object fulfills the filter criteria. An object meets the filter criteria when the comparison specified in this group box matches the actual value of the corresponding object property.

In order to define a comparison, you can select one of five different logical comparison types using the radio buttons. Each radio button has one or two edit boxes in which the value or values for the comparison can be entered.

- 'equal' radio button

If this radio button is selected an object fulfills the filter criterion if the value of the object property in question is equal to the value specified in the "Value' edit box.

- 'unequal' radio button

If this radio button is selected an object fulfills the filter criterion if the value of the object property in question is not equal to the value specified in the "Value' edit box.

- 'less' radio button

If this radio button is selected an object fulfills the filter criterion if the value of the object property in question is smaller than the value specified in the "Value' edit box.

- 'between' radio button

If this radio button is selected an object fulfills the filter criterion if the value of the object property in question is between the values specified in the two 'Value' edit boxes.

- 'greater' radio button

If this radio button is selected an object fulfills the filter criterion if the value of the object property in question is greater than the value specified in the "Value' edit box.
- 'Interpret value as' section

In the fields of this section you specify how WinAdmin is to interpret the values entered in the 'Comparison' group box and the actual values of the object properties.

 - 'Text' radio button

If this radio button is selected WinAdmin interprets the values as character strings. In this case further settings can be made with the 'Interpret '*' and '?' as wildcards' and 'Case sensitive' check boxes.
 - 'Interpret '*' and '?' as wildcards' check box


If this box is checked, WinAdmin interprets the characters "*" and "?" as wildcards if they are contained in the comparison values (see above).
 - 'Case sensitive' check box


If this box is checked, the comparison will be case-sensitive. In this case, the comparison is more powerful than if it were not case-sensitive.
 - 'Number' radio button


If this radio button is selected WinAdmin interprets the values as numbers. This is important, for example, in smaller than/larger than comparisons. If, for instance the values 10 and 9 are compared as numbers, then 10 is larger than 9, but if they are interpreted as character strings, then "9" is larger than "10", because the first character in the strings is compared first and, in ASCII code, the value "9" is greater than the value "1".


14.5 The lower button bar


A row of buttons can be seen at the lower edge of the object list window:

- Button 

Permits the automatic setting of width of all the columns present in the list.
- Button 

Increases the font size used in the object list and the column header.
- Button 

Reduces the font size used in the object list and the column header.
- Button 

Displays the object list's properties dialog.
See the section "The property dialog in the object list" on page 57.
- Button 

Displays the print preview window for the object list from which you can print the object list.
- 'Add' button


Allows you to create new objects of the same type as those present in the object list. This button is not present in all object lists.

- 'Edit' button
Allows you to edit (modify) the object or objects that are selected in the object list. The property dialog for the selected objects is displayed.
- 'Remove' button
Allows you to remove the object or objects that are selected in the object list. This button is not present in all object lists.
- 'Refresh' button
Allows you to refresh the content of the object list. This button is not present in all object lists.
- 'Help' button
Opens the online Help system.

Other list-specific buttons may also be present below the object list.

14.6 The property dialog in the object list

Each object list has its own property dialog in which you can set all the properties for the list. The settings are saved in an object type-specific and WinAdmin user-specific form in the configuration database and are automatically re-used the next time the object list is opened.

You can open the property dialog by clicking the button  below the object list. This contains a number of different sheets that are described below.

14.6.1 The 'Content' sheet

This sheet contains information about the content of the object list.

- 'Data Source' text box
Here you specify the source for the list data, i.e. the "name" of the configuration database in the case of WinAdmin objects or the "active UTM applications" in the case of administration objects.
- 'Object Type' text box
Specifies the object type for the list data.
- 'Available Objects' text box
Specifies the total number of objects present in the list. If a display filter is active then objects which are not present may also be displayed.
- 'Displayed Objects' text box
Specifies the number of objects actually displayed in the list. If a display filter is active then this number may be smaller than the number of available objects.
- 'Selected Objects' text box
Specifies the number of objects currently selected in the list.
- 'Display Filter Active' text box
Specifies whether a display filter is currently active or not.

- 'Applications' list

This list is displayed only for UTM object. The list lists all the UTM applications whose objects are present in the list (or should be present - the applications may be temporarily unavailable). Alongside the names of the applications and hosts, any UTM clusters to which the UTM applications belong are also displayed.

14.6.2 The 'Display Filters' sheet

This dialog sheet allows you to activate and deactivate display filter. For more information see section "Display filters

" on page 53.

14.6.3 The 'Columns' sheet

This sheet contains a list which lists all the available list columns for this object type.

It is possible to give individual list columns a user-defined name ("alias"). Such names are then used instead of the default names in the column headers of the corresponding object lists. For example, you might decide that the column name "Application Entity Qualifier" is too long. In this case, you might assign the alias "AEQ" to the column in order to improve the clarity of the object list.

In addition, you can assign conditions to the individual list columns. These specify the conditions under which the column is to be included in the list. This is sensible, for example, when the content of a column is not available for all UTM applications on all operating system platforms. Thus the 'Listener-Id' property of a UTM Bcamappl is not available under BS2000 systems although it is under Unix/Linux or Windows systems. In this case, you could assign the condition 'Unix or Windows involved' to the column. The column would then be included in all Bcamappl lists which contain a UTM application under Unix or Windows. In the case of a pure BS2000 list, the column would be absent.

The column properties are set in a separate dialog. This is displayed when you double-click a column name in the list or choose the 'Properties' command in a column's context menu (right-click on the column name). You can enter your own choice of name for the column in the dialog box's edit box. You can use the 'Condition' dropdown box to choose the display condition for the column.

Note:

The column properties you set here apply to all lists for the object type. They are automatically used when you open another list of this type. Furthermore, the column properties are stored for each user individually in the configuration database. This has the advantage that two different WinAdmin users can set different column properties for the same object type.

14.6.4 The 'Column Sets' sheet

WinAdmin allows you to define multiple so-called "column sets" for an object list type. This is a particular advantage in the case of object types which possess many object properties (and consequently many list columns in the Detail display mode). When viewing such object types, it would be necessary to scroll horizontally repeatedly in order to see all the properties of the objects. However, when you do this, the object name, which is usually located in the first column in the list, disappears from the screen and it is very difficult to match the displayed data to the individual objects.

This problem can be overcome by constructing multiple column sets for an object type. For example:

Object type A has the columns:

- Name, Application, Host, Property1, Property2, ..., Property20.

On the basis of this, it is possible to create, for example, three different column sets. You first assign each column set the columns Name, Application and Host together with a subset of the actual attributes:

- Set 1: Name, Application, Host, Property1, Property2, ..., Property7

- Set 2: Name, Application, Host, Property8, Property9, ..., Property13

- Set 3: Name, Application, Host, Property14, Property15, ..., Property20

Each individual object property can be present in one or more column sets. It can also be entirely omitted. The column sets can be defined in such a way that they can be generally be displayed in full on screen. This removes the need to scroll horizontally. Since the object name can be included in all the column sets, it is very easy to establish the correspondence between the list data and the individual objects. When the object list is active, you can switch between the individual column sets by pressing **CTRL+L** or by using the dropdown box above the object list.

You can also use the 'Column Sets' sheet in the property dialog to rename column sets, create new ones, delete existing ones and modify the column set sequence. You do this using two lists. You execute the individual functions from context menus that are displayed when you right-click on individual cells in the lists.

You can also access this property dialog sheet directly by choosing the 'Edit Columns' command in the column header's context menu.

■ 'Column Sets' list

The names of all the currently defined column sets and their sequence are displayed here. The sequence of the column sets is followed when you "page through" the sets using the **CTRL+L** key combination. The context menu for a column set is displayed when you right-click on the name of the corresponding set and contains the following commands:

- 'Forward'
Moves the column set forward in the sequence.
- 'Backward'
Moves the column set backward in the sequence.
- 'Delete'
Deletes the column set. You cannot delete the last remaining column set.
- 'Rename'
Allows you to rename a column set.
- 'New Set'
Generates a new column set. You can assign the name immediately after choosing the menu command. You define the object properties that are to be assigned to the set in the 'Order of the columns in the sets' list.

■ 'Proposal' button

If you click this button, WinAdmin suggests how the existing columns could be assigned to the various column sets. You can then click 'Apply' or 'OK' to accept this suggestion.

■ 'Order of the columns in the sets' list

You use this list to define which object properties should appear in what order in the individual column sets. Each row in the list represents an object property (list column) that has been

defined for the object type. The first column of the control lists the names of the object properties. The other columns in the control each represent a defined column set. The numbers in the cells of these columns indicate the position of the corresponding object property in the column set. An empty cell indicates that the corresponding object property is not present in the column set.

The functions of this control are also available via context menus. Each cell that belongs to a defined column set has its own context menu. The available functions apply to the corresponding column set and the associated object property that is represented by the cell. The cells in the **first column**, which have the name 'Columns', have **no** context menu.

If you sort the content of the list on the basis of the sequence of object properties within a column set by clicking on the corresponding column header, you will often find that the result is a lot clearer than when you sort the list on the names of the object properties.

The context menus offer the following functions:

- 'Forward'
Moves the object property forward one position in the corresponding column set.
- 'Backward'
Moves the object property backward one position in the corresponding column set.
- 'Remove'
Removes the object property from the corresponding column set.
- 'Add'
Adds the object property to the corresponding column set. The property is initially positioned at the end of the column set.
- 'Default'
Default allocation of the column set. This consists of all the object properties in the sequence defined by WinAdmin.

You can also use drag-and-drop to modify the sequence of object properties within a column set. To do this, click in a cell in the corresponding column set and use the mouse to drag it onto any other (occupied) cell in the same column set. If you now release the mouse button, the "dragged" object property is positioned at the location specified by the target cell. The action performed when you "drop" the property is displayed below the list during the "drag" operation.

In particular when modifying the column sequence via drag-and-drop, you are highly recommended to sort the object properties in the corresponding column set on the basis of their sequence (by clicking the corresponding column header).

You can also modify the sequence of the individual columns in the lists themselves. To do this, simply use the mouse to drag the column header of the required object property to a different position.

14.7 User-defined layout

The layout of the object lists, i.e. the sequence of individual columns can now be adapted to meet the user's individual requirements. It is also possible to change the width of the columns by dragging the boundary line between two column headers. It is also possible to move a column within the list by dragging the column title to the required position with the left mouse button held down (drag-and-drop).

The column header with the column titles contains a context menu which allows you to make the following settings:

The following settings can be defined using the context menu for the column headers:

- 'Move Column Left' and 'Move Column Right'
To move the column one position to the left or right respectively.
- 'Remove Column "<name>"'
To remove the column.
- 'Best Fit'
To adapt the column width to the contents of the column. A submenu contains the items 'This Column', 'All Columns', 'This Column (without Header)' and 'All Columns (without Header)'. WinAdmin sets the column width of the current column or of all columns to the minimum width that allows the complete column contents to be displayed (including or excluding the column header).

It is also possible to adapt the columns and column sets via the object list's property dialog (see section "The property dialog in the object list" on page 57).

An object list layout that has been defined in one of the ways described above is automatically saved user-specifically in the configuration database when the object list is closed. The next time an object list of the same object type is opened, it is loaded again and used for display purposes.

14.8 Sorting objects

Object lists can be sorted simply and quickly according to various criteria. Simply click on the appropriate column header. The list is then sorted in ascending or descending order on the basis of this column. The column currently used to sort the list is marked with a ▲ or ▼ icon below the column header. ▲ represents ascending and ▼ descending order. If you click on an unmarked column, this column is first sorted in its standard order (this may be ascending or descending, depending on the property displayed in the corresponding column). Clicking on a marked column reverses the sort sequence.

Using this mechanism, all connected users or all users with the same start format can be displayed in succession in a Users list, for example.

14.9 Context menus for the list objects

You can select individual or multiple objects from an object list using standard Windows methods. For example, simply click on the object to select it. Multiple objects can be selected by using the Shift or Control key. And selections can also be made using the keyboard. The selected objects are highlighted in color. All of this is familiar from the Windows Explorer, for example.

A click on the selected object (or one of the selected objects) with the right mouse button opens a context menu.

The context menus contain the following menu commands for lists with UTM objects.

- 'Add to Object Selection' command
This menu command adds the selected object(s) to the corresponding selection of the currently opened collection.
- 'Properties' command
This command opens the associated property dialog. It can also be opened by double-clicking the object.
- Create <object type> command
This command is only available for dynamically creatable object types (Con, Kset, Lses, Ltac, Lterm, Pterm, Program, Tac, User). Furthermore, it is only available if just one object is selected. The menu command opens a property dialog for generating a new object of the type specified.).
- 'Delete <object type> command
This command is only available for dynamically deletable object types (Con, Kset, Lses, Ltac, Lterm, Pterm, Program, Tac, User). It opens the dialog box 'Delete Object(s)' where the deletion needs to be confirmed again, before it is actually carried out. In the case of user objects, you can choose between delayed deletion and immediate deletion.
You cannot delete an object as long as it is referenced by another object.
- 'Send Message...' or 'Send Message to assigned Lterm...' command
This function is only available for Lterm and User objects if the relevant UTM application is running on a BS2000 computer. You can enter a message in a dialog box and the message is issued in the system status line of the relevant dialog terminal. Users have to be connected with a terminal at the time 'Send Message' is issued.
- 'Spoolout...' command
This command is only available for Lterm objects. You can establish a connection to a specific printer using this menu command (however, this only makes sense in connection with printer Lterms).
- 'Update IP-Address...' command
This menu command is only available for objects of the types Con, Osi Con and Pterm. It causes openUTM to update the IP address of the selected partner with the data from the host name database (only makes sense for TCP-IP-APPLI protocol type or for RFC1006 with Unix, Linux or Windows systems).

In addition, the context menus of navigation tree nodes that represent a Tac list of a UTM cluster or a standalone UTM application contain the 'Reset All TAC Statistics' command. This command allows you to reset the values of certain counters for all the Tacs present in the list to zero. You can do this for the following six values for each Tac:

- 'Used'
- 'Errors' (number of incorrectly terminated program unit runs)
- 'Elapsed Time (msec)' (average elapsed time per program unit run)
- 'Avg. CPU Time (msec)' (average CPU time per program unit run)
- 'DB-Counter' (average number of database calls per program unit run)
- 'DB Elapsed Time (msec)' (average elapsed time for database calls per program unit run)

These statistics are displayed on the 'Current' sheet of the Tacs property dialog. Here you can also reset these counters for specific Tacs by checking the corresponding 'reset' boxes.

Furthermore, the context menu of a navigation tree node that represents a Tacclass list (Tacclasses) of a UTM cluster or a stand-alone UTM application, contains the additional command "Reset All Tacclass Waittimes". By means of this command, the statistic value "Average Waittime (msec)" is reset for all Tacclasses contained in the list.

This statistic value is displayed in the property dialog of a Tacclass on the "General" sheet. This statistic value for a specific Tacclass is reset by checking the box "reset".

For lists of WinAdmin objects, the context menu offers the following commands:

- 'New Host...' / 'New Application...' / 'New Upic Connection...' / 'New WinAdmin User' commands
You can open a property dialog allowing you to create a new object of the specified type using this menu command. The fields of the dialog box contain the parameters of the selected "source object".
- 'Delete 'WinAdmin User...' / 'Delete Host...' / 'Delete Application...' / 'Delete Upic Connection...' commands
If you select this menu command, a dialog box will be opened in which the deletion of the selected object has to be confirmed before it is actually carried out.
- 'Properties' command
This menu command opens a property dialog in which the properties of the selected object are displayed and can be modified.

14.10 Data refresh

An active window can be refreshed by pressing the F5 key. The 'Refresh' button below the object list has the same effect. All of the data in this window is then retrieved again from the UTM applications and the screen contents are refreshed.

14.11 Linked objects

Some UTM objects are linked with other objects (e.g. a user can possess a Kset or be linked to an Lterm/Pterm). Information on the properties of these linked objects can be obtained by the context menu of the linked object. For each linked object, a submenu is appended to the context menu. This contains the complete context menu for the linked object.

14.12 Definition of partial lists

Many object lists of a UTM application can contain a very large number of objects. For example, this is frequently the case for user or Lterm lists. In some cases it can take a very long time to retrieve the necessary data. The filtering of the objects is implemented in WinAdmin itself and not in UTM (post-selection), with the result that it is not possible to restrict the number of objects retrieved by performing a pre-selection in WinAdmin.

However, the objects are passed by the UTM administration interface in lexicographic sequence and the caller of the interface can specify the name that the first passed object is to have. In WinAdmin as of V6.1, this procedure is used to request partial object lists from UTM. An object partial list is defined by one or more name ranges. Only the objects whose names fall within these name ranges are requested from UTM and displayed in the corresponding list. Thus, for example, you can define a partial user list with the range "K-M" in which only those users whose names start with the letters K, L and M are displayed (and retrieved). A partial list defined with the range "BR-BR", for example, would contain all those users whose names start with the letters "BR".

The names returned by UTM are ordered lexicographically and it is important to note that the sequence is dependent on the underlying character code used by the UTM application. In the case of UTM applications under BS2000 systems, the code used is EBCDIC and, in the case of UTM applications that run under Unix, Linux or Windows systems, it is ASCII. Since WinAdmin also runs under Unix, Linux or Windows systems, it uses the ASCII code when sorting names.

The three character ranges that are important for object names are uppercase letters (A-Z), lowercase letters (a-z) and numbers (0-9). Within these three ranges, the sequence of the characters is the same in both character codes. However, these three ranges themselves are ordered differently in the two character codes. In the ASCII code, the numbers come before the uppercase letters and the uppercase letters before the lowercase letters. In EBCDIC, however, the lowercase letters come before the uppercase letters and the uppercase letters before the numbers.

ASCII: 0-9, A-Z, a-z
EBCDIC: a-z, A-Z, 0-9

It is necessary to take this difference into account when specifying name ranges.

Example:

If, for example, you define a name range "-A" in WinAdmin, then in the case of applications that run under Unix, Linux or Windows systems, all the objects whose names start with A and all objects whose names start with numbers are returned because, in ASCII code, the numbers come before the letter A. In the case of an application running under BS2000 system, all the objects whose names start with the letter A are also returned. However, all objects with names beginning with lowercase letters are also returned because in EBCDIC, the lowercase letters precede the letter "A". In contrast, objects whose names start with a number are not returned because, in the lexicography of the EBCDIC code, the numbers come after the letter "A".

14.12.1 Adding a new partial list

In WinAdmin, partial lists are displayed as nodes below the complete list. By default, there are initially no partial lists. The context menu of an object list node contains the command "Add Partial List". Choosing this command opens a dialog box in which you have to define the new partial list (e.g. "A-M"). When you click "OK" to close this dialog box, WinAdmin adds a node representing the partial list below the node corresponding to the associated complete object list in the navigation tree.

The dialog box also allows you to enter several different name ranges separated by commas:

```
<range1>[, <range2> [, ..., <rangeN>]]
```

You can enter these name ranges in the following ways:

```
<from-prefix>-<to-prefix>   or
-<to-prefix>                  or
<from-prefix>-               or
<prefix>-
```

Examples:

D-F: All names that start with D, E or F.

BR-BT: All names that start with BR, BS or BT.

-K: All names that start with K or any other character that comes before K in the character code.

M-: All names that start with M or any other character that comes after K in the character code.

TEST: All names that start with TEST.

The character code used in the UTM application (BS2000: EBCDIC, otherwise ASCII) determines the sequence of uppercase characters, lowercase characters and numbers (see above).

WinAdmin saves partial list definitions in the configuration database separately for each application so that they can be automatically added to the navigation tree in future program sessions.

14.12.2 Removing a partial list

The context menu of a partial list node contains the command "Remove Partial List". If you choose this command then the definition of the partial list is removed from the navigation tree without any confirmation query. The entry in the configuration database is deleted.

15 Property dialogs

Property dialogs can be displayed for UTM objects. These property dialogs always contain all of the properties of the corresponding objects supplied by the administration program interface and cannot be parameterized. All properties which cannot be modified are displayed as static text. Properties which can be modified are displayed either as dropdown boxes with the available values or as edit boxes in which you can enter values.

The property dialogs contain one or more sheets which can be opened individually using normal methods. Each property dialog has a "General" sheet containing the most important properties of the object, and may have one or more additional sheets with additional properties. If a UTM object contains links to other UTM objects (e.g. a user is linked with an Lterm/Pterm), the "General" sheets of these linked objects are also included in the property dialog. However, no changes can be made in these sheets.

In addition, there are always buttons beside the names of the linked objects which allow you to call the property dialog of each linked object directly.

The property sheets contain a 'Refresh' button. If you click this button, the content of the property dialog is refreshed, i.e. the current values are again polled from the UTM applications..

If the object displayed in the property dialog is referenced by one or more other objects then the dialog also contains a 'Referenced By...' sheet. This lists the referencing objects together with their names, types and application, host or cluster names.

15.1 Displaying the property dialog for an object

Simply double-click on the name or associated icon of an object displayed in a window, e.g. an object list. Alternatively, right-click the object and select the 'Properties of...' command from the context menu that then appears.

15.2 Displaying multiple objects in a property dialog

If you have selected multiple objects in a window (see above), you can call up a property dialog for these objects which presents all the selected objects together. To do this, simply right-click one of the selected objects. A context menu appears with the 'Properties of selected Objects...' command.

If you select this menu command, WinAdmin combines all the selected objects. The properties are thus compared with each other (if the values are not the same for all objects, "<mixed>" is displayed), or they are added (e.g. the number of used Tacs for users), or mean values are formed (if the property itself represents a mean value).

Amongst other things, the property dialog provides a good overview of the properties that are common to all selected objects. This means that a property can be changed for multiple objects at the same time. For example, the status of a number of users can be set to "On" simultaneously, regardless of whether or not the individual users already had the status "On". To do this, the value for Status is simply changed from "<mixed>" or "Off" to "On" in the property dialog.

A dialog in which multiple objects are displayed does not possess a "Referenced By..." sheet.

15.3 Modifying object properties

Provided openUTM permits the modification of an object's properties, these properties can be modified using the WinAdmin property dialog described above. If a property can be modified, it is displayed as an edit box or as a drop-down box. You then simply need to enter or select the new parameter values and click on 'OK' or the 'Apply' button.

In general, the syntax of any input in edit boxes must comply with the rules of the KDCDEF configuration tool.

Some of the modified parameters are checked locally by WinAdmin; otherwise they are checked by the UTM application(s). If a modification is rejected by WinAdmin, the values last entered in the property dialog sheets are retained so that they can then be corrected. To redisplay the applicable values in the UTM application or the configuration database, simply click on the 'Refresh' button.

If multiple objects are presented together in the property dialog, a modification job for each of the objects is created internally when you click 'OK' or 'Apply'. It may happen that the modification is implemented for some of the objects and not for others.

16 Creating and deleting objects

WinAdmin V6.4 allows you to create and delete the following objects that are to be administered dynamically: Con, Kset, Lses, Ltac, Lterm, Program, Pterm, Tac and User. In order to create new administration objects, corresponding table entries must be reserved when generating the UTM application (RESERVE statement).

New objects can be created as follows with WinAdmin:

- Right-click an application name or an 'All Applications' node in the navigation window and
Select the 'Create' menu command and the desired object type in the context menu which is then displayed.
- Right-click an object list name in the navigation window and
Select the 'Create <object_type>' command in the context menu which is then displayed.

A property dialog is then opened, which is similar to the property dialog described above for existing objects. Here you must specify the name and the parameters of the new object and then issue the create job by clicking on 'OK' or the 'Apply' button. If there are default values for parameters, these are preset by WinAdmin. Mandatory object parameters which have no default value (e.g. the name of the new object) are marked by WinAdmin with the symbol "<=>".

Objects are deleted by right-clicking a displayed object name or its icon in an object list and selecting the 'Delete...' command in the context menu which is then displayed. It is also possible to select multiple objects and delete them together.

A dialog box is then opened, in which you must confirm the deletion request. If you want to delete multiple objects, you can exclude individual objects from the deletion request by deactivating the object's checkbox. In the object lists, the icons of deleted objects are crossed out in red if they are not to be deleted immediately (delayed). In this event, the tag "(deleted)" appears beside the object name in the property dialogs.

Objects that are deleted immediately are removed from the object lists directly after their successful deletion.

17 Functions of the Application and Cluster menu

The Application or Cluster menu is displayed when you right-click on a node in the navigation tree that represents a UTM application or UTM cluster application or a node application of a UTM cluster.

The Application or Cluster menu offers the following application-related or cluster-related functions:

- 'Check Availability'

If the 'Check Availability' command is selected, WinAdmin checks whether the affected application(s) can be accessed via the UPIC interface. If the applications can be accessed (i.e. the applications are running), the availability of the application(s) is indicated by a blue application icon; otherwise, the same icon is 'crossed out' in red. An actions dialog (see also the section "The actions dialogs" on page 156) provides information on the result.

When this is done, all the activated node applications are checked in the Cluster menu.

- 'Start...'

You use the 'Start...' command to start a UTM application. This function can only be used if the file transfer product openFT is available both on the administration computer and on the application host (see also section "File transfer functions in WinAdmin" on page 129). If one application is to be started, then a dialog box with the application name and host is opened. This allows you to make the following specifications:

- Start Procedure or Service

Both for BS2000 and for Unix, the name of the procedure which WinAdmin uses to start the UTM application must be entered here. In BS2000 system this has to be an enter file that can be called with "/ENTER-JOB <name>" and under Unix/Linux systems this has to be a shell script that can be called using "sh <name>". For Windows systems, the name of a service is required which WinAdmin will then start with the command "net start <name>".

- User ID

You should enter the user ID under which the start procedure is available on the host or under which the service is started. This is particularly important if the 'FTAC Admission' box is not checked. However, even if this box is checked, WinAdmin still expects a user ID to be entered here. WinAdmin uses the unique host/user ID data pair to save the access data in the configuration database as well as for its subsequent re-use.

- Password

If the procedure is successful then you should specify the password of the user ID (if you are not using a FTAC profile).

- Account or Working Directory

The account number of the user ID must be entered here for BS2000 applications (if FTAC profiles are not used) and for Unix, Linux or Windows applications the directory into which WinAdmin transfers a job file.

- 'FTAC admission' checkbox and edit box

If the box at the left of the row is checked then the specified FTAC admission is used for file transfer instead of the user ID and password.

If the 'FTAC Admission' box is checked then the FTAC admission of the FTAC profile defined on the host must be entered in the edit box.

WinAdmin stores the entries made in this dialog box in the configuration database and automatically proposes them when the next call is issued with the result that no further user entries are required unless there is any change in the access data. As indicated above, the access data is stored using the unique host/user ID data pair.

The time needed to start the applications differs, depending on the system and the start procedure used. The message 'OK' in the action dialog initially only means that the start procedure has been called up on the host. You can use the 'Check Availability' command to determine whether or not the start was successful.

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster.

■ **'Shut...'**

The 'Shut...' command is used to shut down UTM applications. An intermediate dialog box allows you to choose between SHUT WARN, SHUT GRACEFUL, SHUT NORMAL or SHUT KILL. In the case of SHUT WARN and SHUT GRACEFUL, you can also specify the interval after which the application(s) is (are) closed and the text of a warning message.

In the Cluster menu, this command shuts down the entire UTM cluster application.

■ **'Run KDCDEF/UPD' / 'KDCDEF...'**

You can use this command to start a KDCDEF run. This is only possible if the openFT file transfer product is available both on the administration machine and on the application host (for more information, see the section "File transfer functions in WinAdmin" on page 129). The associated dialog box displays the names of the UTM application and host and also contains the following text boxes:

- Procedure

You specify the name of the KDCDEF procedure here.

In BS2000 systems, the procedure is called using the command "/CALL-PROC <procedure>, PROC-PAR=(<input-file>)", under Unix/Linux systems in the form "sh <procedure> <input-file>" and under Windows systems with "<procedure> <input-file>". The first parameter passed to the defined procedure is the input file together with KDCDEF control statements.

- Input File

Specifies the KDCDEF input file that is to be used for the KDCDEF run. This is passed to the called procedure as a parameter.

- User ID, Password, Account and FTAC Admission

As described in the explanation of the "Start..." command.

WinAdmin saves the selected file names and access information for the destination computer in the configuration database and automatically proposes these values the next time the function is called.

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster.

■ **'Get KDCDEF/UPD' / 'KDCDEF Protocol...'**

This function can only be executed if WinAdmin has already executed the 'Run KDCDEF...' function at least once for the corresponding UTM application and saved the information required to access the host in the configuration database. The dialog box that is opened contains the following elements:

- You enter the name of the source file on the host in the 'File' edit box in the 'Source' section.
- In the 'Transfer' section, you can choose whether the file is to be transferred using openFT or FTP.

- In the 'Destination' section, you use the 'File' edit box or the button with the three dots on the right of it to specify the path for the destination file. In the 'Mode' dropdown box, you can also specify whether an existing file of the same name is to be overwritten or whether the new file is to be appended to it.
- If the 'Subsequently show the file' box is checked then a new window displaying the protocol is opened after the successful completion of the file transfer.

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster.

■ **'Run KDCDEF/UPD' / 'KDCUDP...'**

The explanation of the 'Run KDCDEF...' command provided above also applies accordingly to this function. The associated dialog box differs in only two points:

- Instead of an edit box in which you specify the input file, here there are two edit boxes in which you define the new and old base names of the KDCFILES. To call the required KDCUPD procedure, it is necessary to pass the new base name as the first parameter and the old base name as the second parameter. However, these parameters are optional and you can leave the two edit boxes empty. In this case, it is assumed that the base names are defined in the procedure.

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster.

■ **'Get KDCDEF/UPD' / 'KDCUPD Protocol...'**

This function can only be executed if WinAdmin has already executed the 'Run KDCUPD...' function at least once for the corresponding UTM application and saved the information required to access the host in the configuration database. The dialog box that is opened contains the following elements:

- You enter the name of the source file on the host in the 'File' edit box in the 'Source' section.
- In the 'Transfer' section, you can choose whether the file is to be transferred using openFT or FTP.
- In the 'Destination' section, you use the 'File' edit box or the button with the three dots on the right of it to specify the path for the destination file. In the 'Mode' dropdown box, you can also specify whether an existing file of the same name is to be overwritten or whether the new file is to be appended to it.
- If the 'Subsequently show the file' box is checked then a new window displaying the protocol is opened after the successful completion of the file transfer.

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster.

■ **'Functions' / 'Message Dump'**

Defines the conditions for a message dump (as of UTM version 5.3)

(See section "Message Dump" on page 138.)

This command is not present in the context menu of cluster node applications.

■ **'Functions' / 'Create Diagdump'**

Creates a diagnostic dump for the application(s)

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster.

-
- 'Functions' / 'Inverse KDCDEF...'

You use the 'Inverse KDCDEF...' command to initiate an inverse KDCDEF run which generates files or in BS2000 library elements with KDCDEF statements for dynamic administration objects on the server running the application in question. In an intermediate step, a dialog box is opened which asks you to select the statements to be generated and enter the names of the output files or library elements and also specify other parameters.

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster.
 - 'Functions' / 'Send Message...'

The 'Send Message...' command enables you to send a message to all active UTM dialog stations for the application(s). A dialog box is opened in which you can the message. (Note: Messages to individual or multiple dialog stations can be sent via the context menu of the selected Lterms in an Lterm list.)

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster.
 - 'Functions' / 'Spoolout...'

The 'Spoolout...' command enables you to establish connections to all printers of the application(s) for which messages are available and which are not connected.

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster.
 - 'Functions' / 'Change Application...'

The 'Change Application...' command enables the application program to be changed while the application is running. In an intermediate step, you are asked whether you want to switch to the next highest or to the previous file generation.

This command is not present in the context menu of cluster node applications.
 - 'Functions' / 'Switch Uslog'

The 'Switch Uslog' command switches the user log file(s) to the next file generation group.

'Functions' / 'Switch Syslog'

The 'Switch Syslog' command switches the SYSLOG file to the next file generation group. (Note: Other functions relating to the syslog file, e.g. changing the size or writing the virtual memory to the file, are available via the syslog sheet of the property dialog for application properties).

This command is not present in the context menu of cluster node applications.
 - 'Functions' / 'Switch Protocol Files'

The 'Switch Protocol Files' command switches the application protocol files.

This command is not present in the context menu of cluster node applications.
 - 'Functions' / 'Encryption...'

This command provides you with the security functions offered by openUTM to encrypt user data during transfer between a UPIC client and a UTM application. The 'Encryption' sheet of the property dialog for the UTM application in question is displayed.

This sheet ...

 - ... displays the public part of the currently active RSA key.
 - ... displays the public part of a previously generated but not yet activated RSA key (if present).
 - ... lets you delete a new and not yet activated RSA key pair.

- ... lets you generate a new RSA key pair.
- ... lets you activate a new and not yet activated RSA key pair.

This command is not present in the context menu of cluster node applications.

■ 'Functions' / 'Update IP-Addresses...'

The 'Update IP-Addresses...' command allows the IP addresses stored in the object tables of UTM applications to be compared with the IP addresses stored in the host name database. The addresses of all the affected communication partners of types Pterm, Con and Osi-Con are updated in the applications.

■ 'Functions' / **'Show Open Transactions in PTC State'**

Displays a list of all the currently open transactions that have the PTC state.

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster.

■ 'Functions' / 'Security.'

See section "Support for UTM access control" on page 75.)

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster.

■ 'Functions' / **'Online Import'**

This command is only available in a Cluster menu or in the Application menu of a cluster node application.

It performs an online import of a node application which is not currently running into this node application.

The non-running node application whose data is to be imported into this node application must be selected in an intermediate dialog box.

This menu item is grayed (disabled) if WinAdmin considers that all activated node applications are currently running.

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster.

■ 'Functions' / **'Explicitly Sign-Off User...'**

This command is only available in a Cluster menu or in the Application menu of a cluster node application.

This command is only possible for node applications which WinAdmin does not currently consider to be running. It explicitly signs off any UTM users who are still signed on at the non-running node application with UTM. The command is executed by the UTM node application that is currently identified as "master" by WinAdmin.

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster which WinAdmin does not currently consider to be running.

■ 'Functions' / **'Abort Bound Services...'**

This command is only available in a Cluster menu or in the Application menu of a cluster node application.

This command is only possible for node applications which WinAdmin does not currently consider to be running. For a detailed description of this function, see the UTM administration guide (KDCADMI operation code KC_LOCK_MGMT, sub-opcode KC_ABORT_ALL_BOUND_SERVICES). The command is executed by the UTM node application that is currently identified as "master" by WinAdmin.

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster which WinAdmin does not currently consider to be running.

- **'Functions' / 'Repair User File Lock...'**

This command is only available in a Cluster menu or in the Application menu of a cluster node application.

This command allows you to cancel the lock on the UTM user file if this is still set as a result of an aborted KDCDEF run.
- **'Functions' / 'Host'**

Opens the property dialog for the UTM application host.

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster.
- **'Functions' / 'Current UPIC Connection'**

Opens the property dialog for the UPIC connection to the selected UTM application.

In the Cluster menu, this command is node-specific, i.e. it contains a submenu with an entry for each activated node application in the cluster.
- **'Create...'**

New administration objects can be created dynamically with the 'Create' command (see section "Creating and deleting objects" on page 67).
- **'Transaction'**

You use this command and the associated submenu to find out information about an open transaction and terminate or reset the transaction (see section "Transaction-oriented administration jobs" on page 97).
- **'Command Mode'**

You use the 'Command Mode' command to activate the command mode's Tac interface (see section "Command mode" on page 136).
- **'Refresh data'**

Refreshes **all** the object and application parameters of the UTM application.
- **'Remove from Collection'**

Removes the selected UTM application from the collection that is currently open.

This command is not present in the context menu of cluster node applications.
- **'Layout' / 'Add Category...'**

Adds a new category to the subtrees of UTM applications or UTM cluster applications.

See the section "Layout of the Application and Cluster subtrees" on page 45.
- **'Modify Cluster Name'**

This command is only available in a Cluster menu.

It is used to modify the name of a cluster as displayed in WinAdmin.
- **'Properties'**

The 'Properties' command allows you to view the properties of one or more UTM applications or an UTM cluster application in a special property dialog and to change some of these properties.

18 Object selections

Each collection has an object selections which is displayed in its own window. You can copy UTM objects (including objects of different types) into this window. Thanks to the object selections, frequently required UTM objects can be displayed centrally, for example. However, they are not suitable for storing very large numbers of objects.

The highest level of the WinAdmin navigation tree contains the "Object Selection" leaf in the "Administration" subtree. Click on this entry to open the appropriate selection window. Alternatively, you can select the menu command 'Show Object Selection' in the context menu of each node or press the F3 key. A selection window displays the selected objects in list form, together with their names, object types, and associated application and host or cluster.

There are various ways of adding UTM objects to an object selection. The first step is always to display the object list for the relevant objects and highlight the objects in this list. You can then choose the 'Add to Object Selection' command in the context menu of the highlighted objects. Another way is to drag the highlighted objects to the "Object Selection" node in the navigation tree or to the open Object Selection window (drag-and-drop). You can also copy and paste selected objects. To do this, press the keyboard shortcut **Ctrl+C** (to copy) in the relevant object list, then activate the selection list (by clicking in the list) and then press the keyboard shortcut **Ctrl+V** (paste) to add the objects to the object selection.

The objects in the object selection are saved in the configuration database so that they are available for use in subsequent sessions. It should be noted that only a reference to the objects is stored. The object properties are then requested from the UTM applications when they are to be displayed.

To remove objects from the object selection, open the object selection, highlight the corresponding objects in the list and click the "Remove" button below the list.

You should note that the object selection may possibly contain references to objects that have since been removed from the associated UTM application.

19 Support for the UTM cluster concept

As of Version 6.0, UTM supports the creation of clusters which consist of multiple similar UTM applications (node applications). This section describes how WinAdmin supports the UTM cluster concept.

Note:

In this document, the terms "cluster" or "UTM cluster" are frequently used as abbreviations for "UTM cluster application".

19.1 UTM clusters and node applications in the navigation tree

In the navigation tree, UTM clusters are depicted in a separate subtree with the name 'Cluster', whereas, as of WinAdmin Version 6.0, standalone applications are listed under the 'Stand-Alone' subtree.

For each cluster in the open collection, the subtree below the 'Cluster' node contains a node with the name of the associated cluster. This node and the associated subtree represent the global view of the entire cluster and it is therefore structured in a similar way to the node and subtree for an individual application below the 'Applications' node in the 'Stand-Alone' subtree. The local view of the individual applications in a cluster is provided by the navigation tree's 'Cluster Nodes' node (second child node below a cluster tree node). The subtree of this tree node contains a separate tree node for each node application in the cluster. This allows access to the local data and information for the node.

Consequently, the following basic rule applies for users:

Actions that are performed at the cluster tree node are global actions within the cluster whereas actions that are performed at node application tree nodes are local to the application (node-local actions). The same principle applies to the displayed data: Data that is displayed at the cluster tree node (and in the lists corresponding to it) is global within the cluster, whereas data that is displayed at node application tree nodes (and in the lists corresponding to them) is local to the node.

Example:

The 'Shut...' command in the cluster's context menu shuts the entire cluster whereas the same command in a node application's context menu only terminates the corresponding application.

For this reason, the cluster tree node's context menu only contains the commands that have a global effect in the cluster and only those nodes that are available globally throughout the cluster are presented as child nodes (e.g. there is no 'Command Mode' child node below the cluster tree node; this is, however, present below the tree nodes of the individual node applications).

In WinAdmin Version 6.0, there was not only the node for the individual node applications but also the node 'All Activated Node Applications' below the 'Cluster Nodes' node. This node is no longer present in the tree in the current version because the tooltips functionality (see section "Tooltips in lists and property dialogs" on page 159) provides a very powerful way of displaying and comparing the individual node applications in the object lists that apply globally to the cluster.

Example:

The UTM cluster contains a user with the name 'User1'. In the global list, the value '<mix>' is displayed for the property 'Signed-On' for this user. This means that this user is signed on to one or

more node applications and is not signed-on to one or more other node applications. To find out which node applications this user is signed-on to, you simply need to look at the tooltip for the corresponding cell in the object list. For each individual node application, this indicates the value local to the node and therefore allows you to identify quickly and easily which node applications the user is and is not signed-on to.

19.2 Cluster definition in WinAdmin

No explicit definition of a cluster object is required in WinAdmin. You declare a cluster node application in WinAdmin in the usual manner (host object, application object, UPIC connection). The application is then initially listed as a Stand-Alone application in an open collection. The first time that contact is successfully established between WinAdmin and the application (e.g. by explicitly checking availability), WinAdmin identifies that the application is a cluster node application. If no cluster name has so far been assigned to the cluster node application, WinAdmin asks for the name of the cluster as it is to be displayed in WinAdmin. The application is then removed from the Stand-Alone subtree and the 'Cluster' node is created (if it is not already present). The cluster with the name you have just defined is then listed below this. WinAdmin then reads the cluster node data of the UTM application which it has just successfully contacted and (if this has not already been done) automatically creates the WinAdmin objects of the other node applications that belong to the cluster (i.e. hosts, applications and UPIC connections). A message is output in the interface for the newly generated WinAdmin objects. These objects are automatically written to the configuration database and are therefore already present when the next program session is started. In addition, all the cluster node applications are automatically taken over into the collection that is currently open and are inserted in the cluster tree node's 'Cluster Nodes' subtree.

You are therefore advised to explicitly define the WinAdmin objects (host object, application object, UPIC connection) of **only one** cluster node application. The relevant WinAdmin objects of the other cluster node applications are then automatically generated by WinAdmin as described above.

WinAdmin can implicitly generate the objects required for the other cluster node applications due to the fact that a cluster's individual cluster node applications are generated in the same way and any other data that is required (e.g. the name of the computer on which the node applications run) is present in the cluster node data of each individual node application.

The checks described here are always performed when a UTM (cluster node) application has been successfully contacted during a program session. This ensures that the applications of newly generated cluster nodes are also automatically added to WinAdmin and do not have to be explicitly added by the user.

19.2.1 UPIC users in the UTM cluster

If the UTM cluster was generated with MULTI-SIGNON=NO then a different UPIC user must be used for each node application during communications with WinAdmin. It is otherwise possible that the attempt to establishment a connection from WinAdmin to one of the node applications will be rejected because the relevant UPIC user, with which WinAdmin wants to sign on to UTM, is still signed on at UTM.

The simplest way of avoiding such problems is to generate a USER conforming with the name pattern <UserNamePrefix><nn> for each node application. Here, <UserNamePrefix> is a freely selectable name prefix up to six characters in length. <nn> is a two-digit number in the range from '01' to the number of generated cluster nodes.

Example:

If six cluster nodes are generated then the six following USERS could be generated for communication with WinAdmin:

WINADM01
WINADM02
WINADM03
WINADM04
WINADM05
WINADM06

When contact has been successfully established with a node application belonging to a cluster, WinAdmin checks whether the UPIC user specified for the UPIC connection to the node application complies with the described name convention. In particular, it checks the number indicated in the last two digits of the name. If this section of the name corresponds to the index of the relevant cluster node then WinAdmin assigns the UPIC users for these connections in accordance with the convention when importing the other node applications.


The index of the cluster node can be determined indirectly from the KDCDEF input used for the generation of the cluster. The first cluster node specified ('CLUSTER-NODE' statement) has the index 1, the second has the index 2, etc.

Example:

If WinAdmin successfully establishes contact with the node with index 3 and if the UPIC user for the connection has the name 'WINADM03', then WinAdmin assigns the UPIC users 'WINADM01', 'WINADM02', 'WINADM04' etc. for the remaining node applications.

Alternatively to this automated method of assigning UPIC users, they can, of course, also be modified manually after the node applications have been imported by WinAdmin. To do this, you can, for example, choose 'Functions' / 'Current UPIC Connection' from the context menu of a node application and modify the UPIC user accordingly. This change takes effect the next time the collection is loaded.

19.3 Deactivating cluster nodes

Cluster nodes that are generated for future use in UTM, but for which the corresponding applications are not currently (not yet) being run, can be disabled in WinAdmin. This can be done by choosing the item 'Deactivate Node Application' in a cluster node application's context menu (in a cluster's 'Cluster Nodes' subtree). Any node applications deactivated using this menu item are displayed with a special icon () in the navigation tree. The subtree of a node application that has been deactivated in this way is empty. It is therefore not possible to access the objects and properties of these node applications. WinAdmin does not take account of deactivated node applications during cluster actions (availability checks, object retrievals etc.).

The context menus of deactivated node applications contain only the menu item 'Activate Node Application' which can be used to reactivate them.

19.4 Context menu of a UTM cluster

This context menu is displayed if you right-click a node in the navigation tree that represents a UTM cluster application.

The individual commands in this menu are described in detail in the section "**Fehler! Verweisquelle konnte nicht gefunden werden.**" on page 69.

19.5 Context menu of node applications in the UTM cluster

This context menu is displayed if you right-click a node in the navigation tree that represents a UTM cluster node application.

The individual commands in this menu are described in detail in the section "Functions of the Application and Cluster menu" on page 69.

19.6 Properties sheet for UTM clusters and node applications

The properties sheets for UTM clusters and node applications contain an additional page named 'Cluster' which contains the general properties of the UTM cluster.

In a UTM cluster's property sheet, it is possible to set the following values on this page (this is not possible in a node application's properties sheet because the changes have global effect throughout the cluster):

- 'Alive Timer (sec)'
- 'Alive check retry'
- 'Reply Timer (sec)'
- 'Restart Timer (sec)'
- 'File Lock Timer (sec)'
- 'File Lock Retry'
- 'Deadlock Prevention'

19.7 List of UTM cluster nodes

UTM clusters and node applications possess another object list with the name 'Cluster Nodes' which contains the properties of the individual cluster nodes which belong to the UTM cluster. Like all other object lists, this list can be accessed via the navigation tree.

It is not possible to modify any values in the property sheets of cluster nodes which have been displayed via a local, node-level list, whereas the following (global) values can be edited in the properties sheets of cluster nodes which have been displayed via the global cluster-level list:

- 'Node hostname'
- 'Virtual Host'
- 'Filebase'
- 'CATID A'
Only for UTM clusters in BS2000 systems.

You can only make these changes to cluster nodes whose node applications are currently not running!

19.8 Object lists of a cluster

Below a UTM cluster's tree nodes, you can see a list of the nodes which represent the object lists of these clusters. If you click one of these nodes or choose the 'Show Object List' command in the node's context menu then a list of objects of the corresponding type is displayed. This list "combines" the objects of all node applications that are currently activated and available. This means, for example, that the total number of program unit runs of a given openUTM TAC is output here (the total number of all program unit runs for this TAC in all currently activated and available node applications in the relevant cluster).

The properties sheets of the objects in these lists also represent the "combined" object. If properties that are not totaled when combining the objects differ between the individual cluster applications, the text '<mix>' is output here. However, this should not normally occur since the cluster applications should be generated in the same way. However, it should be noted that WinAdmin is not able to retrieve data from all the cluster applications simultaneously and that this type of <mix> status may arise in the case of values which may be dynamically modified. This may also occur for properties that have been changed by the administrator and where the administration command has not yet taken effect in all node applications,

These object list nodes are also listed below the individual node applications' tree nodes. Here, however, they simply refer to the relevant node application (local, node-level lists).

20 Support for UTM access control

You can protect certain UTM objects against access by other UTM objects. This is achieved by means of a lock/key concept. It is also possible to identify Tacs as administration or SAT administration Tacs. These Tacs can then only be accessed by users who possess the administration or SAT administration privilege (Admin/Satadm Permit).

Access control has been extended in openUTM V5.1. To a very large extent, these extensions are supported by WinAdmin. Below, we first present a brief outline of the familiar ("old") UTM access control and then describe the extensions and the new possibilities available to users. This is followed by a detailed description of how WinAdmin supports the new access control concept.

20.1 Accessing UTM services – keys and locks

In outline terms, when "users" make use of a UTM application, they always access a "service" via a "coupler". This therefore always results in the following access hierarchy:

"User" → "Coupler" → "Service"

In order to protect services against unauthorized access, both services and couplers can have locks in UTM. Users can thus only access the coupler and, consequently the service, if they possess the appropriate keys for both locks. Depending on the type of service that the user wants to access, the employed coupler may also need a key that fits the service's lock.

Keys and locks are both represented by a number (keycode/lockcode). Multiple keys can be combined to form a keyset (keyset, UTM object type, Kset). The privileges (authorizations) owned by users and couplers are therefore represented by their keysets.

If a service or coupler does not possess a lock (or has a lock with lockcode 0) then it is unprotected. If the user or coupler possesses no keyset (or an empty keyset) then no special authorization exists and only unprotected couplers and services can be accessed.

Thus users can only access a service if

1. one of their keys fits the service's lock or the service is unprotected
and
2. one of their keys fits the coupler's lock or the coupler is unprotected
and (possibly)
3. one of the coupler's keys also fits the service's lock or the service is unprotected

20.1.1 "Services"

The services of a UTM application are represented by objects of type Tac or Ltac. In the case of a Tac, the local UTM application provides the service itself by executing the program unit corresponding to the Tac. In the case of an Ltac, the service is provided by another application.

The services can be protected by a lock (LOCK parameter in the TAC/LTAC generation statement).

If a user wants to access a Tac that is protected by a lock then the coupler that is used must also possess a key that corresponds to the Tac lock. In contrast, when an Ltac is accessed the user only requires the appropriate key.

The queues introduced in UTM version 5.1, i.e. Tac queues (TAC..., TYPE=Q,...) and user queues, also count as services. In the case of queues, it is possible to set up separate protection for read and write access. This is achieved by using two different Access Lists (see below).

20.1.2 "Users"

Users are represented by UTM objects of type "User". Users can explicitly generate "User" objects (by means of USER generation statements). In this case, the user's authorizations are determined by the user Kset (KSET parameter in the USER generation statements).

In the case of certain couplers, the "User" objects are implicitly generated by the UTM utility KDCDEF. The authorizations possessed by users in such cases are described in the next section.

20.1.3 "Couplers"

Users can be coupled to a UTM application in the following ways:

■ Terminal coupling (Lterm/Tpool)

If this type of coupling is used, the coupler can be either an Lterm or a Tpool.

If the UTM application has been generated without users then openUTM implicitly generates a user. This implicitly generated user "inherits" the Kset of the Lterm or Tpool. Otherwise, the user signed on via the Pterm or Tpool (and thus also the user's Kset) is used.

If this type of coupling is used, the coupler can be protected by the lock used by the Lterm or Tpool (LOCK parameter in the LTERM or TPOOL generation statement).

The coupler's authorization is defined in the Kset of the Lterm or Tpool (KSET parameter in the LTERM or TPOOL generation statement).

■ Upic/Appli coupling (Lterm/Tpool)

Here again the coupler can be either an Lterm, in which case the associated Pterm has the PTYPE *Upic* or *Appli*, or a Tpool with PTYPE *Upic* or *Appli*.

If the coupler is an Lterm, openUTM implicitly generates a user. This implicitly generated user "inherits" the Lterm's Kset.

If the coupler is an Lterm, openUTM implicitly generates a user. These Tpool users inherit the USER-KSET of the Tpool if a USER-KSET is present. Otherwise they inherit the Tpool's Kset. If a user is specified when the connection (PTYPE=*Appli*) or conversation (PTYPE=*Upic*) is set up then this user and the associated Kset are used.

If this type of coupling is used, the coupler can be protected by the lock used by the Lterm or Tpool (LOCK parameter in the LTERM or TPOOL generation statement).

The coupler's authorization is defined in the Kset of the Lterm or Tpool (KSET parameter in the LTERM or TPOOL generation statement).

■ LU6.1 coupling (Lpap)

For each Lses belonging to an Lpap, openUTM implicitly generates an Lses user who inherits the Lpap's Kset.

If this type of coupling is used it is not possible to protect the coupler itself.

The coupler's authorization is also defined by the Kset of the Lpap (KSET parameter in the LPAP generation statement).

■ OSI-TP coupling (Osi-Lpap)

For each OSI Association that belongs to an Osi-Lpap, openUTM implicitly generates an Association User. This user inherits the ASS-KSET of the Osi-Lpap if an ASS-KSET is present. Otherwise the user inherits the Osi-Lpap's Kset. When an OSI Association is set up, openUTM initially assigns the Osi-Lpap the Association User generated for the association. However, if a user is specified when a conversation is set up then the specified user is assigned to the Osi-Lpap.

If this type of coupling is used it is not possible to protect the coupler itself.

The coupler's authorization is defined by the Kset of the Osi-Lpap (KSET parameter in the OSI-LPAP generation statement).

20.2 The "new" concept – roles and access lists

Although the access control concept that was previously employed is clearly extremely powerful (protection mechanisms of any level of complexity are conceivable) it is nevertheless somewhat confusing for users.

On the one hand, this is due to the exclusive use of numbers for lockcodes and keycodes at the UTM interfaces (configuration, administration).

On the other, at the time of UTM generation it is necessary to decide on the services which individual users are to be able to access. However, it is frequently easier to view things from the opposite direction: which users should be able to access this service?

The "new" access control concept adheres to this second point of view.

This concept introduces the idea of "roles". It is possible to assign one or more roles to each user (e.g. "departmental head", "clerk" or "caretaker"). If you want to control access to a particular service then you set up an access list for this service. This access list in turn contains one or more roles. Only users who are assigned at least one of the roles in the list are then able to access the service. For example, the access list for the "Payroll" service might contain the roles "departmental head" and "salary administrator" since all departmental heads and salary administrators are able to use this service whereas caretakers cannot.

This concept is largely derived from the "old" lock/key concept: the keycodes/lockcodes simply need to be interpreted as roles. Any given keycode/lockcode corresponds to a specific role (e.g. the code "1" might correspond to the role "clerk").

A keyset thus corresponds to a role list or access list.

The definition of user or coupler authorizations in the form of role lists thus corresponds to generating a Kset for the user and coupler objects.

In order to permit the definition of access lists for services, the generation statements for the Tac and Ltac object types have been extended by the ACCESS-LIST parameter which makes it possible to assign an access list in the form of a keyset.

To keep the new concept as simple and clear as possible, no access lists are assigned to couplers. The LOCK parameter for Lterm and Tpool couplers continues to be supported for reasons of compatibility. However, you are advised not to assign any lockcodes to the couplers. Although this means that all the couplers are unprotected this does not matter since it is not the couplers themselves but the services in the form of Tacs and Ltacs that are the resources that need protection.

Thus, in the new concept, users can only access a service if

1. they have a role that is stored in the service's access list or the service is unprotected
- and**
2. the coupler that is used has a role that is stored in the service's access list or the service is unprotected.

To a very large extent, this concept is supported by WinAdmin. This support is described in the following sections.

20.3 Roles

In WinAdmin, roles are depicted as "normal" UTM objects, for example as Ksets. However, the role objects are known only to WinAdmin and are not communicated to the UTM administration interface since openUTM is still only able to use lockcodes and keycodes.

The main purpose of the role objects is to link the uncommunicative numerical lockcodes and keycodes with user-definable symbolic names. The "name" of a role object corresponds to the lockcode/keycode that is passed to openUTM. The "Alias" property of the role objects specifies the symbolic name of the role, for example "departmental head" or "clerk", and can be edited as required. Wherever roles are displayed in WinAdmin, for example in object lists and property dialogs, the alias of the role is specified instead of the name.

Because there are no generation functions in this version of WinAdmin, the role lists are located in the administration part of the navigation tree in the same way as all the other UTM object lists. As in the past, the role objects are known only to WinAdmin and are stored in the configuration database.

In the case of roles, it is possible to distinguish between user roles (represented by Ksets) and access roles (represented by locks and access lists).

The table below provides an overview of the security-related UTM object types and generation parameters.

Object type	Type	Generation parameter	Role type
USER	User	KSET	User roles
	Service (queue)	Q-READ-ACL	Access roles (read access to the user queue)
		Q-WRITE-ACL	Access roles (write access to the user queue)
LTERM	Coupler	LOCK	Access role
		KSET	User roles
TPOOL	Coupler	LOCK	Access role
		KSET	User roles (of the Tpool and an implicit user if no USER-KSET is present)
		USER-KSET	User roles (of an implicit user)
LPAP	Coupler	KSET	User roles
OSI-LPAP	Coupler	KSET	User roles (of the Osi-Lpap and an implicit user if no ASS-KSET is present)
		ASS-KSET	User roles (of an implicit user)
TAC	Service	LOCK	Access role (access to an asynchronous or dialog Tac)
		ACCESS-LIST	Access roles (access to an asynchronous or dialog Tac)
	Service (queue)	Q-READ-ACL	Access roles (read access to the Tac queue)
		Q-WRITE-ACL	Access roles (write access to the Tac queue)
LTAC	Service	LOCK	Access role
		ACCESS-LIST	Access roles

20.3.1 Roles in object lists

The object lists for all the object types that possess locks, Ksets or access lists contain additional columns corresponding to the roles represented by the locks, Ksets and access lists. The individual roles are comma-separated in the display. If no role object is assigned to a lockcode/keycode then the role is instead represented by its printable code.

Example:

An access list contains the codes 1, 2 and 5. Code 1 is the role with the alias "departmental head" and code 5 the role with the alias "clerk". As yet, no role has been assigned to code 2.

In this case, the "Access Roles" column would contain "Departmental Head,2,Clerk".

20.3.2 Roles in property dialogs

The property dialogs for objects that may reference Ksets or access lists have (at least) one additional dialog sheet which lists not only the Kset or access list but also the user roles or access roles. In the case of Tac and Ltac dialogs this sheet also accommodates the LOCK parameter so that all the parameters relating to access rights can be found here.

In the case of applications which use the new role concept, i.e. for which at least one role (alias) has been defined, the user and access rights it contains are displayed in the form of a list of comma-separated roles or codes in the same way as in object lists. Alongside this, a list window displays all the roles defined in the application together with their aliases and codes and an indication of whether the role is present in the Kset (in the access list). You can use the 'Add to Role List' or 'Remove from Role List' commands in the context menu to add selected roles to or remove them from the Kset (access list).

You can also use the context menu to add roles to the object selection and delete roles (provided they are not referenced) or open the property dialog for roles.

Adding or removing roles modifies the Kset. To make this clear, the Kset name "RLSTnnn (*)" is displayed in the combo box for the corresponding KSET= or ACCESS-LIST= parameter. When you implement a change with 'OK' or 'Apply', WinAdmin performs an internal check to determine whether there is already a Kset with the new role combination. If there is then this Kset is used. If there is not then either the old Kset is modified (if it is not referenced by other UTM objects) or a new Kset with the generic name RLSTnnnn is created.

Tpool and Osi-Lpap dialogs both contain two user role sheets (for Kset and User-Kset or Ass-Kset). Tac dialogs contain two access role sheets (for read and write access to the queue) for Tac queues (TYPE=Q). In user dialogs there are three role sheets: one sheet for the user role and two sheets with the access roles for read and write access to the user queue.

The roles are not displayed if the property dialog reproduces the combined properties of multiple objects and multiple applications are involved or if the associated Kset or access list parameters have different values (<mix>).

20.4 The Security view

The Security view depicts all security-related objects and their interrelations in a tree structure (similar to the navigation tree). There is always one Security view for the administration data for every UTM application. You can open the view via the navigation tree in the same way as UTM object lists. Alternatively, you can use the Application menu. This contains the 'Security' menu command which can also be used to display the Security view to the application.

The Security view contains the subtrees 'Security Resources', 'Objects That May Be Privileged', 'Objects That May Be Protected', 'Local Services', 'Remote Services' and 'Queues' which are described in greater detail in the sections that follow. The first three of these subtrees provide an overview of the employed security resources (i.e. roles, codes, Ksets/access lists) and privileged/protected objects. The final three subtrees present the complete access hierarchy of all users to the corresponding services.

You can make settings for the Security view above the Security tree. Thus, for example, you can define the security attributes that are to be taken into account in the view.

- 'Consider Codes/Roles'
If you select this option then all the Ksets, access lists and lock codes of the UTM objects are considered.
- "Consider Admin Permit"
If you select this option the view takes account of whether Tacs are administration Tacs and whether users have to possess the Admin Permit.
- "Consider Satadm Permit"
If you select this option, the view takes account of whether Tacs are SAT administration Tacs and whether users have to possess the Satadm Permit.

If you select all three options then the Security view takes account of all these considerations when displaying in the same way that openUTM does when deciding whether a user is able to access a service.

The Security view is able to display all possible user accesses to services ('Possible Access' radio button) as well as all impossible accesses ('Impossible Access' command). If the view displays the impossible accesses then it also specifies why the individual accesses are not possible.

20.4.1 'Security Resources' subtree

This subtree informs you of any other UTM objects that are using the security resources. The security resources consist of roles, codes (i.e. lockcodes/keycodes to which no role has been assigned), Ksets (or access lists) and the Admin or Satadm property. There is a subtree for each of these resource types. Below each subtree there is an entry for each object under which (grouped by object type) all the UTM objects that reference the selected subtree are listed.

The subtrees are as follows:

■ Roles

This subtree is only available if you have selected the 'Consider Codes/Roles' option.

Below this node you will find an entry for each role that is defined in the UTM application. Since a role can be referenced by objects of the types Kset, User, Lterm/Tpool, Lpap/Osi-Lpap, Ltac/Tac and Queues there is a subtree for each of these types below each role. If you expand one of these nodes, WinAdmin displays below it all the objects of the corresponding type that reference the selected role.

The 'Tacs' subtree does not contain any Tac queues (TAC...,TYPE=Q). These are listed in the 'Queues' subtree.

■ Codes

This subtree is also only available if you have selected the "Consider Codes/Roles" option.

Below this node you can see all the codes from 1 to MAX KEYVALUE to which no role has as yet been assigned. The subtree is structured in the same way as the 'Roles' subtree.

■ Ksets

This subtree is also only available if you have selected the "Consider Codes/Roles" option.

This subtree handles all the Ksets defined in the UTM application in the same way as the two other subtrees. A Kset can be referenced by objects of the types User, Lterm/Tpool, Lpap/Osi-Lpap. It can also be referenced as an access list by objects of the types Ltac/Tac as well as by queues.

Here, too, the 'Tacs' subtree does not contain any Tac queues (TAC...,TYPE=Q). These are listed in the 'Queues' subtree.

■ Admin Permit

This subtree is only available if you have selected the "Consider Admin Permit" option.

The subtree contains all the objects that have the Admin Permit or Admin property. Objects of type User and Lpap/Osi-Lpap may have the Admin Permit. Tac objects and queues may also possess the Admin property.

The 'Tacs' subtree does not contain any Tac queues (TAC...,TYPE=Q). These are listed in the 'Queues' subtree.

■ Satadm Permit

This subtree is only available if you have selected the 'Satadm Permit' option.

It takes account of the Satadm Permit and Satadm property and is structured in the same way as the 'Admin Permit' subtree.

20.4.2 'Objects' That May Be Privileged' subtree

This subtree provides you with an overview of all the objects that can have privileges, i.e. objects of the types User, Lterm/Tpool and Lpap/Osi-Lpap. The security resources and privileges selected using the options above the Security tree are also taken into account here.

The 'unprivileged' subtree lists all the objects (grouped by object type) that have no privileges, i.e. to which no or only an empty Kset has been assigned and do not possess the Admin/Satadm Permit. The 'privileged' subtree displays all the privileged objects in the same way.

20.4.3 'Objects' That May Be Protected' subtree

This subtree is the counterpart to the 'Objects That May Be Privileged' subtree. However, it additionally offers an overview of all the objects that can be protected, i.e. objects of the types Lterm/Tpool, Lpap/Osi-Lpap, Ltac/Tac and Queues. The security resources and privileges selected using the corresponding options are again taken into account here.

The 'unprotected' subtree lists all the objects (grouped by object type) that have no protection, i.e. which have no lockcode or to which no access list has been assigned, and do not possess the Admin/Satadm property.

The 'protected' subtree displays all the protected objects in the same way. Objects to which an empty access list is assigned are also protected (no access at all is permitted to these objects).

Here, too, the 'Tacs' subtree does not display any Tac queues. These are listed in the 'Queues' subtree.

20.4.4 "Local Services" subtree

This subtree displays the complete access hierarchy to local services, i.e. Tacs. Tac queues are not taken into account here. There is a separate subtree named 'Queues' for the access hierarchy to queues.

Depending on the settings above the Security tree, either the possible or impossible user accesses to services are displayed here. The impossible access display is useful if you want to find out why a user (working via a given coupler) is unable to access a certain service. The Security view gives the reason for every impossible access (e.g. user does not have a suitable key for the coupler).

There is one view from the users' viewpoint ('Users Sight' subtree) and one from the services' viewpoint ('Services Sight' subtree). The 'Users Sight' indicates the Tacs that each user may or may not access (and via which couplers). The 'Services Sight' indicates which users may or may not access each service (and via which couplers).

■ 'Users Sight' subtree

The 'Users Sight' subtree contains a node for each generated user of the UTM application. It also contains the nodes 'Tpool Users', 'Lses Users' (in the case of UTM-D generation) and 'Association Users' (in the case of an OSI-TP generation). 'Tpool users' refers to users who are implicitly generated by UTM and employed for a terminal, Appli or Upic connection. 'Lses Users' refers to users who are implicitly generated and employed for an LU6.1 connection. 'Association Users' refers to implicitly generated users for an OSI-TP connection.

On the next level below the Users node you will find a list of the types of coupling permitted for the user in question, i.e. terminal, Appli, Upic, LU6.1 and OSI-TP connections.

Below these are located the individual coupling objects that can be accessed by the users in question. In the case of terminal, Appli or Upic connections, these can be Lterms or Tpools. In the case of an LU6.1 connection, only Lpaps are possible and for an OSI-TP connection only Osi-Lpaps.

The level below the possible couplers lists all the Tacs that the user can or cannot access via the coupler.

Directly below each User node, there is a 'Tacs' node which enables you to determine precisely which Tacs any given user can or cannot access irrespectively of the coupler. This does not apply to Tpool, Lses and Association Users since the authorizations of these implicit users are derived from the employed couplers.

- 'Services Sight' subtree

This subtree is practically speaking the inverse of the 'User Sight' subtree. It contains a node for each Tac defined in the corresponding UTM application (Tac queues are represented in the 'Queues' subtree).

On the next level below these Tac nodes you will find a node for each possible connection type i.e. terminal, Appli, Upic, LU6.1 and OSI-TP connections.

Below this is a list of all the possible couplers for the connection type that can access the corresponding Tac. In the case of terminal, Appli or Upic connections, these can be Lterms or Tpools. In the case of an LU6.1 connection, only Lpaps are possible and for an OSI-TP connection only Osi-Lpaps.

The level below the possible couplers lists all the users (including implicit users) who can or cannot access the Tac via the corresponding coupler.

The additional 'Users' node is also present directly below the Tac nodes in the 'Services Sight' subtree in order to allow you identify which users may or may not access a given Tac directly, i.e. without taking account of the coupler.

20.4.5 "Remote Services" subtree

This subtree displays the complete access hierarchy to remote services, i.e. Ltacs. It is structured in the same way as the 'Local Services' subtree.

20.4.6 'Queues' subtree

This subtree displays the complete access hierarchy to queues (i.e. Tac queues (TAC...,TYPE=Q) and User queues). It is structured in the same way as the 'Local Services' subtree.

Since it is possible to set read and write access control separately for queues, this subtree always indicates whether the permitted (or prohibited) access refers to read access ('Read') and/or write access ('Write').

21 WinAdmin message destination - message collectors

There are so-called "service-controlled" queues in openUTM (see also section "Service-controlled queues in WinAdmin" on page 107). You can use FPUT or DPUT to send messages to such a queue from within a UTM program unit. Another program unit can then read the messages by means of DGET calls. The queues operate in accordance with the FIFO principle, i.e. the messages are read out in the same sequence as they are sent to the queue. Alongside the possibility of creating and using such queues on a temporary basis (during a program unit run), it is also possible to use Tac and user queues.

An interesting aspect of the Tac and user queues is that they are permissible message destinations for UTM messages. This queue property has been used to make WinAdmin into a kind of "message destination" for UTM messages. WinAdmin is able to retrieve messages from the addressed UTM queues either cyclically or at the request of the user, enter these in lists and, if necessary, save them in the configuration database.

The following sections describe what you have to do to make WinAdmin into a "message destination" and the various possibilities that this offers to users.

21.1 Actions on the UTM side

On the UTM side, you must make sure that UTM messages are actually sent to a queue from which they can be retrieved by WinAdmin. To do this, you must carry out the following steps:

- Define messages with the message destination USER-DEST-1...4
You must specify the (additional) message destination USER-DEST-1 (or USER-DEST-2...4) for all messages that WinAdmin is to be able to retrieve. You can do this using the MODMSG statement and the UTM utility KDCMMOD.
- Create a new message module
You can use the KDCMMOD utility to create a separate message module for the UTM application (on the basis of the file containing the MODMSG statements).
- Generate the separate message module
If openUTM is to be able to use the created message module, you must generate it using the KDCDEF MESSAGE statement.
- Link the message destination USER-DEST-1...4 with a queue object
This is done using the KDCDEF statement MSG-DEST. Here the NAME and DEST-TYPE parameters define the queue to which the corresponding UTM messages are to be sent. In the case of MSG-FORMAT, you should specify PRINT.

Example:Source for KDCMMOD:

```

GEN SYS, LANG=D, MODULE=OWNMSG
*
MODMSG 036, COMPRESSION = NO, USER-DEST-1 = YES
ENDMSG
.
.
.
END

```

KDCDEF input:

```

MESSAGE LIB=$XYZ.OML.MSG, MODULE=OWNMSG
*
MSG-DEST USER-DEST-1, NAME=MSGQUEUE, DEST-TYPE=TAC, MSG-FORMAT=PRINT
*
TAC MSGQUEUE, TYPE=Q, QLEV=500, QMODE=WRAP-AROUND

```

21.2 Actions on the WinAdmin side

Before WinAdmin can fetch messages from the UTM applications, you must define a so-called message collector. As far as WinAdmin is concerned, message collectors are objects - in the same way as UTM users - that are known only to WinAdmin. Their properties determine the UTM queues from which WinAdmin fetches messages, when it fetches them and what it then does with these messages.

In the Administration subtree of the navigation tree, you will find the entry "Message Collectors" below a UTM application. The context menu for this entry contains the 'Show Message Collectors' command, that displays a list of all the message collectors defined for the UTM application (or the cluster) in question. Below the list, there is an 'Add' button that allows you to create new message collectors.

21.3 Properties of message collectors

The property dialog for message collectors contains the following controls:

- 'UTM Queue Objects' section

Here you define the UTM queue from which the messages are to be fetched. The types 'Tac-Queue' (i.e. TAC...,TYPE=Q) and 'User-Queue' are possible. User objects can be used as queues if the generated QLEV is greater than zero. 'Name' specifies the name of the UTM object (i.e. either the name of a Tac or user queue).

- 'Poll Mode' dropdown box

Specifies when WinAdmin is to retrieve the messages from openUTM. Here you can either set 'On Demand' or choose a defined, cyclic interval after which WinAdmin automatically fetches the messages. You can choose intervals between 30 seconds and two hours.

If you choose 'On Demand' then there are the following ways of initiating message retrieval. You can either choose the 'Poll Messages' command in the message collector's context menu or refresh the corresponding message list by pressing F5, or the 'Poll Messages' button below the list.

You can display the message list by choosing the "Show Message List" command which is also present in the message collector's context menu. Single-clicking the node for a message collector in the navigation tree displays the collector's message list.

- 'Message Type' dropdown box

Specifies the type of message that WinAdmin can expect from the queue.

- 'UTM Message'

This is a message generated by openUTM in PRINT format.

- 'Printable Text'

The messages can also be printed but may not be UTM messages.

- 'Binary Data'

This can be any type of data, including data which may not be printable. If you select this message type, WinAdmin converts the data at the UTM side of the connection into a printable, hexadecimal representation and displays the converted data in the message lists.

- 'Message Lifetime' dropdown box

This property determines how long WinAdmin retains the gathered messages.

- 'Don't Store'

The grouped messages are not stored in the configuration database. If you close the current collection then the messages are irrevocably lost. However, if you wish you can save individual messages manually in the configuration database. You do this by choosing the 'Store Selected Message' command from the context menu for a grouped message.

- 'Delete On Demand'

This is the opposite of 'do not store'. In this case, the messages are stored in the configuration database and are not deleted until this is explicitly requested. You do this by choosing the 'Delete Message' command from the context menu for a grouped message.

- 'Keep The Last <n>'

In this case, the last 100, 500, 1000 or 5000 grouped messages are stored in the database. If more messages than specified are present in the database then the oldest messages are deleted.

- 'Keep 1 Day/2 Days/1 Week/1 Month'

The messages are stored in the database for the specified period. The messages are deleted when this period expires. The period is measured from the time at which WinAdmin retrieves the messages, not the message generation time in openUTM.

- 'Current Application Run'

Only the messages from the current run of the relevant UTM application are stored in the database. The application start time is taken from the KDCADMI interface structure `kc_curr_par_str der` (`start_date_year`, `start_date_month`, `start_date_day`, `start_time_hour`, `start_time_min`, and `start_time_sec` fields). This is the (cold) start time of the UTM application (system time of the application host).

If the message is of 'UTM message' type then the time at which the message was sent by UTM (system time of the application host) is taken as the message time. Otherwise, the time at which WinAdmin retrieved the message from UTM (system time of the WinAdmin host) is taken as the message time.

- 'Consume Messages' dropdown box

With this parameter you can specify, whether the messages shall be consumed when reading them from the queue (this is the standard behaviour) or not. If the messages shall not be consumed, they will stay in the concerning queue after reading. By means of that it is possible, for example, to display the content of messages from the dead letter queue.

If you select 'No' for this parameter, you have to select the value 'do not store' in the dropdown box 'Message Lifetime', i.e. in this mode the messages can not be stored in the configuration database.

21.4 Message collectors in a cluster

In a UTM cluster, message collectors are also objects that are local to the node. However, unlike in Version V6.0, this version of WinAdmin ensures that the message collectors in a UTM cluster are "synchronized". This means that if a message collector is created, modified or deleted in a node application, then this action is automatically performed in all the activated node applications. As a result, the same message collectors are present with the same settings in all the activated node applications.

In addition, the cluster also possesses a global cluster list of message collectors. However, unlike the global cluster lists of UTM objects, the objects displayed here are not "assembled" global objects but, as indicated above, local node objects.

21.5 The grouped messages list

You can activate the grouped messages lists via the associated message collectors. You do this by choosing the 'Show Message List' command from the context menu for a message collector. The list that is now displayed contains all the messages for this message collector that are stored in the database and/or have been fetched by WinAdmin during the current session.

The columns have the following meaning:

Column	Meaning
Sequence No.	Sequential number of the message. Each message collector possesses a message counter which is incremented by one each time a message is polled. If you sort the messages on this column then you will see them in the sequence in which they were generated by UTM (provided that the sequence has not been changed using the message queue administration (DADM) (see section "Administering message

Column	Meaning
	queues" on page 100).
Polled at	Time when WinAdmin retrieved the message in the format YYYY-MM-DD hh:mm:ss.
Stored	Specifies whether or not the message is stored in the configuration database.
UTM No.	Specifies the number of a UTM message (e.g. K036).
Sent at	Specifies the time at which a UTM message was generated in the format YYYY-MM-DD hh:mm:ss..
Message	Message text. If the message type 'binary data' has been chosen for the associated message collector then the Binary Data is displayed in a printable hexadecimal form. If a message consists of multiple message segments then the individual segments are displayed in sequence here (separated by ">").

21.6 Tips and notes on configuration

If WinAdmin is to be used as a "message destination", you should make sure that the UTM application is able to satisfy the resulting resource requirements. You must set suitable configuration parameters, in particular the size of the page pool and the restart area. It is important to bear the following points in mind:

- Each message that is present in the queue occupies an entire page in the page pool. It is advisable to configure a suitably large page pool via the PGPOOL parameter of the MAX statement. The number of page pool pages specified here should be at least as large as the generated queue level for the message queue.
- You can limit the maximum number of messages that can be held in the queue via the QLEV parameter. This prevents a page pool overload. You are also recommended to set QMODE=WRAP-AROUND in order to ensure that openUTM continues to accept new messages and discards old messages from the queue when the configured queue level is reached.
- WinAdmin reads messages from the queue in the KDCWADMI program unit by means of a loop of DGET calls at the KDCS interface. On each DGET call, openUTM writes information to the restart buffer. If the size of the restart area is insufficient, the loop is interrupted and an additional communications step is necessary. For performance reasons, you should therefore use the RECBUF=(..., length) parameter of the MAX statement to set up a sufficiently large restart area: Approximately 30 bytes per message are required for restart information.

Tip:

It can be very useful to use WinAdmin as a message destination for messages that are generated by the program unit itself. The best solution here is probably to set up a separate UTM queue for this purpose, e.g. a Tac OWNMSGQ. Diagnostic messages, statistical application data, status messages etc. can then be sent to this queue and can be displayed directly by WinAdmin. Here again, it is advisable to send the messages to the queues in printable form. To do this you should set the message type to 'Printable Text' in the associated message collector.

Note:

When using this functionality offered by WinAdmin and openUTM, you should remember that the messages that WinAdmin fetches from the UTM queues actually disappear from the queues, if they are read with the 'Consume Messages = Yes' option chosen for the message collector. WinAdmin should therefore only poll queues with the 'Consume Messages = Yes' option that are precisely intended for this purpose, i.e. whose messages have been sent to the queue specifically for WinAdmin. If any other queues are polled by WinAdmin in this way, then this is sure to create havoc with the logic of the UTM application since important messages will suddenly disappear. If you need to monitor and administer other queues, then you can also do this from WinAdmin, see section "Administering message queues" on page 100.

Additional note:

If read access to the selected UTM queue is protected by an access list (configuration parameter Q-READ-ACL of the Tac or User queue), then you must make sure that WinAdmin possesses the privileges required in order to read from the queue.

22 Transaction-oriented administration jobs

Administration jobs are executed at various times in UTM. In general, there are three different execution times:

- immediate (**I**)

Many jobs are executed immediately in the administration program unit and are not transaction-oriented. These jobs are identified as type **I** in the administration guide. Here, **I** stands for "immediate".

- on termination of the transaction (**P**)

Other jobs do not take effect until the transaction within which the administration program unit was called is terminated. As long as the transaction has not been terminated, it can still be rolled back. In this case the administration job is not executed. This type of job is identified with a **P** in the administration guide. The **P** stands for "PEND" (Program End).

- independent (**A**)

In the case of this type of action, only a job for the subsequent execution of the action is initially created in UTM. This is then executed at a later time by UTM. Administration jobs of this type are not transaction-oriented. They are executed at a later time by UTM independently of the administration program unit. This type of job is identified with an **A** in the administration guide. The **A** stands for "announced" or for "action".

The openUTM "Administering Applications" manual indicates which of the three types described above each administration job belongs to. In many objects in which it is possible to modify several different properties, the individual changes belong to different types. For example, the change to the status of a Tac takes effect immediately (**I**), whereas a change to the lock code is transaction-oriented and belongs to the second type (**P**).

WinAdmin allows you to find out the modification characteristics of individual object properties. These are displayed in the tooltips in the tables (see section "Tooltips in lists and property dialogs" on page 159). This also applies to the tooltips that are displayed in property dialogs when the mouse pointer is located over the name of an object property for a short time. In addition, the "Table" dialog sheet in the property dialogs of UTM objects and UTM applications contains a table column entitled "Modification Characteristics".

22.1 Combining jobs in a transaction

WinAdmin makes it possible to group together the administration jobs for a UTM application in a transaction. openUTM then ensures that either all the jobs are executed or none at all. A transaction can be reset at any time if it has not yet been concluded. However, this is only possible for the second type of administration job (**P**).

Administration jobs can only be grouped together in a transaction if they relate to the same UTM application. Cross-application transactions are not supported.

Usually, modifying administration jobs are treated as separate transactions as in the past. The "Keep Transaction Open" checkbox that is present in all the relevant dialog boxes for modifying, creating or deleting objects as well as for other transaction-oriented jobs is used to combine multiple

administration jobs to form a single transaction. If this box is checked, openUTM executes the job without writing a synchronization point (the UTM-program unit terminates with PEND KP). The action is therefore only temporarily effective and can still be undone. It is not until a job for the termination of the transaction, for example a modify job without the selected "Keep Transaction Open" option, that all the jobs previously gathered within the transaction become effective and the transaction is concluded.

Please note:

WinAdmin is only able to group administration jobs together to form a single transaction if the UTM application is "suitable" for this. This is not the case if the administration TAC KDCWADMI was not generated with CALL=BOTH or the WinAdmin USER was not generated with RESTART=YES.

22.2 Displaying open administration jobs

If a UTM application still has an open transaction that contains one or more administration jobs that have not yet been executed then this is indicated by the presence of a red application icon (**A**) in the WinAdmin user interface. If the relevant application is a node application in a UTM cluster then the cluster icon is also displayed in red (**C**).

If at least one UTM application in the currently opened collection has an open transaction then the icon **A** is also displayed in the WinAdmin status bar.

In such cases, the context menu of this type of UTM application contains the 'Transaction' command which in turn possesses the 'Info' command in its submenu. This command can be used to open a dialog box which displays the administration actions that have not yet been executed. For each open action, this displays the type of action (for example, create, delete or modify) and, if applicable, the name and type of the corresponding object.

22.3 Terminating open administration jobs

The context menu of a UTM application in which open administration jobs still exist contains, alongside the 'Info' command described above, two further commands which can be used to close the open transaction:

■ 'Close'

If you choose this command, WinAdmin closes the open transaction and openUTM definitively applies the jobs that have been temporarily accepted.

■ 'Reset'

All the jobs of the second type (**P**) already issued in the open transaction are undone. The data is reset to its state prior to the start of the transaction.

You can also close open transactions in the Info dialog box which is displayed when you choose the above-mentioned 'Info' command. You do this using the radio buttons below the table in the dialog box which allow you to close or reset the transaction. If you choose one of the two radio buttons and then close the dialog box by clicking 'OK', the open transaction is either closed or reset.

If there is an open transaction for a UTM application and, at the same time, WinAdmin wants to run a job that has to terminate this transaction (e.g. the time-controlled polling of UTM messages by a message collector), then the Info dialog box is displayed and you are asked to decide whether WinAdmin should continue. Again, you do this using the radio buttons below the table. There are three possibilities:

- 'Cancel current request' radio button
Cancels the action that is currently being executed by WinAdmin and does not close the current open transaction.
If you choose this option then WinAdmin remembers this decision for the current open transaction and all other jobs that are performed automatically by WinAdmin are also canceled until the open transaction has been concluded. A message is output in the protocol window for each action that is canceled due to this procedure.
This action is also performed if you cancel the Info dialog box instead of exiting it with "OK".
- 'Close transaction and continue current action' radio button
This first of all closes the open transaction and then continues the action currently being performed by WinAdmin.
- 'Reset transaction and continue current action' radio button
This first of all resets the open transaction and then continues the action currently being performed by WinAdmin.

If an open collection is to be closed (this also occurs implicitly when another collection is opened or when WinAdmin is shut down) and if this collection contains at least one UTM application with an open transaction then WinAdmin outputs a message informing you of this and requests you to first terminate the open transaction (by closing or resetting it). The collection is not closed directly and WinAdmin is also not shut down if the program had to be terminated.

Please note:

While a transaction is open, WinAdmin eventually displays its modified status. However, from the point of view of openUTM the changes remain temporary as long as the transaction is still open. If, in this status, you again read data from the UTM application, this re-establishes the status prior to the start of the transaction.

openUTM resets transactions itself if there is no user input for an extended period. You can set this period in the TERMWAIT= parameter of the MAX statement (default: 10 minutes). An open transaction also occupies UTM application resources. Transactions should therefore be kept open only when required and for a limited period.

23 Administering message queues

WinAdmin allows you to administer asynchronous jobs which are buffered in the local message queues of a UTM application and awaiting processing. The entire functionality of the KDCS program interface DADM call is available in an easy-to-use form via the WinAdmin graphic user interface.

You can:

- display information about asynchronous jobs that are buffered in a message queue (including any user information).
User information will be requested from the UTM applications in a maximal length of 1024 bytes by WinAdmin.
- promote an asynchronous job within a message queue so that it is processed before all the other jobs in the queue.
- delete an asynchronous job or all the jobs in the message queue.
- Display the messages of asynchronous jobs in service-controlled queues
Service-controlled queues are temporary queues, Tac queues and User queues (see also section "Service-controlled queues in WinAdmin" on page 108).
The messages will be requested from the UTM applications in a maximal length of 1024 bytes by WinAdmin.

The following UTM objects may be attached to a message queue:

- Service-controlled queues
- Asynchronous Tacs
- Lterms
- Lpaps and Osi Lpaps

The context menu (right mouse button) of these objects in the WinAdmin administration part contains the 'Show Message Queue' command if the context menu refers to precisely one object. If you choose this command, WinAdmin displays all the asynchronous jobs in the message queue for the selected object in a new object list, the message queue list. It should be noted that in UTM clusters, this command is only present in object lists that apply locally to the node and not in those that apply globally in the cluster.

The following items of information are output for each job:

Column	Meaning
Job ID	The ID of the asynchronous job assigned by UTM.
Position	Position number of the asynchronous job within the message queue.

Column	Meaning
Submitting User	UTM user who initiated the asynchronous job.
Generation Time	Time at which the job was submitted in the form "nnn / hh:mm:ss", where nnn is the number of the day in the year (from 1 to 366) and hh:mm:ss is the time of day in hours, minutes and seconds (system time of the application computer).
Start Time	In the case of time-controlled jobs (DPUTs), this is the desired start time in the form "nnn / hh:mm:ss" (system time of the application computer).
Destination	Name of the message destination.
Destination Type	Type of the message destination.
Positive Acknowledge Job	Specifies whether there is a positive confirmation job for the asynchronous job.
Negative Acknowledge Job	Specifies whether there is a negative confirmation job for the asynchronous job.
User Information State	Provides information concerning the status of the user information for the asynchronous job. The following outputs are possible: 'not requested': The user information has not yet been requested by the WinAdmin user. 'incomplete': The user information has been requested and read but has not been retrieved in full because of its length. The user information will be requested from the UTM applications in a maximal length of 1024 bytes by WinAdmin. 'complete': The user information has been requested and read and is available in full.
User Information	User information concerning the asynchronous job in a maximal length of 1024 bytes. The following outputs are possible: '<not requested>': The user information has not yet been requested by the WinAdmin user. '<none>': The user information has been requested but there is no user information available for this asynchronous job. other values: Specify the requested and read user information either in printable or hexadecimal, printable format.
Message State	Provides information concerning the status of the message of the asynchronous job in a service-controlled queue. The following outputs are possible: 'not requested': The message has not yet been requested by the WinAdmin user. 'incomplete': The message has been requested and read but has not been retrieved in

Column	Meaning
	full because of its length. The message will be requested from the UTM applications in a maximal length of 1024 bytes by WinAdmin. 'complete': The message has been requested and read and is available in full.
Messages	Message of the asynchronous job in a service-controlled queue in a maximal length of 1024 bytes. The following outputs are possible: '<not requested>': The message has not yet been requested by the WinAdmin user. other values: Specify the requested and read message either in printable or hexadecimal, printable format.
Redelivery counter	Counts the number of deliveries of a message. For Service-controlled queues if the message was already read.
Data Read at...	Time when the data of the asynchronous job was retrieved from the UTM application (system time of the WinAdmin computer).

If a message queue list is active then the WinAdmin menu bar contains the 'Queue' menu with the following commands.

- 'Chane Sequence...'
This command is available if precisely one job has been selected and this job is not already at the head of the message queue (position 1). You use this command to tell openUTM to move the selected job to the front of the queue so that it is the next to be executed.
- 'Delete Job...'
Deletes the selected job(s) from the message queue.
- 'Delete All Jobs...'
Deletes all the jobs in the message queue.
- 'Move Job...'
This command is only available for a dead letter queue.
Moves the selected jobs from the dead letter queue to the original destination or to another destination. The destination can be assigned in the follow-up dialog. Available destinations are asynchronous Tacs or Tac queues with the exception of the Tacs "KDCDLETQ" and "KDCMSGTC".
- 'Move All Jobs...'
Same as 'Move Job...'. In contrast to this, it affects all jobs in the dead letter queue.

The first five of these commands are also present in the context menu that is opened if you select one or more jobs in the message queue list and right-click on a Job ID.

Above the message queue list, there is another area which you can use to request and display user information (if available) and the messages corresponding to the entries in the message queue. The

'Request' button retrieves the user information or the messages corresponding to the individual message queue entries from UTM. Check the 'Get Automatically' box if you want WinAdmin to retrieve the data from UTM automatically when the list is opened in the future. You can use the 'Printable' and 'Hexa Printable' radio buttons to define the way in which the user information or messages are displayed.

24 Administering printers and print jobs

WinAdmin allows you to administer the printers of a UTM application together with the associated print jobs. The entire functionality of the KDCS program interface PADM call is available in an easy-to-use form via the WinAdmin graphic user interface.

You can:

- display information concerning all the printers assigned to a printer control Lterm.
- lock and release printers.
- set up and terminate the connection to a printer.
- change the printer/Lterm partner assignment.
- activate or deactivate the confirmation mode for a printer control Lterm or individual printers.
- display information on the print jobs that are to be confirmed.
- confirm or repeat print jobs.

24.1 Printer control Lterms

To make it easy to identify printer control Lterms (Cterms), the Lterm administration lists now contain the column 'Printer Control Lterm' which can have the values 'Yes' or 'No'. By defining an appropriate display filter or sorting the list on this column, you can view all the printer control Lterms together.

The context menu for the printer control Lterms contains the following commands which are of use when administering printers and print jobs:

- 'Show Printer List'
Displays the printer list for the UTM application to which the selected printer control Lterm belongs (see below).
- 'Show Print Jobs to be Confirmed'
Displays the list of print jobs for confirmation for the UTM application to which the selected printer control Lterm belongs (see below).
- 'Activate Confirmation Mode' and 'Deactivate Confirmation Mode'
Activates or deactivates confirmation mode for all the printers that belong to this printer control Lterm.

Note: toggling the confirmation mode with this context menu does not change the transaction status of the UTM application. If there is an open transaction for the application when you toggle the mode then the transaction remains open and the changeover does not take effect until the transaction is concluded.

24.2 Administering printers

WinAdmin provides a list of all printers for UTM applications V5.1 and higher. You can access this either via the administration part of the navigation tree (in the same way as other UTM object lists) or via the context menu of a printer control Lterm (Cterms, see above).

In the Details view, the printer list contains the following information for each printer:

Column	Meaning
Printer ID	"Name" of the printer, i.e. the CID of the associated Pterm.
Application	UTM application to which the printer belongs.
Host	Host of the UTM application to which the printer belongs.
Cterm	Printer control Lterm to which the printer is assigned.
Status	Printer status ('On' or 'Off').
Connected	Specifies whether or not there is a connection to the printer
Confirmation Mode	Specifies whether the confirmation mode is active or inactive for this printer.
Pterm	The Pterm associated with the printer (the printer only "exists" by virtue of a Pterm's CID parameter. This CID is the printer ID).
Lterm	The Lterm assigned to the printer. You can use this Lterm's context menu ('Show Message Queue' command) to display any queued print jobs (for which there is as yet no confirmation) for this printer. You can, of course, call the related context menu directly from the printer list by right-clicking on the Lterm name.
Print Jobs	Number of print jobs for the printer (excluding time-controlled jobs, i.e. Dputs).
Dput Print Jobs	Number of time-controlled print jobs (Dputs) for the printer.
Data Read at...	Specifies the time at which the displayed data was retrieved from the UTM application (as in all administration object lists).

If you select one or more printers in the printer list and right-click on a printer ID, a context menu is opened. This contains the following printer administration commands:

- 'Show Print Jobs to be Confirmed'

This command is available if all the selected printers belong to the same UTM application. When you choose this command, a list of all the print jobs of the corresponding UTM application that are awaiting confirmation is displayed (see below).
- 'Activate Confirmation Mode' and 'Deactivate Confirmation Mode'

You use this command to activate or deactivate the confirmation mode for the selected printers.

Note: toggling the confirmation mode with this context menu does not change the transaction status of the UTM application. If there is an open transaction for the application when you toggle the mode then the transaction remains open and the changeover does not take effect until the transaction is concluded.

- 'Properties'

Opens a property dialog for the selected printer or printers. In this property dialog, you can lock or release printers, set up or interrupt printer connections or assign alternative Lterms. You can also activate or deactivate confirmation mode.

24.3 Administering print jobs

WinAdmin also provides a list of the print jobs of UTM applications as of V5.1 requiring confirmation. You can access this either via the administration part of the navigation tree (in the same way as other UTM object lists) or via the context menu of a printer control Lterm (Cterms, see above). The third way of displaying this list is via the printer's context menu (see above).

The Details view of the list of print jobs for confirmation displays the following information for each print job.

Column	Meaning
Job ID	Job ID of the print job.
Application	UTM application to which the print job belongs.
Host	Host of the UTM application to which the print job belongs.
Cterm	Printer control Lterm to which the printer at which this print job is to be printed is assigned.
Job Submitter	Submitter (UTM user) of the print job.
Printer ID	Printer ID of the printer at which this print job is to be printed.
Generation Time	Time at which the print job was generated in the form "nnn / hh:mm:ss", where nnn is the number of the day in the year (from 1 to 366) and hh:mm:ss is the time of day in hours, minutes and seconds (system time of the application computer).
Start Time	Desired start time for printing in the case of time-controlled jobs (Dputs) in the same format (system time of the application computer).
Positive Confirmation Job	Specifies whether there is a positive confirmation job for this print job (job complex).
Negative Confirmation Job	Specifies whether there is a negative confirmation job for this print job (job complex).
Data Read at...	Specifies the time at which the displayed data was retrieved from the UTM application (as in all administration object lists).

If you select at least one print job in this list, the two commands 'Confirm Print Output' and 'Repeat Print Output' become available in the context menu that is opened for the job ID (right mouse button).

Note: At any given time, a maximum of one print job can be awaiting confirmation for each printer defined in a UTM application and can thus be present in this list. If further print jobs are queued for this printer then these continue to be present in the message queue of the Lterm that is assigned to the printer (not the printer control Lterm of the printer!). You can display this message queue via the context menu for the corresponding Lterm ('Show Message Queue' - for example in the printer list, see above).

25 Service-controlled queues in WinAdmin

So-called "service-controlled queues" have been introduced in version 5.1 of UTM (these are frequently simply referred to as "queues"). In the corresponding openUTM manuals you will find a detailed description of precisely what a service-controlled queue is, under what circumstances it can be of use and how it should be handled.

There are three types of service-controlled queues. These differ primarily in the way they are created. However, there is generally no difference in their function or mode of use. The three variants are:

- Temporary queues

Temporary queues are generated and deleted again in UTM program units. This is done using the KDCS calls QCRE (Create) and QREL (Delete).

- Tac queues

Tac queues are generated in the same way as Tacs, i.e. either with a KDCDEF "TAC" statement (with TYPE=Q) or dynamically by means of a KDCADMI program interface call. Tac queues are represented by a special UTM object variant of type TAC.

- User queues

As of version 5.1, every user simultaneously represents a potentially exploitable user queue unless the QLEV property is explicitly set to zero. A "User" type object is generated either with a KDCDEF "USER" statement or dynamically by means of a KDCADMI program interface call.

WinAdmin also uses the service-controlled queues when it is specified as a message destination (see section "WinAdmin message destination - message collectors" on page 91).

The section below describes how WinAdmin supports service-controlled queues and indicates where you may encounter them.

25.1 The queue list

UTM applications of version 5.1 or higher contain the "Queues" entry in the navigation tree (in the same way, for example, as the "Users" and "Tacs" entries). You can use this entry to display a list of all the (potential) queues of an active UTM application. The administration queue list contains all the existing temporary queues for the UTM application, all the TAC queues (TAC...,TYPE=Q) and all the users whose QLEV is not equal to zero. The Tac and user queues that are listed here are also present in the corresponding Tac and user lists since they also constitute objects of TAC and USER type. Accordingly, the list only presents the queue-related properties of these objects. These are as follows:

Column	Meaning
Name	Name of the service-controlled queue.
Application	UTM application to which the queue belongs.
Host	Host of the UTM application to which the queue belongs.
Type	Queue type, i.e. 'temporary queue', 'Tac queue' or 'user queue'. The type can, of course, also be identified by the icon displayed in the Name column.
Qlev	Queue level of the queue. Specifies the maximum number of messages that may be present in the queue. Qlev 32767 means that the number of messages is unlimited.
Queue Length	Number of messages present in the queue at the time the data was retrieved from the UTM application.
Queue Mode	Queue mode, i.e. either 'Standard' or 'Wrap-around'. Standard means that no new messages are accepted when the value set for Qlev is reached. If Wrap-Around is set then the oldest messages are deleted from the queue when Qlev is reached and a new message is sent to the queue.
Queue Read Acl	Access list which regulates read access to the queue. Temporary queues cannot be protected and therefore have no queue read access list ("<.>" is displayed in the list).
Queue Read Access Roles	Access roles present in the queue read access list. Here again, "<.>" is displayed for temporary queues.
Queue Write Acl	Access list which regulates write access to the queue. Temporary queues cannot be protected and therefore have no queue write access list ("<.>" is displayed in the list).
Queue Write Access Roles	Access roles present in the queue write access list. Here again, "<.>" is displayed for temporary queues.
Data Read at...	Specifies the time at which the displayed data was retrieved from the UTM application (as in all administration object lists).

The context menus of the queue objects in this list contain the 'Show Message Queue' command. If you choose this command, WinAdmin displays the message queue for the corresponding object (see section "Administering message queues" on page 100).

25.2 Creating and deleting objects

You create Tac and user queues in the same way as "normal" Tacs and users. In the case of Tac queues, you must set the 'Type' property to 'Queue'. In both cases, you must, of course, pay attention to the 'Qlev' and 'Queue Mode' properties. In addition, you can restrict access to the queue that you want to create on the 'Queue Read Access' and 'Queue Write Access' dialog sheets.

WinAdmin is also able to create temporary queues. To do this, right-click on the 'Queues' entry in the administration part of the navigation tree.

All queues can be deleted again using the corresponding command in the object's context menu (provided that the object in question is currently available for deletion).

25.3 Displaying a queue's message queue

The context menu of every administration queue object contains the 'Show Message Queue' command. If you choose this command, WinAdmin displays the message queue list for the corresponding queue object (see section "Administering message queues" on page 100).

25.4 Queues in the Security view

The Security view (see section "The Security view" on page 87) contains an additional "Queues" node which displays the possible (or impossible) accesses to Tac and user queues. No temporary queues are depicted here since these cannot be protected.

Unlike the Tac lists which contain not only dialog and asynchronous Tacs but also the Tac queues (TAC...,TYPE=Q), no Tac queues are listed under the "Tacs" node in the Security view. These can be found under the corresponding "Queues" node.

26 Statistics

WinAdmin offers functions for collecting statistical data on the UTM applications. The collected data can be displayed in a graph, copied to the clipboard and then displayed in Microsoft Excel, for example, or it can be saved to a (text) file.

The data collected is automatically stored in the configuration database.

What are known as "statistic collectors" are responsible for collecting statistic values. The statistic collectors are able to collect both global data from the UTM applications (as displayed in the property dialog of a UTM application) and data relating to individual UTM objects (e.g. Tacs). As a result, the "in-queues" of a specific Tac, for example, can now be monitored using WinAdmin.

An upper and a lower threshold can also be defined for each statistic collector. If the upper or lower value is exceeded, a "threshold alert" is triggered by WinAdmin.

In the view of WinAdmin the statistic collectors are objects like any other UTM objects. They are stored in the configuration database so that they are available to other WinAdmin instances that work with the same database.

Once first contact has been established with the UTM application, defined and activated statistic collectors automatically begin to collect values after the collection has been opened.

26.1 Statistics window

The collection of statistical data is controlled in the Statistics window. The window is activated in exactly the same way as other UTM object lists (e.g. the Tac list). For this purpose the navigation tree has a 'Statistics' entry below the UTM applications and UTM Clusters. Depending on context this allows the Statistics window of a concrete UTM application, a concrete UTM cluster, of all stand-alone applications on a concrete host, or all stand-alone applications included in the active collection to be displayed.

Alternatively, the Statistics window of all stand-alone applications of the active collection can be displayed using the corresponding icon on the highest level of the 'Stand-Alone' subtree in the navigation window.

The Statistics window is divided into an upper section, which displays all available statistic collectors with their current settings, and a lower section which displays a simple graphical representation of the statistics of the collectors selected in the list.

You can modify the way the Statistics window is displayed so that either the list with the collectors **and** the graph (this is the state when you open the window) or **only** the list containing the collectors or **only** the graph is displayed. You can change between the three different display modes by pressing the keyboard shortcut **Ctrl+U**. Alternatively, you can also use the two small icons in the thin bar between the two areas to switch between the display of the collector list only or the graph only.

























You can also adjust the size of the two sections of the Statistics window by dragging the bar separating the list and the graph.

26.1.1 Collector list

The upper section of the Statistics windows shows the list of all collectors that belong to the corresponding window. This list can be operated in exactly the same way as a UTM object list (see section "Object lists" on page 42).

The statistic values of the collectors selected in this list are shown in the statistics graph in the lower half of the Statistics window.

The list has the following columns:

Column	Meaning																
Type	<p>Icon and type of the statistic collector and specification of the unit of measurement (# for number, % for percent).</p> <p>Possible icons:</p> <table border="1"> <tr> <td></td> <td>Collector is inactive, it can be activated using its context menu or the properties dialog.</td> </tr> <tr> <td></td> <td>Collector is sleeping (see 'State' column).</td> </tr> <tr> <td></td> <td>Collector is active.</td> </tr> <tr> <td></td> <td>Collector is active and is currently requesting a statistic value.</td> </tr> <tr> <td></td> <td>Active collector for which the last statistic value collected exceeded the defined upper threshold.</td> </tr> <tr> <td></td> <td>Active collector for which the last statistic value collected exceeded the defined lower threshold.</td> </tr> <tr> <td></td> <td>Active collector whose assigned "collecting" object does not exist or no longer exists in the relevant UTM application.</td> </tr> <tr> <td></td> <td>Active collector for which the last attempt to collect a statistic value failed. Possible cause: The statistic value to be collected is not available in the operating system version or variant of the relevant UTM application.</td> </tr> </table>		Collector is inactive, it can be activated using its context menu or the properties dialog.		Collector is sleeping (see 'State' column).		Collector is active.		Collector is active and is currently requesting a statistic value.		Active collector for which the last statistic value collected exceeded the defined upper threshold.		Active collector for which the last statistic value collected exceeded the defined lower threshold.		Active collector whose assigned "collecting" object does not exist or no longer exists in the relevant UTM application.		Active collector for which the last attempt to collect a statistic value failed. Possible cause: The statistic value to be collected is not available in the operating system version or variant of the relevant UTM application.
	Collector is inactive, it can be activated using its context menu or the properties dialog.																
	Collector is sleeping (see 'State' column).																
	Collector is active.																
	Collector is active and is currently requesting a statistic value.																
	Active collector for which the last statistic value collected exceeded the defined upper threshold.																
	Active collector for which the last statistic value collected exceeded the defined lower threshold.																
	Active collector whose assigned "collecting" object does not exist or no longer exists in the relevant UTM application.																
	Active collector for which the last attempt to collect a statistic value failed. Possible cause: The statistic value to be collected is not available in the operating system version or variant of the relevant UTM application.																
UTM Application	Name of the UTM application to which the collector belongs.																
Host	Name of the host on which the UTM application is running.																
Object Type Object Name	Type of object from which data is collected. Name of the object that is assigned to the statistic collector.																
Activated	Indicates whether the statistic collector has been activated or not.																

Column	Meaning
State	<p>State of the statistic collector. Possible values:</p> <p>Inactive The collector has been deactivated and is not collecting any statistic values.</p> <p>Active The collector is collecting statistic values.</p> <p>Sleeping Indicates that the statistic collector was activated but is not currently collecting statistic values because the application has not yet been contacted, is not available, or is no longer available.</p> <p>Without object The “collecting” object assigned to the collector is not available or is no longer available in the UTM application.</p>
Interval	Specifies the period between two polls for statistic values.
Automatic Wake-Up	Indicates whether the statistic collector automatically switches from the sleeping to active state or not when the application is available.
Statistic value lifetime	<p>Limits the number of statistic values stored in the configuration database.</p> <p>Don't store The statistic values are not stored in the configuration database. If you close the current collection then the statistic values are irrevocably lost.</p> <p>Delete On Demand In this case, the statistic values are stored in the configuration database and are not deleted until this is explicitly requested.</p> <p>Keep The Last <n> In this case, the last 100, 500, 1000 or 5000 statistic values are stored in the database. If more values than specified are present in the database then the oldest values are deleted.</p> <p>Keep 1 Day/2 Days/1 Week/1 Month The statistic values are stored in the database for the specified period. The values are deleted when this period expires.</p> <p>Current Application Run Only the statistic values from the current run of the relevant UTM application are stored in the database. The application start time is taken from the KDCADMI interface structure kc_curr_par_str.</p>
Values	Specifies the number of statistic values polled for the collector in the current WinAdmin session and possibly also the number of all available values (in parentheses).
Lowest Value Last Value	Specifies the lowest, last and highest value measured until now in the current WinAdmin session. If no successful poll has yet taken

Column	Meaning
Highest Value	place, '<.>' is shown here for each of the values.
Next Sample At	Specifies the time at which the statistic value will next be polled. If no poll has yet taken place, '<.>' is shown here.
Upper Threshold	The upper threshold set for the collector or '<without>' if no upper threshold has been defined.
Upper Threshold Exceeded	Indicates whether the last value polled exceeds an upper threshold that may have been defined.
Upper Threshold Alert	Type of alert when the upper threshold is exceeded.
Lower Threshold	The lower threshold set for the collector or '<without>' if no lower threshold has been defined.
Lower Threshold Exceeded	Indicates whether the last value polled exceeds a lower threshold that may have been defined.
Lower Threshold Alert	Type of alert when the lower threshold is exceeded.
Remark	Comment on the statistics collector.

26.1.2 Statistics graph

The lower half of the Statistics window displays a graph for the collected statistic values of the statistic collectors selected in the upper half of the window.

No statistics graph is displayed if no is selected in the list.

The graph window has the following elements.

- Dropdown box 'Time Range'

Here you can specify the time range for which the graph is displayed

- Graph

Each polled statistic value appears as a node in the graph, and adjacent nodes are connected by a line. The x axis (horizontal) of the graph represents the time, the y axis (vertical) the scaled values of the statistic collector.

Below the graph, there is a key with the depictions of the curves in the graph together with the associated statistic collector names.

If the curves of more than one statistic collector are displayed in the graph then the individual curves are displayed in different colors and the nodes in the graph are also displayed differently. The key below the graph indicates which curves correspond to which statistic collectors.

A right-click in the graph opens a context menu (see below).

The current graph can be printed out (see below).

The actions available in the statistics graph are described briefly below.

Action	Description
Decrease/increase time range	You can use the 'Time Range' dropdown box above the graph to select the required time range.
Select a section from the graph	You can select a section in the graph by left-clicking in the graph and then dragging the mouse pointer while keeping the left mouse button held down. The display then zooms in on this section when you release the mouse button. Note: After you have done this, the time range of the graph generally no longer corresponds to the value indicated in the 'Time Range' dropdown box.
Move graph extract	If you left-click in the graph with the Ctrl key held down and then move the mouse with the mouse button held down, the extract from the graph that is displayed also moves.
Show context menu for graph	Right-click in the graph to display the corresponding context menu.
Copy graph	You can use the 'Copy' command in the graph's context menu to copy the graph to the clipboard.
Save graph	You can use the 'Save as...!' command in the graph's context menu to save the graph in a (Portable Network Graphics) PNG file.
Print graph	You can use the 'Print...!' command in the graph's context menu to print the graph.
Zoom in/out	If the mouse pointer is positioned over the graph then you can use the mouse wheel to zoom into or out of the graph. The graph's context menu also contains the "Zoom In" and "Zoom Out" commands which you can use to zoom on a specific axis.

Action	Description
Automatic scaling	If you choose a different statistic collector from the list or set a different time range in the 'Time Range' dropdown box then the graph is scaled automatically. The graph's context menu also contains the "Auto Range" command which you can use to scale automatically on a specific axis.
Graph node tooltip	If you leave the mouse pointer over a node in the graph for a short time then a tooltip describing the graph node below the pointer is displayed (title of curve, time and value).

26.2 Statistic collectors

26.2.1 Creating statistic collectors

Statistic collectors can be created in the same way as other UTM objects.

- Using the administration context menu of a UTM application by selecting the 'Create' command and then the object type 'Statistic Collector'.
- Using the context menu of a 'Statistics' icon in navigation tree ('Create Statistic Collector' command).
- Using the context menu of another statistic collector ('Create Statistic Collector' command).

In addition to these standard ways of creating a statistic collector, it is possible to create a collector specifically for another UTM object (for a Tac, for example). The command 'Create Statistics Collector for <Object Type> "<Name>"' is available for this purpose in the context menu of the corresponding objects.

This option is available for the following UTM object types.

- Con
- Printer
- Lpap
- Ltac
- Lterm
- Mux
- Osi Lpap
- Pterm
- Tac
- Tacclass
- Tpool
- User

26.2.2 Deleting statistic collectors

Like other UTM objects, statistic collectors can be deleted using their context menu ('Delete Statistics Collector' command).

When a collector is deleted, all statistic values associated with the collector are deleted from the configuration database.

26.2.3 Statistic collectors in clusters

In a UTM cluster, statistic collectors are also objects that are local to the node. However, unlike in Version V6.0, this version of WinAdmin ensures that the statistic collectors in a UTM cluster are "synchronized". This means that if a statistic collector is created, modified or deleted in a node application, then this action is automatically performed in all the activated node applications. As a result, the same statistic collectors are present with the same settings in all the activated node applications.

In addition, the cluster also possesses a global cluster list of statistic collectors. However, unlike the global cluster lists of UTM objects, the objects displayed here are not "assembled" global objects but, as indicated above, local node objects.

26.2.4 Context menu of statistic collectors

The context menu of statistic collectors contains the following commands.

Menu command	Meaning
Create Statistics Collector	Displays the property dialog for creating a new statistic collector. The collector selected is used as a template.
Delete Statistics Collector "<Name>" ... Delete Statistics Collectors...	Deletes the selected collectors.
Activate Deactivate	Activates or deactivates the selected collectors.
Wake Up	Checks the availability of the UTM applications to which the selected sleeping collectors belong. If they are available, the corresponding collectors are woken and begin to collect data.
Copy Statistics Values	Copies all statistic values collected by the selected collectors to the clipboard.
Save Statistics Values	Saves all statistic values collected by the selected collectors to a file.
Delete Statistics Values...	Deletes the statistic values of the selected collectors from the configuration database. A dialog box with various delete options is

Menu command	Meaning
	opened when this menu command is selected.
Set Sample Interval...	Allows the sample interval of the selected collectors to be set.
Threshold Alert/Test Upper Alert Threshold Alert/Test Lower Alert	If an alert has been defined for an upper/lower threshold, it can be tested using these commands. Test results are always output in English.
Properties of Statistics Collector "<Name>" Properties of Selected Statistics Collector	Displays the property dialog of the selected collectors.

26.3 Threshold alert

An upper and a lower threshold can be assigned to each individual statistic collector on the 'Threshold' sheet of the property dialog. The thresholds defined there are also displayed in the statistics graph if the graph shows only the values of a single collector. WinAdmin outputs a threshold alert if values exceed the set thresholds.

This has the following effect.

- Collectors whose last value collected is above the upper threshold are shown with this icon:



- Collectors whose last value collected is below the lower threshold are shown with this icon:



- If WinAdmin is not the active application on the computer (because, for example, WinAdmin is currently minimized), the WinAdmin main window and/or the WinAdmin icon in the taskbar flash until WinAdmin becomes the active application.

If an 'Alert' has also been assigned to the particular threshold (this is also done on the 'Threshold' sheet of the property dialog), the specified action is performed by WinAdmin when a value exceeds the corresponding threshold for the **first time**.

Any action already defined can be selected as an alert. How and which actions can be defined is described in the "Actions" section on page 139.

If an action of the type 'Call Script' were selected as a threshold alert, WinAdmin would pass the following arguments to the specified script.

- /T:<Time>
<Time>: The time when the threshold alert occurred. Format: YYYYMMDDhhmmss.
YYYY: Year (always four digits)
MM: Month (01 to 12)
DD: Day of the month (01 to 31)
hh: Hour (00 to 23)
mm: Minute (00 to 59)
ss: Second (00 to 59)
- /A:<Application>
<Application>: Name of the UTM application as defined in WinAdmin.
- /H:<Host>
<Host>: Name of the host as defined in WinAdmin.
- /O:<Object type>
<Object type>: The type of the "collecting" UTM objects. The following types are possible.
 - APPLICATION-PARAMETER
 - CON
 - LPAP
 - LTERM
 - MUX
 - OSI-LPAP
 - PRINTER
 - PTERM
 - TAC
 - TACCLASS
 - TPOOL
 - USER
- /N:<Object name>
<Object name>: Name of the "collecting" object. <Object name> is blank for the object type "APPLICATION-PARAMETER".
- /S:<Statistic type>
<Statistic type>: Type of the statistic value.
When the type is "Connected users (#)" for instance, this will be "CONNECTEDUSERS".
The values are listed in the table in section "Statistic types supported" on page 120.
- /V:<Value>
<Value>: Collected value that triggered the threshold alert. Percentage values include the percentage sign.

The arguments are passed in the order specified above.

26.4 Statistic values - saving, deleting, exporting

The values collected by the statistic collectors are automatically saved in the configuration database depending on the selected statistic value lifetime so that they are available in subsequent WinAdmin sessions and for other WinAdmin instances. Values in the database can be deleted using the context menu of the associated collectors. Selecting the 'Delete Statistic Values...' command displays a dialog box in which the delete mode can be defined. Available delete modes are:

- Delete all values
- Retain the most recent <n> values
- Delete values older than <...>

It is also possible to specify that values are to be saved in a (text) file before they are deleted in the database.

The collected values can either be copied to the clipboard or written to a file. Values are copied and saved in text form. The statistic values copied/saved in this way can then be pasted or imported into other applications (e.g. Microsoft Excel).

The context menu of the statistic collector provides the 'Copy Statistic Values' and 'Save Statistic Values...' commands for this purpose. These commands apply to all selected collectors. If the Statistics window is active, the File menu also includes the 'Save Statistic Values' command. This command applies to all statistic collectors of the active Statistics window and likewise saves the relevant statistic values to a file.

The 'Settings...' command in the Extras menu of WinAdmin can be used to specify that the collected statistic values are to be saved to a text file automatically when the collection is closed (see "The 'File Transfer' sheet" on page 166). The files are saved with the name

"Stat.<collection-name>.<session-number>.txt"

in the directory

"<WinAdmin-directory>\Statistic"

or in the directory selected on the 'General' sheet of the 'Settings' dialog (see "The 'General' sheet" on page 164).

26.5 Statistic types supported

The table below lists all statistic types supported by WinAdmin, the object type of the associated "collecting" object and the "Source field" from which the information is obtained.

Unless otherwise specified, the source field specifies the name of the structure and the field name within the structure at the administration programming interface.

Refer to the corresponding UTM manual for the exact meaning.

The two names listed in the "Type" column are the statistic type used at the WinAdmin GUI and the one that will be transferred to a procedure during a threshold alert as the "/S" argument.

Type	Object type	Source field
Terminal Input Messages (#) TERMINOUTMSGS	Application	kc_curr_par_str term_input_msgs
Terminal Output Messages (#) TERMOUTPUTMSGS	Application	kc_curr_par_str term_output_msgs
Current Max. Asyntasks (#) CURRMAXASYNTASKS	Application	kc_curr_par_str curr_max_asyntasks
Current Max. Tasks in PGWT (#) CURRMAXTASKSINPGWT	Application	kc_curr_par_str curr_max_tasks_in_pgwt
Current Tasks (#) CURRTASKS	Application	kc_curr_par_str curr_tasks
Current Active Asyntasks (#) CURRASYNTASKS	Application	kc_curr_par_str curr_asyntasks
Current Tasks in PGWT (#) TASKSINPGWT	Application	kc_curr_par_str curr_tasks_in_pgwt
Tasks Waiting in PGWT (#) TASKSWAITINGINPGWT	Application	kc_curr_par_str tasks_waiting_in_pgwt
Connected Users (#) CONNECTEDUSERS	Application	kc_curr_par_str connected_users
Open Dialog Services (#) OPENDIALSERVICES	Application	kc_curr_par_str open_dial_services
Open Asyn. Services (#) OPENASYNSERVICES	Application	kc_curr_par_str open_asyn_services
Dialog Transactions / sec (#) DIALTAPERSEC	Application	kc_curr_par_str dial_ta_per_100sec
Asyn. Transactions / sec (#) ASYNTAPERSEC	Application	kc_curr_par_str asyn_ta_per_100sec
Dialog Steps / sec (#) DIALSTEPPERSEC	Application	kc_curr_par_str dial_step_per_100sec
Max. Dialog Transactions / sec (#) MAXDIALTAPERSEC	Application	kc_curr_par_str max_dial_ta_per_100sec
Max. Asyn. Transactions / sec (#) MAXASYNTAPERSEC	Application	kc_curr_par_str max_asyn_ta_per_100sec
Max. Dialog Steps / sec (#) MAXDIALSTEPPERSEC	Application	kc_curr_par_str max_dial_step_per_100sec
Maximum Page Pool Size (%) MAXPOOLSIZE	Application	kc_curr_par_str max_pool_size
Current Page Pool Size (%) CURRPOOLSIZE	Application	kc_curr_par_str curr_pool_size

Type	Object type	Source field
Average Page Pool Size (%) AVGPOOLSIZE	Application	kc_curr_par_str avg_pool_size
Cache Hit Rate (%) CACHEHITRATE	Application	kc_curr_par_str cache_hit_rate
Cache Wait For Buffer (%) CACHEWAITBUFFER	Application	kc_curr_par_str cache_wait_buffer
Unprocessed Asyn. Tacs (#) UNPROCATACS	Application	kc_curr_par_str unproc_atacs
Unprocessed Print Jobs (#) UNPROCPRINTS	Application	kc_curr_par_str unproc_prints
Waiting Dputs (#) WAITDPUTS	Application	kc_curr_par_str wait_dputs
Abnormally Terminated Services (#) ABTERMSERVICES	Application	kc_curr_par_str abterm_services
Wait For Resources (%) WAITRESOURCES	Application	kc_curr_par_str wait_resources
Max. Wait For Resources (%) MAXWAITRESOURCES	Application	kc_curr_par_str max_wait_resources
Wait For System Resources (%) WAITSYSTEMRESOURCES	Application	kc_curr_par_str wait_system_resources
Max. Wait For System Resources (%) MAXWAITSYSTEMRESOURCES	Application	kc_curr_par_str max_wait_system_resources
Deadlocks (#) DEADLOCKS	Application	kc_curr_par_str deadlocks
Periodic Writes (#) PERIODICWRITES	Application	kc_curr_par_str periodic_writes
Pages Per Periodic Write (#) PAGEPWRITE	Application	kc_curr_par_str pages_pwrite
Logfile Writes (#) LOGFILEWRITES	Application	kc_curr_par_str logfile_writes
Current Job-Receiving Services (%) CURRJR	Application	kc_curr_par_str curr_jr
Max. Job-Receiving Services (#) MAXIMUMJR	Application	kc_curr_par_str maximum_jr
Program File-Generation-Group (#) PROGRAMFGG	Application	kc_curr_par_str program_fgg
Uslog File-Generation-Group (#)	Application	kc_curr_par_str

Type	Object type	Source field
USLOGFGG		uslog_fgg
Maximum Workload (%) MAXLOAD	Application	kc_curr_par_str max_load
Current Workload (%) CURRLOAD	Application	kc_curr_par_str curr_load
Average Saved Pages per Data Compression (#) AVGSAVEDPGSBYCOMPRESSION	Application	kc_curr_par_str avg_saved_pgs_by_compr
Maximum Cluster Page Pool Size (%) CLUSTERCURRPARMAXCPGPOOLSIZE	Application	kc_cluster_curr_par_str max_cpgpool_size
Current Cluster Page Pool Size (%) CLUSTERCURRPARCURRCPGPOOLSIZE	Application	kc_cluster_curr_par_str curr_cpgpool_size
Average Cluster Page Pool Size (%) CLUSTERCURRPARAVGCPGPOOLSIZE	Application	kc_cluster_curr_par_str avg_cpgpool_size
Current Reserved Cluster Pagepool Pages (#) CLUSTERCURRPARNODERESERVEDCPGPOOLPAGES	Application	kc_cluster_curr_par_str node_reserved_cpgpool_pages
Con: Connection Time (min) (#) CON_CONTIMEMIN	Con	kc_con_str contime_min
Con: Messages (total) (#) CON_LETTERS	Con	kc_con_str letters
Con: Connection Losses (#) CON_CONBAD	Con	kc_con_str conbad
Printer: DPUT Print Jobs (#) PRINTER_NBRDPUTPRINTJOBS	Printer	PADM: kc_ptinf kcdpmsgs
Printer: Print Jobs (#) PRINTER_NBRPRINTJOBS	Printer	PADM: kc_ptinf kcfpmsgs
Lpap: Output Queue (#) LPAP_OUTQUEUE	Lpap	kc_lpap_str out_queue
Lpap: Number of Dputs (#) LPAP_NBRDPUTS	Lpap	kc_lpap_str nbr_dputs
Ltac: Used (#) LTAC_USED	Ltac	kc_ltac_str used
Lterm: Nbr. of Dputs (#) LTERM_NBRDPUTS	Lterm	kc_lterm_str nbr_dputs
Lterm: Security Violations (#) LTERM_SECCOUNTER	Lterm	kc_lterm_str seccounter

Type	Object type	Source field
Lterm: Incounter (#) LTERM_INCOUNTER	Lterm	kc_lterm_str incounter
Lterm: Output Queue (#) LTERM_OUTQUEUE	Lterm	kc_lterm_str out_queue_ex
Mux: Currently Connected (#) MUX_ACTCON	Mux	kc_mux_str actcon
Mux: Letters (Total) (#) MUX_LETTERS	Mux	kc_mux_str letters
Mux: Letters (Input) (#) MUX_INCOUNT	Mux	kc_mux_str incnt
Mux: BCAM Waits (#) MUX_WAIT	Mux	kc_mux_str wait_go
Mux: Shortage (#) MUX_SHORTAGE	Mux	kc_mux_str shortage
Mux: Send Retried (#) MUX_RTRYO	Mux	kc_mux_str rtryo
Mux: Receive Retried (#) MUX_RTRYI	Mux	kc_mux_str rtryi
Osi Lpap: Active Connections (#) OSILPAP_CONNECTNUMBER	Osi Lpap	kc_osi_lpap_str connect_number
Osi Lpap: Output Queue (#) OSILPAP_OUTQUEUE	Osi Lpap	kc_osi_lpap_str out_queue_ex
Osi Lpap: Number of Dputs (#) OSILPAP_NBRDPUTS	Osi Lpap	Kc_osi_lpap_str nbr_dputs
GSSB Pages (#) NRGSSBPAGES	Pagepool	kc_pagepool_str gssb_pages
LSSB Pages (#) NRLSSBPAGES	Pagepool	kc_pagepool_str lssb_pages
TLS Pages (#) NRTLSPAGES	Pagepool	kc_pagepool_str tls_pages
ULS Pages (#) NRULSPAGES	Pagepool	kc_pagepool_str uls_pages
Pages for Dialog Conversations (#) NRDIALCONVPAGES	Pagepool	kc_pagepool_str dial_conv_pages
Pages for TACCLASS Queues (#) NRTACCLASSPAGES	Pagepool	kc_pagepool_str tacclass_pages
Pages for FPUT Management (#)	Pagepool	kc_pagepool_str

Type	Object type	Source field
NRFPMPAGES		fpmm_pages
Pages for Asyn Messages (#) NRFPMPAGES	Pagepool	kc_pagepool_str fput_pages
Pages for MSGTAC Messages (#) NRMSGTACMPAGES	Pagepool	kc_pagepool_str msgtac_pages
LPUT Pages (#) NRLPUTMPAGES	Pagepool	kc_pagepool_str lput_pages
Pages for Physical Output Messages (#) NRPHYSMSGMPAGES	Pagepool	kc_pagepool_str phys_msg_pages
Pages for OSI-TP Log Records (#) NRLOGRECPAGES	Pagepool	kc_pagepool_str log_rec_pages
Pages for Reset Messages (#) NRRESETMSGMPAGES	Pagepool	kc_pagepool_str reset_msg_pages
Other Pages (#) NROTHERMPAGES	Pagepool	kc_pagepool_str other_pages
Pterm: Connection Losses (#) PTERM_CONBAD	Pterm	kc_pterm_str conbad
Pterm: Letters (Total) (#) PTERM_LETTERS	Pterm	kc_pterm_str letters
Pterm: Contime (min) (#) PTERM_CONTIMEMIN	Pterm	kc_pterm_str contime_min
Tac: Nbr. of Ack. Jobs (#) TAC_NBRACKJOBS	Tac	kc_tac_str nbr_ack_jobs
Tac: Nbr. of Dputs (#) TAC_NBRDPUTS	Tac	kc_tac_str nbr_dputs
Tac: CPU Time (msec) (#) TAC_TACCPUMSEC	Tac	kc_tac_str taccpu_msec
Tac: CPU Time (microsec) (#) TAC_TACCPUMICROSEC	Tac	kc_tac_str taccpu_micro_sec
Tac: DB Elapsed Time (msec) (#) TAC_DBELAPMSEC	Tac	kc_tac_str db_elap_msec
Tac: DB-Counter (#) TAC_DBCOUNTER	Tac	kc_tac_str db_counter
Tac: Errors (#) TAC_ERROR	Tac	kc_tac_str number_errors_ex
Tac: Used (#) TAC_USED	Tac	kc_tac_str used

Type	Object type	Source field
Tac: Tac Units / min (#) TAC_USEDPERMIN	Tac	kc_tac_str used
Tac: In-Queue (#) TAC_INQUEUE	Tac	kc_tac_str in_queue_ex
Tac: Elapsed Time (msec) (#) TAC_TACELAPMSEC	Tac	kc_tac_str tac_elap_msec
Tac: Nbr. of Transaction Commits (#) TAC_NBRTACOMMITS	Tac	kc_tac_str nbr_ta_commits
Tacclass: Waiting Messages (#) TACCLASS_WAITINGMSGS	Tacclass	kc_tacclass_str waiting_msgs
Tacclass: Average Wait Time (msec) (#) TACCLASS_AVGWAITTIMEMSEC	Tacclass	kc_tacclass_str avg_wait_time_msec
Tpool: Currently Connected (#) TPOOL_ACTCON	Tpool	kc_tpool_str actcon
User: Output Queue (#) USER_OUTQUEUE	User	kc_user_str out_queue_ex
User: Number of Executed Service Routines (#) USER_NUMBERTACS	User	kc_user_str number_tacs
User: CPU Time Used (sec) (#) USER_CPUTIMESEC	User	kc_user_str cputime_sec
User: CPU Time Used (msec) (#) USER_CPUTIMEMSEC	User	kc_user_str cputime_msec
User: Signed Clients (#) USER_CLIENTSSIGNED	User	kc_user_str clients_signed
User: Open Asynchronous Services (#) USER_ASYNSERVICES	User	kc_user_str asyn_services
User: Security Violations (#) USER_SECCOUNTER	User	kc_user_str seccounter

27 Security - users and authorizations

Access to WinAdmin and the type of actions that can be executed are controlled by a user and authorization concept. In WinAdmin, various users can therefore be defined and passwords and privileges assigned. WinAdmin users must not be confused with UTM users who are used for communications with the UTM applications.

As of version 5.1 of WinAdmin, the authorization concept has been significantly extended. It is now possible to assign specific WinAdmin users read and write authorizations for individual UTM applications and you can even assign different privileges at object type level.

27.1 Passwords and overall privileges

You can use the WinAdmin user property dialog to assign a password and overall privileges to users. This is described in detail in section "WinAdmin users" on page 29. You can also assign and withdraw overall privileges in the 'WinAdmin Objects'/WinAdmin Users' subtree of the navigation tree as described in the following section.

If a password has been assigned to a WinAdmin user then WinAdmin can only be started under this user name if the correct password is entered in the sign-on dialog. When defining a new user, a user with the master privilege has the option of directly defining a password for this user. The 'Change Password...' command in the Extras menu can be used at any time to change the password of the current user or to delete it by entering a blank password. A password can also be changed by clicking the 'Change...' button in the sign-on dialog box when WinAdmin is started.

WinAdmin can be minimized and locked by selecting the 'Minimize and Lock' command in the Extras menu or by clicking the appropriate icon in the toolbar. If you subsequently want to reactivate WinAdmin, the password is queried again.

The overall privileges that can be assigned to a WinAdmin user in the corresponding property dialog cover the so-called master privilege as well as the read and write authorizations. "Overall" means that any authorization attributed here applies to all UTM applications and the corresponding objects. If you want to assign different privileges for different UTM applications or object types then you must specify 'No' for the corresponding overall privilege and then assign the various authorizations via the navigation tree as described in the next section.

Only users with the "write privilege" can make administrative modifications to UTM applications via WinAdmin (create, delete, modify objects, set up connections, etc.). Users without this privilege can only view the data and objects of UTM applications. If a user does not have the write privilege, then the write actions are deactivated in the menus and are therefore unavailable.

The following overall privileges are available:

- Read authorization for administration
A user with this privilege may view but not modify the administration data of all UTM applications and of their objects. The Command mode may not be used.
- Write authorization for administration
This privilege includes read authorization for administration. A user with this privilege may view and modify the administration data of all UTM applications and of their objects.

- Master privilege

A user with this privilege has all the privileges mentioned above and can also create, modify or delete hosts, applications, UPIC connections, Collections and WinAdmin users in the configuration database.

27.2 Privileges for individual applications and object types

You assign privileges at application and object type level via the WinAdmin navigation tree. You can navigate from the 'WinAdmin Objects' node to the 'WinAdmin Users' subtree. For each existing WinAdmin user, this subtree displays the user's authorizations for each application and each type of UTM object associated with the application. Provided that you are authorized to do so, you can use the context menus available here to assign and withdraw authorizations.

The subtrees for each WinAdmin user are constructed in the same way:

If a user possesses a privilege then this is displayed in **bold** in the subtree (for example '**Administration Read**'), otherwise the privilege is displayed in brackets using a normal font (for example, Administration Write)').

You can withdraw an existing privilege using the 'Withdraw Privilege' command in the context menu. In contrast, if no privilege has been assigned, the context menu offers the 'Grant Privilege' command.

At the topmost level of each subtree, there are five entries which correspond to the possible overall privileges: 'Master', 'Administration Write' and 'Administration Read'. If a WinAdmin user possesses the master privilege then only the **Master** node is displayed since this implicitly includes all the other privileges. The context menu of this node can be used to withdraw the master privilege from the user. The other four nodes are then displayed.

If a user possesses one of the two overall privileges 'Administration Write' or 'Administration Read', then there is no subordinate subtree below the corresponding node. The node's context menu can be used to withdraw the corresponding overall privilege from the user. If one of these four overall privileges is not assigned, then a subtree below the node permits access to the application level authorizations.

Here you can find all the UTM applications known to the configuration database. You can use the context menu to grant or withdraw each privilege separately. If the WinAdmin user does not possess the appropriate privilege for an application then the individual object types for this application are displayed one level lower in the hierarchy. Here you can grant or withdraw authorizations specifically for certain object types relating to the application.

The authorizations of a user who is currently working with WinAdmin are evaluated at a number of different points and affect the way WinAdmin behaves. For example, a user who does not possess any administration rights for an application will not see the application in the administration part of the navigation tree. In the same way, for each application in the navigation tree only those object types are displayed for which the current user possesses at least a read authorization. In dialogs and object lists it is only possible to access the property dialogs of related objects (via buttons in the dialog boxes or by choosing 'properties' in the context menu of a list box) if the user possesses the corresponding authorizations.

28 File transfer functions in WinAdmin

To perform certain operations, WinAdmin requires file transfer functionality. These operations are as follows:

- Starting UTM applications
(file transfer with follow-up processing)
- Starting KDCUPD runs
(file transfer with follow-up processing)
- Getting KDCUPD reports
- Getting files (general)
- Sending files (general)

The use of the file transfer product openFT (as of version 8.1) is not always a precondition. If openFT follow-up processing is not required for a function, for example in the case of a simple file transfer, then it is also possible to use FTP if the application computer is an FTP server.

You must make a number of specifications in WinAdmin before it is possible to use the file transfer functions. You specify the relevant values in the "File Transfer" sheet of the Settings dialog (see section "The 'File Transfer' sheet" on page 166).

When performing file transfers, WinAdmin first checks whether openFT is available on the computer. If it is, openFT is used for the file transfer operation. If it is not, WinAdmin uses FTP. If the file transfer requires follow-up processing, WinAdmin rejects the job in this case.

All the actions listed above can be only be performed without restriction if openFT is available both on the administration PC and on the server. In addition, openFT must be configured on both sides of the connection in such a way that the partner machine is known on the opposite side. For further details, please refer to your openFT documentation.

WinAdmin relies to a great extent on openFT's capabilities for follow-up processing. When an application is started, for instance, a dummy file is transferred to the server and, following the transfer, the start procedure is initiated. For this to be possible, openFT must allow follow-up processing on all servers. The option 'Inbound follow-up processing allowed' has to be activated in the authorization record for this.

The dummy files cannot be deleted on servers with Unix, Linux or Windows operating systems. WinAdmin generates the following files in the work directories listed below:

- Function 'Start Application'
WinAdminStartApplication<n> (Unix/Linux systems) or
WinAdminStartApplication<n>.cmd (Windows systems)
- Function Run KDCDEF'
WinAdminRunKdcdef<n>
- Function Run KDCUDP'
WinAdminRunKdcupd<n>

where <n> is the ID number of the corresponding UTM application.

If openFT is used for file transfer, WinAdmin may address the partner via its ID, password or account number, or by means of the transfer admission of an FTAC profile (File Transfer Access Control). The corresponding dialogs contain a check box labeled 'Use FTAC profile' which allows you to select the required addressing mode. If user ID and password are to be used, the multi-user variant of openFT must be installed on Windows servers, as only the multi-user variant supports access with the user ID and the password.

Compared to access via user ID and password, the use of FTAC profiles provides better protection for the server systems by separating the file transfer admission from the logon/login authorization as well as through the user and partner system-specific assignment of access rights. The flexible grading of the access rights linked to an FTAC profile is also possible.

Before communication with remote partners is possible via openFT or FTP, you must enter the access data. If openFT is used, you may be able to enter an FTAC transfer admission. However, the standard access data comprise a user ID, a password and, in BS2000 systems, an account.

The table below indicates which access data is required for the various communication systems:

Communication system or procedure	Access data
openFT	FTAC transfer admission or user ID, account (in BS2000 system), password
FTP	User ID, account (in BS2000 system), password
Local processing	None

The following operating system-dependent syntax rules and restrictions apply to the access data:

System	Parameter	Description
BS2000	User ID	1 - 8 alphanumeric characters WinAdmin automatically converts these to uppercase. Specified without leading '\$'.
	Account	0 - 8 alphanumeric characters WinAdmin automatically converts these to uppercase. Some user IDs have no account number.
	Password	1 - 8 character long C-string (in the form <pass> or '<pass>' or C'<pass>') or 1 - 16 character long X-string (in the form X'<hexpass>')
	Transfer admission (only possible if openFT is used)	8 - 32 alphanumeric characters (in the form <admission>) or 8 - 32 character long C-string (in the form '<admission>' or C'<admission>' Important: If the transfer admission was not entered as a C-string or X-string in BS2000 then BS2000 automatically converts it to <u>lowercase</u> , e.g. XXX becomes xxx. Similarly, WinAdmin also converts a transfer admission entered in the form <admission> to lowercase.
Unix/ Linux	User ID	1 - 32 characters
	Account	Not used
	Password	Alphanumeric characters (maximum length is system-dependent)
	Transfer admission (only possible if openFT is used)	8 -16 characters
Windows	User ID	1 - 36 characters (in the form <userid> or possibly with leading domain name in the form <domain>/<userid>)
	Account	Not used
	Password	8 - 32 characters
	Transfer admission (only possible if openFT is used)	8 -32 characters

WinAdmin administers four different access data records in the configuration database. The following table describes the four records and the way they are used in greater detail:

Access record	Description	Modify
Start	Used to start UTM applications There is one record for each UTM application.	In the 'Start Application' dialog.
KDCDEF	Used to send KDCDEF inputs, start KDCDEF runs and fetch KDCDEF protocols. There is one record for each UTM application.	In the "Send KDCDEF input" and "Run KDCDEF" dialogs. The "Get KDCDEF protocol" dialog requires an existing record.
KDCUPD	Used to start KDCUPD runs and fetch KDCUPD protocols. There is one record for each UTM application.	In the "Run KDCUPD" dialog. The "Get KDCUPD protocol" dialog requires an existing record.
Get/send files	Used for the (general purpose) retrieval and sending of files. There is one record for each host.	In the "Get file" and "Send file" dialogs.

If the password is changed for the Start, KDCDEF or KDCUPD access records then WinAdmin automatically changes the password of the other two records for the same UTM application and of the record for the corresponding host if the user ID is identical.

The following table indicates when certain actions are possible together with the communication system (or procedure) via which they are performed.

Remote partner: WinAdmin host and partner host are different

Action	openFT installed on...		Communication system or procedure	Possible	
	WinAdmin host	Partner host			
<ul style="list-style-type: none"> - Send file - Distribute KDCDEF input - Get file - Get KDCDEF report - Get KDCUDP report 	Yes	Yes	openFT	Yes	
		Unknown, no	openFT (attempt)	?	
	Unknown	Yes, unknown	If ncpdll32.dll is found then openFT (attempt), otherwise FTP (attempt)	?	
			No	FTP (attempt)	?
			Yes, unknown, no	FTP (attempt)	?
<ul style="list-style-type: none"> - Start UTM applications - Initiate KDCDEF runs - Initiate KDCUPD runs 	Yes	Yes	openFT	Yes	
		Unknown	openFT (attempt)	?	
		No	-	No	
	Unknown	Yes, unknown	If ncpdll32.dll is found then openFT (attempt), otherwise not possible	?	
			No	-	No
No	Yes, unknown, no	-	No		
<ul style="list-style-type: none"> - Display file lists 	Yes	Yes, unknown	BS2000 partner: openFT	Yes	
			UNIX, Windows partner: FTP (attempt)	?	
		No	BS2000 partner: not possible	No	
			UNIX, Windows partner: FTP (attempt)	?	
	Unknown	Yes, unknown	BS2000 partner: If ncpdll32.dll is found then openFT (attempt), otherwise not possible	?	
			UNIX, Windows partner: FTP (attempt)	?	
		No	BS2000 partner: not possible	No	
			UNIX, Windows partner: FTP (attempt)	?	
	No	Yes, unknown, no	BS2000 partner: not possible	Yes	
			UNIX, Windows partner: FTP (attempt)	?	

Local partner: WinAdmin host and partner host are the same

Action	openFT installed on...		Communication system or procedure	Possible
	WinAdmin host	Partner host		
<ul style="list-style-type: none"> - Start UTM applications - Initiate KDCDEF runs - Initiate KDCUPD runs 	Yes, unknown, no		Local processing (copying of the (dummy) file and command execution via cmd.exe or command.com).	Yes
<ul style="list-style-type: none"> - Display file lists 	Yes, unknown, no		Local processing	Yes

29 Other functions

29.1 Dead Letter Queue Functions

openUTM contains the function "Dead Letter Queue".

Controlled by generation (parameter DEAD-LETTER-Q of the TAC generation statement), the loss of asynchronous messages to a transaction code or to a Tac queue can be prevented by storing the message in error cases in the dead letter queue, a Tac queue with the name KDCDLETQ. The messages can be read from there and/or they can be reassigned to an asynchronous Tac or a Tac queue by administrative means. For the messages of the dead letter queue the name and the type of the original destination will be displayed.

This function will be supported by WinAdmin too.

The following GUI elements contain fields that are related to the dead letter queue functionality:

- Property dialog of an openUTM application
On the 'Max Parameter(4)' page there is the field 'Dead Letter Queue Alarm' (not modifiable).
- Property dialog of a Tac
On the 'Specific' page there is the field 'Save in Dead Letter Queue' (modifiable).
- Tac-list Contains the column 'Save in Dead Letter Queue'.
- Message queue list of the dead letter queue
See section "Message Queue Extensions".

29.1.1 Message Queue Extensions

The dead letter queue is represented in WinAdmin by the message queue of the Tac "KDCDLETQ". The queue can be displayed by choosing the menu item "Show Message Queue" in the context menu of the Tac "KDCDLETQ" in the Tac list (in a cluster, this command exists only in Tac lists that are local to the node but not in Tac lists that apply globally to the cluster).

All messages that are currently in the dead letter queue will be displayed in the list. The following additional commands are available for the dead letter queue messages. These commands can be activated either through the "Queue" menu or through the context menu of the selected messages:

- Move Job
By activating this command, all currently selected messages can be moved to another destination.
- Move All Jobs
By activating this command, **all** messages that are currently in the dead letter queue can be moved to another destination.

In both cases a dialog will be displayed that allows the selection of the new message destination. It is possible to select either the original destination of the message or another destination. Available destinations are all asynchronous Tacs or Tac queues that have either the property CALL=FIRST or CALL=BOTH. The Tacs "KDCMSGTC" and "KDCDLETQ" are not allowed as new destinations.

At general there are two additional columns in the message queue list that describe the original destination of the messages: 'Destination' and 'Destination Type'. These two new message properties will be displayed in the property dialog of the messages too.

29.2 Command mode

WinAdmin offers a simple command mode in order to call Tacs in UTM applications directly. When you activate command mode, you will see a dialog box which allows you to send administration Tacs to a UTM application and displays the responses in a text window. The format of the responses corresponds to the familiar output format of openUTM.

Command mode can be activated in any of the following three ways:

- Click on the 'Command Mode' node in the Application subtree of the navigation window.
- Right click on the application name in the navigation window and select the 'Command Mode' command from the displayed context menu.

Unlike in the previous version of WinAdmin, the commands are processed in the background in this new version, i.e. when an entered command is sent to UTM and WinAdmin is waiting for the response, the user is able to enter a new command. However, a newly entered command is not processed until the processing of the previous command has been concluded.

The command mode windows contains the following fields:

- 'Send Last Output Line' button
This button allows you to request a continuation of long output lists if the output of the application could not be transmitted in one go.
If you click this button, WinAdmin uses the last line in the output text window as a command and sends this to UTM.
- 'Send' button
When you click this button, the content of the 'Command' edit box is sent to UTM. The transferred command and the response from UTM are displayed in the output text window.
The values in brackets to the right of this button indicate the number of commands that are still waiting to be processed, whether a command is currently being processed and the number of fully processed commands that are displayed in the output text window. "(2/1/5)" therefore means that two entered commands are waiting to be processed, one is currently being processed and five fully processed commands are displayed in the output text window.
- 'Convert to uppercase' check box
If you check this box then WinAdmin converts the entered command to uppercase before sending it to UTM. However, if the first letter of the entered command is a '.' (period) then the period is removed from the command and the command is not converted to uppercase.
When you exit command mode, this option is automatically stored as a user-specific entry in the configuration database and is re-used the next time you start command mode.

- 'Automatic KDC prefix' check box

If this box is checked, WinAdmin gives the entered command the prefix 'KDC' (if it is not already present) before sending it to UTM. In the same way as for the 'Convert to uppercase' option, the action is not performed if the first character is a period.

When you exit command mode, this option is automatically stored as a WinAdmin user-specific entry in the configuration database and is re-used the next time you start command mode.

- 'Always clear output' check box

If this box is checked, WinAdmin clears the output text window before any new output.

When you exit command mode, this option is automatically stored as a WinAdmin user-specific entry in the configuration database and is re-used the next time you start command mode.

- 'Command' combo box

You enter the command that is to be sent to UTM in the edit box of this combo box. The command is sent to UTM when you click the 'Send' button or (if the 'Send' button is the default button) when you press the Enter key.

The commands that have been used the most are stored in the combo box's dropdown box and can be selected if required. The commands that are located at the top of the combo box are the ones that have been used most frequently. When you exit WinAdmin, these commands are also stored in WinAdmin user-specific form in the configuration database together with their use counters and are loaded again when you next enter command mode.

The commands that have been send using the 'Send last output line' button will not be considered here.

- '...' button (three dots)


Clicking this button opens a dialog box in which you can delete individual or all commands from the 'Command' combo box.

- 'Clear output' button

A click on this button clears the output list box. You can also execute this command from the Edit menu.

- Output text window

The entered commands and UTM responses are displayed in this text window. This window is usually a specially formatted text window as described in the section "Show File command" on page 145.

The beginnings of the commands are indicated by the  icon at the left-hand edge of the text window. Right-clicking this icon opens a context menu that contains the command 'Delete Command...'¹ which allows you to delete individual commands (including existing output) from the text window.

Note:

Non-administration Tacs can also always be entered in command mode.

However, command mode is not suitable for asynchronous Tacs, for performing multi-step transactions or for displaying formatted program outputs. WinAdmin aborts multi-step transactions after the first program unit run.

29.3 Message Dump

As of version 5.3 openUTM offers a significantly extended message dump interface that is supported by WinAdmin too. To do so, WinAdmin has a dialog sheet in the property dialog of an UTM application or an cluster (but not in the property dialog of a node application) that is used to activate and control the message dump functionality. This dialog sheet can be displayed directly by means of the menu command 'Functions' / 'Message Dump' in the Application menu for administration of a single UTM application.

The interface enables the definition of up to three different events, whose occurrences cause UTM to create a diagnostic dump. The following events can be monitored:

- Occurrence of a certain UTM message
In this case up to three different message inserts can be defined additionally that will be compared with specified values by the message dump function to decide whether a diagnostic dump shall be created or not.
- Occurrence of a certain sign-on return code
- Occurrence of a certain RCCC-code
- Occurrence of a certain RCDC-code

The dialog sheet has the following input fields:

- Combo box in which you can choose between the three different dump results. The dump result selected here is output in the following text boxes.
- 'Event Type'
The type of the event that shall be monitored must be specified here. The following types are available:
 - '<none>'
This value indicates that the dump event is not set or shall be reset (deleted) if it is currently set. In this case it is not possible to specify anything else on that dialog sheet.
 - 'Message'
If this value is specified, a certain UTM message ID must be entered, whose occurrence shall lead to a diagnostic dump.
If this event type is specified, up to three message inserts can be defined in the lower part of the dialog sheet (see below).
 - 'Sign-Code'
If this value is specified, a certain sign-on return code must be entered, whose occurrence shall lead to a diagnostic dump.
 - 'RCCC-Code'
If this value is specified, a certain RCCC-code must be entered, whose occurrence shall lead to a diagnostic dump.
 - 'RCDC-Code'
If this value is specified, a certain RCDC-code must be entered, whose occurrence shall lead to a diagnostic dump.

- 'Event'
This field represents the event itself. According to the specified event type this must be either a certain UTM message ID (e.g. 'K065'), a sign-on return code (e.g. 'U04'), an RCCC-code or an RCDC code.
The input length is limited to four characters.
- 'Insert 1', 'Insert 2' und 'Insert 3'
If the event type 'Message' is selected, it is possible to define up to three message inserts that describe the message whose occurrence shall lead to the creation of a diagnostic dump more accurate. For each defined message insert the following attributes must be specified:
- 'Index'
This is the index of the message insert. The first insert of the message has the index 1. The highest index value that can be selected is 20.
The value '<none>' allows the user to reset (delete) a currently defined message insert. It indicates that this message insert is not defined (not specified). In this case it is not possible to enter a value or to select the comparison for that message insert.
- 'Value'
This is the value of the message insert that shall be monitored.
It is possible either to enter a number, an arbitrary printable value or a hexadecimal printable value in the format X'<hexvalue>' (e.g. X'A05C'). If a number shall be interpreted as a printable value rather than as a binary number it must be entered in the format C'<value>' (e.g. C'1234').
Numbers can be up to 10 digits long, printable values have a maximum length of 32 characters and hexadecimal printable values can have a length of 64 characters (one pair of two characters corresponds to one single byte).
- 'Comparison'
This specifies, whether the entered value must be contained in the message insert ('Equality') or must not be contained in the message insert ('Inequality') to cause a diagnostic dump to be written.

29.4 Actions

WinAdmin allows you to define and call "actions". The properties of defined actions are saved in the configuration database so that it is available to all WinAdmin instances that work with the database.

Actions are used not only as statistic threshold alerts (see section "Threshold alert" on page 118) but also to speed up specific activities (e.g. display of a listing) by defining a corresponding action.

There are the following types of action.

- Save Object Properties in File
See section "Save Object Properties in File" on page 141.
- Call Script
A specified script (a text file with commands, generally on Windows systems with the suffix '.cmd' or '.bat' and on Unix/Linux systems with the suffix '.sh') is called.
- Open File
A specified file is opened (if necessary with the help of the program associated with it).

- Show File
A specified (text) file is displayed.
- Execute Program
A specified program is executed.
- Ringing
A standard call ring is played.
- Play Sound File
A specified Sound file is played.

Administration and working with actions is controlled in the navigation window.

A simple click on the 'Actions' node or the 'Actions' menu command of the 'File' menu displays the 'Actions' list. In addition, the 'Actions' node has a context menu that is used to display the actions and to create new actions.

You work with the actions list in the same way as with other object lists (see also section "The navigation window is located at the left-hand edge of the WinAdmin program window. It allows you to access the majority of the objects and functions provided by WinAdmin.

" on page 42).

The actions themselves have a context menu with the following commands.

Menu command	Description
New Action	Creates a new action. The command opens an intermediate dialog box in which you have to specify the type of action to be created
Delete Action	The selected action is deleted.
Execute	The action is executed.
Test	The action is tested; in other words, an attempt is made to execute the action and the result of the attempt is displayed in text form in a dialog box. Result text is generally output in English.
Properties	Displays the property dialog of the action.

When using actions that relate to files, note that the specified file must be present at the same location on all WinAdmin computers that work with the same configuration database so that the action is executed correctly on all computers.

When you use the 'Execute Program' action type, you can also pass program arguments in the 'File' field that follows the name (path) of the program to be executed. The arguments are passed to the program exactly as specified here.

Note that the program name (path) must be enclosed in double quotes if the name contains blanks.

The property dialogs for all action types contain the following properties:

- 'Name' edit box
User-definable name for the action which must be unique within the configuration database. The name can be up to 128 characters in length.
- 'Action' text box
Specifies the type of action.
- Remark
You can enter a comment of your choice here. The comment can be up to 255 characters in length.
- 'File' edit box
This box is not present for the types "Ringing" and "Save Object Properties in File".
For the other types, the relevant file (e.g. the script or program to be executed) is entered in this field.
Click the button with the three dots at the right if you want to select the file via a file selection dialog.

29.4.1 Save Object Properties in File

An action of the "Save Object List in File" type lets you save the contents of a specific object list in the form of a text file. The format of the text file is such that the data can be easily imported into Microsoft Excel. There is one line in the file for each object in the object list. The individual property values for the objects are separated from one another by a tab character. The first line in the file contains the names of the individual properties and are also separated from one another by tab characters. The created files can of course also be opened using a text editor of your choice.

When these actions are performed, all the objects present in the object list at the time of the save operation are saved and **all** the object properties are included.

One new feature in this version of WinAdmin is the possibility to save the application parameters of UTM applications in a file. Because the application parameters are not recurrent objects of the same type the structure of the generated file is somewhat different. Each line in the file represents an application property and contains three entries separated by tab characters (type of application parameter, name of the displayed property and value of the property). The first line in the file contains the description of these three "table columns".

WinAdmin can also save object properties automatically at scheduled times. Whereas in WinAdmin V6.0, object lists could only be saved once a day at a scheduled time, in the current version of WinAdmin you can set this interval to be either shorter (the smallest interval is one hour) or longer (up to 96 hours).

In addition, and in contrast to the previous version of WinAdmin, actions of this type can be performed even if no collection containing the corresponding UTM application is open. WinAdmin attempts to contact the specified UTM application and retrieve the objects of the specified object type from the UTM application irrespective of whether or not any collection is open or the UTM application is present in the open collection. This is particularly useful in the case of actions which perform automatic save operations.

In addition to the above fields, a property dialog of this type also contains the following fields:

- 'Application' edit box
The openUTM application to which the object list belongs.
- 'Object List' dropdown box
Type of object list to be saved.
- Destination Directory edit box
Specifies the directory in which the target file is to be saved.
You can use the button with the three dots on the right to define the destination directory.
The default directory is "<WinAdmin-directory>\statistics".
- Filename Prefix edit box
Filename prefix of the file to be created.
The following file is created when the action is performed:
"<Target-directory>\<filename-prefix>.<Date>.<Time>.<Action-ID>.txt".
The date format is "YYYYMMDD" and the time format is "HHMM".
<Action-ID> is a number that specifies the internal ID of the action.
Example: " C:\WinAdmin\statistics\WadmUserListe.20110519.1005.23.txt"
- 'Automatical Save' section
In this section, you can specify whether and at what intervals the object properties are to be saved automatically by WinAdmin.
 - 'Interval (h)' dropdown box
Here you specify the interval between saves in hours. If you do not want the object properties to be saved automatically, select "No Automatic Save" here.
 - 'Reference Time ((h)h:mm)' edit box
Reference time for the automatic save in the format (h)h:mm.
Example: If the reference time is 13:15 and the interval between saves is 6 hours then WinAdmin attempts to save the object properties at 13:15, 19:15, 01:15, 7:15, 13:15 etc..
 - 'Last / Next Save' text box
This box indicates whether the object property has been automatically saved and, if it has been, the last time this was done. If the action was performed manually by the user (by means of the "Execute" or "Test" command in the action's context menu) then this time is not indicated here.

In addition, WinAdmin specifies the calculated time of the next automatic save here. This time may also be in the past if save operations have been "missed" (see above) or if the relevant UTM application is not available.

Note that WinAdmin must, of course, be running in order to perform the action at the desired time. The corresponding UTM application must also be available at the desired time.

If WinAdmin is not started at a time when the object properties are due to be saved then WinAdmin attempts to catch up the "missed" save operation the next time it is started.

Example: If the interval between saves is 12 hours and the object properties were last successfully saved at 15:00 then the next save operation should be performed at 03:00. If WinAdmin is not started at this time and is started again the next morning at 07:00 then WinAdmin attempts to make up for the "missed" save operation after start-up and saves the list at, for example, 07:02. The next save operation is then performed at the normal time of 09:00.

Accessing saved files

The files created on the execution of the action are listed below the node corresponding to an action of the type "Save Object Properties in File" in the navigation tree. Here, each file is represented by a separate node. The context menu of this type of node contains two commands which can be used to display the contents of the file:

- 'Show File as List'

Displays the file in the form of a list.

The displayed list is simply a clearer presentation of the file content. It cannot be compared to the lists of UTM objects that are retrieved from the running applications.

In particular, it is not possible to define any column sets or complex display filters. Furthermore, there are no property dialogs for the list entries and linked objects cannot be accessed via the context menu. Icons are also not displayed in the list.

However, text filters and column filters can be used and the list can be sorted as usual.

This list can also be displayed by single-clicking the node corresponding to the file in the navigation tree.

- 'Show File'

Displays the file as a text file. The tab characters present in the file are replaced by spaces.

29.5 Importing the BeanConnect container.properties file

You can use the menu item "Import/BeanConnect container.properties-File..." in the File menu to import the host and the UTM application stored in a BeanConnect container.properties file into the configuration database. After this, the imported UTM application is ready to be entered in an application collection.

The container.properties file describes a BeanConnect proxy and is located in the BeanConnect proxy's "config" subfolder (e.g. <BeanConnect folder>/BCCont/config/container.properties).

The user is first asked to select the container.properties file that is to be imported. WinAdmin then checks that the selected file is a valid container.properties file. If it is, then the stored host and, if it is not yet present in the configuration database, the application are generated. If an application is imported then a standard UPIC connection is also generated for it. Messages relating to the import process are displayed in the WinAdmin protocol window.

Before adding to an application collection, it may be necessary to check the WinAdmin objects (host, UTM application and UPIC connection) that are generated during the import since WinAdmin must often make assumptions regarding certain of the object properties. This is particularly true of the communication user's password.

29.6 Get File command

You can transfer any file from another computer to the administration PC using the 'Get File...' command from the File menu. The file transfer product openFT has to be available on both systems for this function to be executable (see also section "File transfer functions in WinAdmin" on page 129) or FTP communications between the two systems must be possible.

When you choose this command, a dialog box appears in which you make specifications for the file transfer operation:

■ 'Source' section

Here you make specifications concerning the source file.

- 'Host' dropdown box
The computer from which the file is to be retrieved can be selected in this dropdown box. The dropdown box contains a list of all computers known to WinAdmin except for those for which both openFT and FTP have been specified as unavailable.
- 'User ID' combo box
The combo box's dropdown list contains all the user IDs that are known to WinAdmin on the selected source host. You can also specify another user ID of your choice in the combo box's edit box.
- 'Password' and 'Account' edit boxes
Here you specify the password for the user ID (if necessary) and the account number (BS2000 systems only).
- 'FTAC Admission' check box and edit box
If file transfer is to be performed using openFT, you can enter an FTAC profile here. The check box at the beginning of the line must be checked as otherwise the specified profile is not used.
- 'File' edit box
Name (and path) of the file that is to be retrieved.

■ 'Transfer' section

Here you make specifications concerning the transfer of the file.

- 'Transfer Binary' check box
If this box is checked, the file is transferred in binary form. Otherwise it is transferred in ASCII mode (if necessary, an EBCDIC-ASCII conversion is performed automatically).
- 'Transfer With openFT/FTP' radio buttons
- If neither openFT nor FTP are indicated as being unavailable on the selected source host then you can specify whether the file is to be transferred with openFT or FTP here.

■ 'Destination' section

Here you make specifications concerning the destination for the file transfer.

- 'File' edit box

Name and path of the destination file. You can also use the ... button on the right end of the line to select the file via a file selection dialog.

- 'Mode' dropdown box

Here you specify whether an existing destination file is to be overwritten, extended or retained. If the transfer is performed using FTP then it is not possible to select 'Extend existing file'.

■ 'Subsequently show the file' check box

If this box is checked, WinAdmin uses the 'Show File' function to display the file once the file transfer has been successfully completed' (see section "Show File command" on page 145). In the case of a binary file transfer, WinAdmin displays the file in dump mode.

Once the file has been transferred correctly, WinAdmin saves the user ID, password, account and FTAC admission specifications in the configuration database and these are then available the next time 'Get File...' is called and can be selected via the 'User ID' dropdown box.

29.7 Send File command

You can use the 'Send File...' command from the File menu to send a file of your choice from the administration PC to another computer.

When you choose this command, you will see a dialog box which corresponds to the one described in section "Get File command" on page 144. However, in this case the source and destination roles are reversed.

29.8 Show File command

WinAdmin can display the contents of any (text) file in a separate window. To do this, you use the 'Show File...' command from the File menu. When you choose this command, the 'Select Text File' dialog box appears and a file can be selected. Click the 'Open' button in the dialog box to open a window in which the file contents are displayed.

Alternatively, you can also use drag-and-drop to display a text file. To do this, you drag a file, for example from the Windows-Explorer, either to the (empty) WinAdmin work area or, alternatively, to the WinAdmin protocol window.

The WinAdmin text file window offers very extensive functions for searching for and selecting areas of text and can also be used to edit the text.

The functions of the window are described in the following sections in detail.

The text file display window contains the following components:

- The text window itself in the left-hand part of the center area.
At the left-hand border of the text window there is a bar which indicates any markings as well as the line numbers if line numbers are activated.
A bar containing the column numbers may be displayed at the top of the text window if column numbers are activated.
- The summary bar to the right of the text window.
This bar provides an overview of all the marked locations and text hits throughout the file.
- The search bar below the text window.
You can enter the search text in this search bar. It functions in a very similar way to the filter bar that is available in tables (see section "Filter criteria" on page 50).
- The button bar below the search bar.
This bar contains a number of buttons which you can use to launch text file actions. The available buttons are listed in the description of the actions in the sections below

If the text file display window is active then the WinAdmin status bar outputs a tooltip for the displayed file. The following information is output:

- The position of the insertion point in the file (line and column) if the insertion point is currently set
- The size of the file in Kilobytes
- The number of rows
- The number of markers that are currently set
- The number of currently selected rows

You can perform a number of different actions on an opened text file such as searching for and marking particular areas of the text, copying parts of the text to the clipboard etc. Most of the actions can be run using the buttons below the text file display window. The text file also has a context menu which you can use to perform the actions. Keyboard shortcuts are also available for most of the actions.

The terms used in this description are described below:

- **Insertion point position**
The current position of the insertion point is displayed by means of blue lines in the text window. The entire line in which the insertion point is positioned is underscored in blue and a vertical blue line is displayed in front of the character position at which the insertion point is located.
You can left-click in the text window to move the insertion point to the character position on which you have clicked. In the case of a jump to a line with a given number, the insertion point is positioned at the first character in this line.
- **Selection**
Selected text is displayed in the colors set for this in the system.
It is only possible to select a continuous block of text.
This can be copied to the clipboard and changed to a marker. Selected text is displayed as a proposal when you enter search text.

- **Marked text**
Marked text is displayed in white against a blue background in the text window.
Text is marked when you enter a search text. Another way of marking text is to change a selection into a marker.
- **Marked lines**
Marked lines are displayed in black against a light blue background in the text window.
Lines can be marked as search text by entering a complex filter mask.
- **Line range**
Line ranges are identified in the text window by a black bar at the right-hand end.
There can only be one continuous line range at any one time.
Line ranges can be selected, hidden or deleted.

The sections below describe the actions that are available when text files are displayed and how you can trigger these.

29.8.1 Selecting text

You can mark areas of text by clicking in the text and then dragging the mouse (as in most text editors).

To select the entire text, press the keyboard shortcut **Ctrl+A**.

You can also select text starting from the current insertion point position by pressing the arrow keys with the Shift button held down.

You can also select a defined range of lines (see below).

Alternative: Use the 'select' text command (see section "Text commands" on page 154).

29.8.2 Copying text to the clipboard

The selected text is copied to the clipboard when you press Ctrl+C.

Context menu: 'Copy Selection'

Alternative: copy text command, see section "Text commands" on page 154

29.8.3 Highlighting identical words

If you use the mouse to click on a "word" in the text then all occurrences of the same word in the text file are highlighted in bright yellow and all the corresponding locations are marked in the summary bar at the right-hand edge of the text window.


If you move the mouse pointer over one of these markers in the summary bar then the pointer changes into a pointing hand. When you click the left mouse button, the text window moves to the corresponding position in the text and selects it.

To cancel the highlighting of identical words in the text, left-click in a free area in the text window.

The word delimiters that are used to define the start and end of a word below the mouse pointer can be edited in the property dialog for the text file display window (see section "Displaying the property dialog

" on page 154).

29.8.4 Marking text or lines

Clicking the  button or pressing the keyboard shortcut **Alt+F** opens a dialog box in which you can enter a search text. If an item of text is already selected when you do this then the selected text is entered by default as the search text in the dialog box.

In this dialog box, you can also specify how the search text is to be interpreted:

- Simple Search Text

The function searches for an exact match with the entered text. Any locations at which the search text occurs are marked.

In the 'Case Sensitive' checkbox, you can also specify whether or not the search is to take account of the case of the text.

- Regular Expression Search

The entered text is interpreted as a regular expression. Any locations at which this regular expression occurs are marked.

In the 'Case Sensitive' checkbox, you can also specify whether or not the search is to take account of the case of the text.

- Complex Filter Search



The entered text is interpreted as a complex filter mask (see section "Complex filter masks" on page 172). In this case, the lines that correspond to the filter mask are marked in full (line marking).

'Find and Mark Text' context menu.

Alternative: Use the 'mark' text command (see section "Text commands" on page 154).


You can also highlight a selected area of text by using the keyboard shortcut **Ctrl+M**.

29.8.5 Jumping between markers

You can use the  and  buttons or the keyboard shortcuts **Ctrl+<** or **Ctrl+Shift+<** to jump from one marker (text or line marker) to the next or to the previous marker. This operation selects the target marked text. The jump always starts at the current insertion point position.

Context menu: 'Jump to → Previous Marker' and 'Jump to → Next Marker'

29.8.6 Jumping to a specific line

You can use the  button or the keyboard shortcut **Ctrl+L** to jump to a specific line number. You enter the required line number in an intermediate dialog box.

If a line with the specified number exists in the text then the line is displayed and highlighted. If no such line number exists, a message is output to inform you of this.


Context menu: 'Jump to → Line'.

29.8.7 Deleting individual markers, the selection or the defined line range

The context menu also allows you to delete specific markers, the current selection or a line range that you have defined.

You do this using the commands in the 'Remove' submenu: 'Line Range', 'Selection', 'Markers' and 'Line Markers'. The 'Markers' and 'Line Markers' commands contain further menu items which allow you to remove specific markers.

29.8.8 Deleting all markers and the selection

You can use the  button or the keyboard shortcut **Ctrl+Del** to remove all the existing markers (text and line markers), a selection or any line range you have defined.

Context menu: 'Remove → All'.

29.8.9 Fast searches for text locations

Below the text window there is usually a search bar which provides you with a fast, easy way to search for specific text areas. The entire text is searched through line-by-line for the entered search text and using the defined options and the identified locations in the text are highlighted and marked in the summary bar at the right-hand edge of the text file. If you modify the search text or options, the displayed hits change immediately to match the new settings. This constitutes a quick way of gaining an overview of certain text items in a text file.

Unlike marked areas of text which remain marked until you explicitly remove the marking, the hits are only temporarily identified. The temporary markings change again as soon as the search criteria is altered and are all removed when the search text is deleted.

If you move the mouse pointer over a corresponding marker in the summary bar then the pointer changes into a pointing hand. When you click the left mouse button, the text window moves to the corresponding position in the text and selects it.

The search bar is very similar to the filter bar that is displayed for tables in terms of both its layout and operation (see section "Layout of the filter bar" on page 51).

Search criteria

The search is primarily determined by the entered search text.

You can also specify whether or not the search is to be case-sensitive.

You can also define how the entered search text is to be interpreted. You can choose between the following search types:

- Simple Text
A simple search is performed for the entered search text.
- Regular Expression
The search text is interpreted as a regular expression.
- Complex Filter
The search text is interpreted as a complex filter mask (see section "Complex filter masks" on page 172).
In this case, the case-sensitivity setting is ignored since this option has to be specified in the search text.

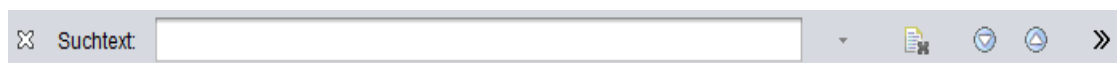
Searches are always performed line-by-line, i.e. areas of text that extend over two or more lines are not found.

By default, only the text box for the search text is displayed in the search bar. In this case, a simple text search is performed and the search is not case-sensitive.

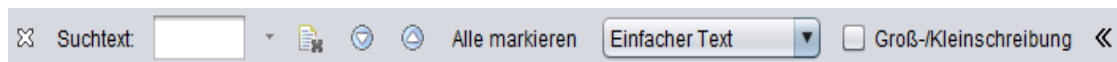
The controls allowing you to select the search type and specify whether a case-sensitive search is to be performed are only displayed if you request them in order to ensure that the appearance of the search bar is not too complex when these items are not required.

Layout of the search bar

As mentioned above, there are two different layouts for the search bar. In the simple layout, it has the following appearance:




You can use the arrow button » at the right-hand end to display the extended search bar:






If the extended layout is displayed, you can click the arrow-button « at the right-hand end to return to the simple search bar layout.

You can hide the search bar by clicking the cross ✖ at the left-hand end of the bar. If the text file display window has the input focus then the search bar can be displayed again using the keyboard shortcut **Ctrl+F**.

You enter the search text in the text box labeled 'Search Text'.

Immediately to the right of this text box there is a small downward-pointing arrow . If you click this then a context menu may be displayed in certain situations (see → "Context menu").

Clicking the  button next to the arrow clears the text box. Clicking this button removes all the highlighting of previous hits in the text and the corresponding markers in the summary bar.

The two buttons  and  can be used to move to the next or previous hit. Instead of using these two buttons, you can also press the keyboard shortcuts **Alt+Plus** and **Alt+Minus** to move to the next or previous hit. The jump to the new position always starts at the current insertion point position.

You use the 'Mark All' button to change all the currently highlighted hits into markers (see section "Marking text or lines" on page 148).

You can use the dropdown box to select the search type (see the section "Search criteria" on page 150).

In the 'Case Sensitive' checkbox, you can specify whether or not the search is to be case-sensitive (see the section "Search criteria

" on page 150).

Search operation

If you enter any search text in the corresponding text box then the text file is searched through again immediately, i.e. after every change to the search text, and the hits are updated in the text file and the summary bar. During the search operation, the background of the text box is displayed bright yellow to indicate that the search operation has not yet terminated. However, this is usually only really visible in large text files.

If no item in the text matches the current search criteria then the background of the text box is displayed bright red.

Context menu

The text box for the search text possesses a context menu that is also displayed when you click the small arrow to the right of the text box.

You can use the commands in the context menu to save the current search criteria. You can also reload search criteria that you have stored in the past. The saved search criteria include the search text, the search type and the case-sensitivity setting.

There is also a command that allows you to delete the entered search text.

The context menu is displayed either if the text box for the search text is not empty or if stored search criteria already exist.

29.8.10 Moving to the start/end of the text

You can use the  button or the keyboard shortcut **Ctrl+Home** to move to the start of the text.

The keyboard shortcut **Ctrl+End** takes you to the end of the text.

29.8.11 Defining line ranges

To define a line range, specify the first and last lines in the range.


You can press the keyboard shortcut **Ctrl+R** to make the highlighted line (the line containing the insertion point) the starting line for the range.

Context menu: 'Line Range → Set End'.

You can press the keyboard shortcut **Ctrl+Shift+R** to make the highlighted line (the line containing the insertion point) the last line for the range.


Context menu: 'Line Range → Set Start'.


Alternative: Use the 'range' text command (see section "Text commands" on page 154).

The start of a defined line range is indicated by the  icon in the bar at the left-hand edge of the text file window. If you move the mouse pointer over this icon then a vertical line that reaches to the end of the line range is displayed below this icon.

29.8.12 Hiding a line range


You can hide a defined line range. This can be useful, for example, if you only want to display certain sections of large documents in order to improve clarity.

To hide a line range, click the  icon in the bar at the left-hand edge of the text file window. The icon has a context menu which contains the 'Hide Line Range' which you can also use to do this.

The  icon is displayed to indicate that there is a hidden line range at the corresponding position and "--== Hidden line range ==--" is displayed in the text line next to the icon.


Alternative: Use the 'hide' text command (see section "Text commands" on page 154).

29.8.13 Showing a hidden line range again


You can show a hidden line range again by clicking the placeholder icon  in the bar at the left-hand edge of the text file window. Alternatively, you can choose the menu item 'Show Hidden Line Range' in the icon's context menu.

Alternative: Use the 'show' text command (see section "Text commands" on page 154).

29.8.14 Selecting a line range

You can select a defined line range by choosing the 'Select Line Range' command in the context menu of the  icon in the bar at the left-hand edge of the text file window.

29.8.15 Selecting the character font



On the 'Font' sheet of a text window's property dialog ( button below the text window itself), you can define the font that is to be used to display the text file.

The text window is intended to be used with monospaced fonts ('Monospaced Fonts'). If you select a proportional font for the display then it is output as if it had a fixed character width. In this case, you can use the 'Character Width' slider in the dialog box to specify the spacing between the individual characters in the proportional font.

The 'Font Size' control allows you to specify the size of the selected font.

The 'Preview' window displays a preview of the selected options.

29.8.16 Increasing and decreasing the font size

You can use the  and  buttons or the keyboard shortcuts **Ctrl+I** and **Ctrl+D** to increase or decrease the font size.

Context menu: 'Font Size → Increase' and 'Font Size → Decrease'

Alternative: Use the 'font' text command (see section "Text commands" on page 154).

29.8.17 Online help

Click the 'Help' button or press the **F1** key to display the online help for the text file display window.

29.8.18 Switching line wrapping on/off

You can use the  button or the keyboard shortcut **Ctrl+W** to switch automatic line wrapping on/off.

If automatic line wrapping is on then overlong lines are always wrapped in such a way that they can be seen in full on the screen. In such cases, the available space is always fully used, i.e. wrapping is also performed within words.

If automatic line wrapping is switched off then a horizontal scroll bar is displayed in the case of overlong lines.

Context menu: Switch Line Wrapping On/Off

Alternative: Use the 'wrap' text command (see section "Text commands" on page 154).

29.8.19 Showing line and column numbers


You can use the **#** button or keyboard shortcut **Ctrl+#** to switch the display of line and column numbers. Any markers present are left intact when you switch the display.

If automatic line wrapping is switched on then only line numbers but no column numbers are displayed.

Context menu: 'Switch Line Numbers On/Off'.

Alternative: Use the 'numbers' text command (see section "Text commands" on page 154).

29.8.20 Displaying the property dialog

The text file display window has a separate property dialog which is opened when you click the  button in the button bar below the text window.

The dialog possesses a 'General' sheet in which you can switch the display of line numbers and automatic line wrapping on and off.

In the 'Word Delimiters' dialog sheet, you can edit the word delimiters that are used when identical words are highlighted (see section "Highlighting identical words

" on page 147).

The 'Keyboard Shortcuts' dialog sheet lists all the keyboard shortcuts available in the text file display window and their meanings. This dialog sheet simply lists the keyboard shortcuts. You cannot edit them here.

29.8.21 Text commands

Pressing the keyboard shortcut **Ctrl+Enter** allows you to enter a text command in an intermediate dialog box.

You can enter the following text commands after pressing the keyboard shortcut **Ctrl+Enter**:

- **range**
Specifies a line range for subsequent actions.
In this case, you specify the start and end line numbers as parameters.
Examples:
range The line range comprises the entire text.
range 13 The line range comprises only line number 13.
range -13 The line range comprises the lines 1 to 13 (inclusive).
range 13- The line range comprises all the lines as of line 13 (inclusive).
range 13-25 The line range comprises the lines 13 to 25 (inclusive).
- **select**
Select an area of text.

In this case, you specify the start and end line numbers as parameters.
For examples, see the "range" command.

- **copy**
Copies the current selection to the clipboard.
- **mark**
Marks all the positions containing a specified search text.
- **font**
Increases or decreases the font
font + Increases the font size.
font - Decreases the font size.
- **fontsel**
Switches the possibility of selecting a character font (see section "Selecting the character font" on page 153) on and off.
fontsel on Switches the selection option on.
fontsel off Switches the selection option off.
- **wrap**
Switches automatic line wrapping on and off.
wrap on Switches automatic line wrapping on.
wrap off Switches automatic line wrapping off.
- **numbers**
Switches line and column numbers on and off.
numbers on Switches numbers on.
numbers off Switches numbers off.
- **occurrences**
Switches the automatic highlighting of identical words (see → "Highlighting identical words") on and off.
occurrences on Switches highlighting on.
occurrences off Switches highlighting off.
- **hide**
Hides an area of text.
In this case, you specify the start and end line numbers as parameters.
For examples, see the "select" command.
The **show** command (without parameters) shows all the hidden lines again (but not the deleted lines).
- **show**
Shows a hidden area of text again.
In this case, you specify the start and end line numbers as parameters.
For examples, see the "select" command.

29.9 Copying UTM objects to the clipboard

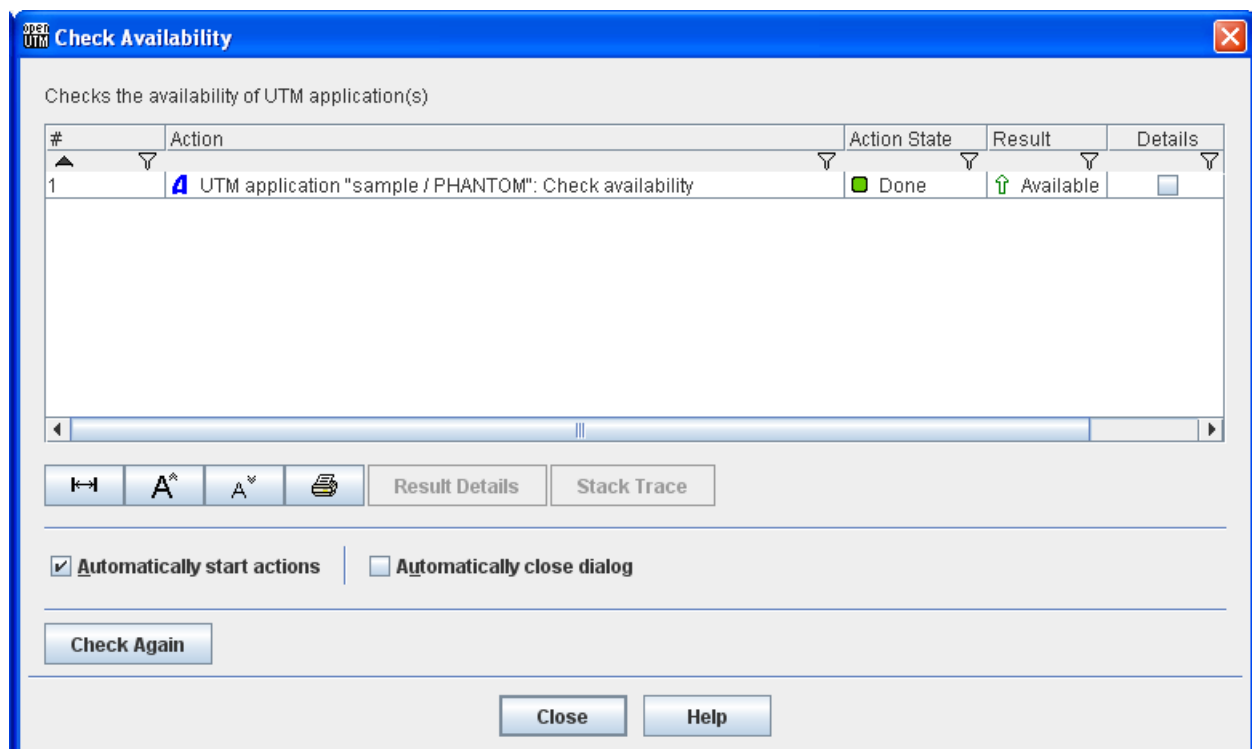
You can copy UTM objects to the clipboard using the key combination Ctrl+C. You can then paste them not only into the object selection but also into other Windows applications (e.g. Microsoft Word or Excel). You can thus edit entire object lists or parts of them using Microsoft Excel, for example. Two different formats are available when copying to the clipboard: the list-oriented and object-oriented formats. The first is particularly suitable for copying multiple objects in list form. The second is more suitable for copying individual objects, for example into a text document. Both methods can be set by selecting 'Settings...' from the Extras menu (see section "The 'General' sheet" on page 164).

29.10 The actions dialogs

WinAdmin uses a so-called "actions dialog" to handle certain actions that may also consist of a number of different subactions. One example of this type of action is a check of the availability of UTM applications.

The actions dialog primarily consists of a table which lists all the subactions. This provides you with an overview of the actions and allows you to identify, for example, which actions have already been executed and the result of their execution. In some cases, it is also possible to display details relating to the result which simplify diagnostic activities, in particular when errors occur. You can also abort the entire action.

Example for an actions dialog:



The table contains the following columns:

■ #

This column contains the action numbers. These numbers specify the sequence in which the subactions are performed.

■ Action

Name of the subaction.

■ Action State

State of the subactions.

This column contains an icon and a text. The following icons are used to identify the state:



The action is still waiting to be performed. In this case, the text 'Waiting' is displayed next to the icon.



The action is currently being performed. In this case, the text 'Running...' is usually displayed next to the icon. However, some actions output specific texts which describe the current status during execution in more detail.



The action has been performed successfully. In this case, the text 'Done.' is output next to the icon.



The action has been performed but could not be completed successfully. In this case, the text 'Done (not OK)' is output next to the icon.

In most cases, further details on the result of the action are available via 'Result Details' (see below).



Indicates that the action was aborted by the user during processing. In this case, the text 'Aborted' is output next to the icon.

Some actions can still be aborted by the user while they are being processed. Most of them, however, are completed once they have been started.



The user has aborted the overall action with the result that subactions labeled with this icon are not performed. The text 'Aborted' is output next to the icon.

■ Result

If the action has been completed then the result is displayed in this column.

If the action was completed without errors then "OK" is usually displayed here.

In the case of an action that has executed an ADMI interface function that has supplied a return code other than "OK" then the main and sub-return codes from the ADMI interface are displayed here.

In this case, KC_MC_xxx indicates the main return code and KC_SC_yyy the sub-return code from the ADMI interface. For a description of the possible return codes, please refer to the UTM manual "Administering Applications".

■ Details

If result details are present for a completed action then the box in this column is checked.

You can then display the result details in the following ways:

- By double-clicking the corresponding subaction in the list.
- By marking the corresponding subaction in the list and clicking the "Result Details" button below the list.
- By choosing the "Show Result Details" command in the context menu of the corresponding subaction.

If result details are present then they are also displayed in the form of tooltips if the mouse pointer is left for a certain time over the corresponding row in the table.

■ Stack Trace

If there is a stack trace for a completed action (if an exception has occurred during the processing of the exception) then the box in this column is checked.

Stack traces that contain the call hierarchy at the time the exception occurred may help you diagnose recurring problems.

You can then display the stack trace in the following ways:

- By marking the corresponding subaction in the list and clicking the "Stack Trace" button below the list.
- By choosing the "Show Stack Trace" command in the context menu of the corresponding subaction.

If stack traces are present then they are also displayed in the form of tooltips if the mouse pointer is left for a certain time over the corresponding row in the table.

Below the action list, you will find the familiar list buttons for setting the column width, increasing/decreasing the font size and printing the list.

You will also find the "Result Details" button here. This button is active if a subaction for which result details are available is marked in the list (see above). There is also a "Stack Trace" button which you can use to display a stack trace if one is present (see above).

Some action dialogs have separate, dialog-specific buttons which are also displayed below the list. In this case, this is the button "Check Again".

WinAdmin usually starts the actions automatically immediately after the dialog box has been displayed.

There are two checkboxes in the lower part of the actions dialog:

■ 'Automatically start actions' checkbox

If this box is checked then the actions in the corresponding dialog box are performed automatically the next time it is opened.

If the box is not checked then you must start the actions by clicking the "Start Actions" button. In this case there is a "Cancel" button next to the "Start Actions" button which you can click to close the dialog box before the actions are started.

- "Automatically close dialog" checkbox
If this box is checked then the dialog box is automatically closed when all the subactions have been successfully completed (all the subactions have a green icon in the "Action State" column). However, the dialog box remains open for at least 3 seconds even if the actions are completed more quickly than this in order to allow the user to check this box.

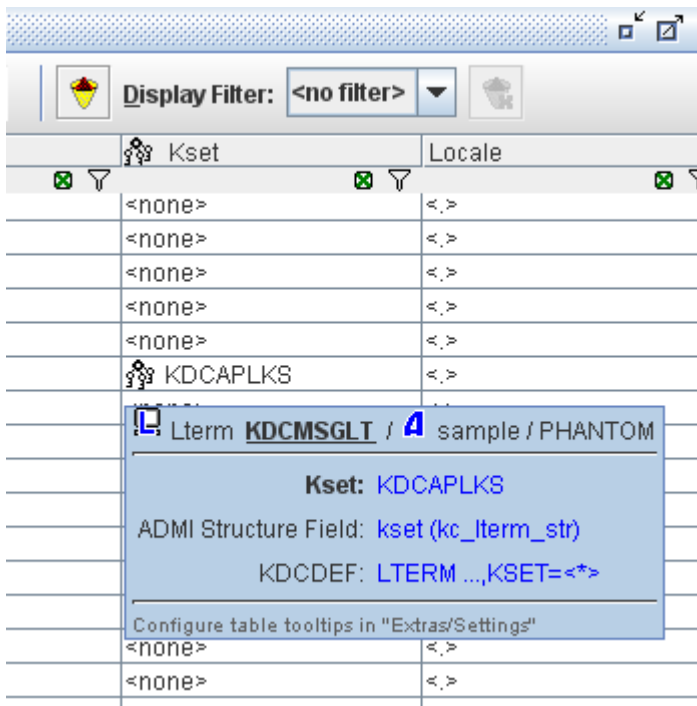
WinAdmin remembers whether or not these boxes are checked for each type of action dialog so that the checkboxes are displayed with their last defined status the next time an action dialog of the same type is opened.

29.11 Tooltips in lists and property dialogs

UTM object lists and the property dialogs of UTM objects display tooltips which provide more detailed information about the displayed object properties. This is particularly useful in object lists and property dialogs which apply globally in the cluster since the tooltips also output the property values for the individual node applications.

Since in global lists and property dialogs that apply globally in the cluster, WinAdmin combines the property values of the individual activated node applications to form a single value, some properties may have the state "<mixed>". The tooltips for this type of property value also display the individual values for the node applications. You can therefore immediately see the individual values that go to creating the value "<mixed>". This also makes it easier to interpret property values that are formed by totaling or averaging combined values. Alongside the individual values for the node applications, the "Combination Type" is also output. This indicates how WinAdmin has combined the individual values.

Below is an example of a tooltip for a table cell in an Lterm list:



Explanation:

- Title

The title of the tooltip indicates the type and name of the corresponding object (including the icon used for the object) and the name of the associated cluster (in this example) or UTM application to which the object belongs.

- Name and value of the property

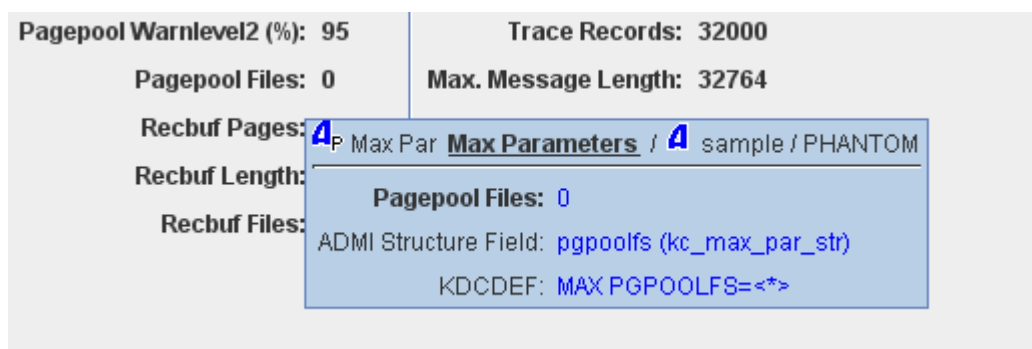
The name of the property displayed in the tooltip is shown in bold (in this example "Connection State") accompanied by the associated value (in this case "<mixed>").

- Combination Type and Node Values

The combination type "Comparison" indicates that the value "<mixed>" was determined by comparing the underlying property values of the node applications. If the values are not the same (as in this case) then the individual node values are output. The host name of the corresponding cluster node is displayed in brackets next to the node values.

The individual node values are displayed if the combination type is not "Comparison" or if the combined value is "<mixed>".

Below is an example for the combination type "Sum" from the property dialog of a UTM cluster application:



- ADMI Structure Field

If the property is read from a data structure in the KDCADMI UTM administration interface (this is the case for most properties), then the name of the field in the structure and, in brackets, the structure name are specified next to "ADMI Structure Field". These specifications make it easy for you to find the exact meaning of the values in the UTM administration guide.

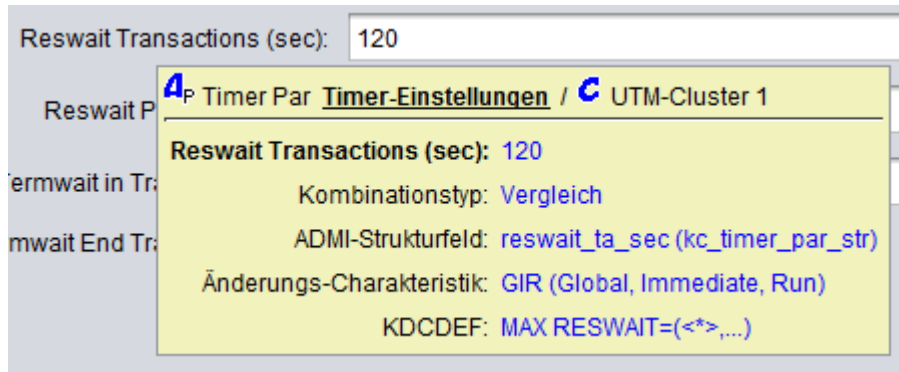
- Modification Characteristics

If it is possible to modify the object property via the UTM administration interface then information on how and when the change takes effect in UTM is indicated next to "Modification Characteristics". The meanings of the specified modification characteristics can also be found in the UTM administration guide.

- KDCDEF

If the object property refers to a KDCDEF generation statement then the name of the generation statement and, if possible, the name of the relevant parameter are specified next to "KDCDEF".

Below is an example of this with a tooltip from the property dialog of a UTM cluster application:



Note: In this example, although the combination type is again "Comparison", the value is not "<mixed>". The individual values of the node applications are therefore all identical and are consequently not output separately.

You can define how and when tooltips are displayed in UTM object lists on the General sheet of the WinAdmin settings dialog ("Tooltips in Tables") (see section "The 'General' sheet" on page 164).

In general, tooltips in the property dialogs of UTM objects are structured exactly as described above. They are described when the mouse pointer is left for a short time over the name of the relevant property (or the button with the property name in the case of referenced objects).

29.12 Refreshing data

WinAdmin works with local data. Whenever necessary, the requested data is polled once from the relevant UTM applications and is kept in the store of the WinAdmin computer. During further processing, WinAdmin works with this local data. In general, the data is only retrieved again when requested by the user, i.e. when a data update (refresh) is initiated.

The data refresh action can be initiated in the property dialogs by clicking on the "Refresh" button. In object lists, simply press the F5 key or select 'Refresh' below the list.

If **all** the objects and application parameters for a UTM application are to be refreshed, you can use the 'Refresh Data' command of a UTM application's Application menu.

29.13 Trace function

For diagnostic purposes, WinAdmin allows you to record information on the program run in files. The scope of this trace information can be regulated in three stages.

The trace function is configured using the Logging/Traces' sheet in the Settings dialog, see section "The 'Logging / Traces' sheet" on page 166.

WinAdmin uses the logging functions provided by the Java Logging Framework Log4j. The configuration settings for the Log4j framework can be found in the file "log4j.properties.xml" in the "config" directory located below the WinAdmin installation directory. It is not usually necessary to edit this file directly in order to modify the settings. Instead, WinAdmin modifies them on the basis of the entries in the above-mentioned dialog sheet of the settings dialog.

To save trace records, WinAdmin uses a mechanism that generates several different logging files. It is necessary to specify the main logging file, the number of backup files and the maximum size of a logging file.

Operation:

The mechanism creates two administration files **<File>.lck** and **<File>.gen**, which are named **logs/logging.txt.lck** and **logs/logging.txt.gen** in most installations, that it needs for operation.

It always writes current information to the logging file **<File>**. At changeover time, the file **<File>** is copied to form a backup file with the name **<FilePrefix>.<next-file-generation>.<FileSuffix>**, and the file **<File>** is written from the start again. If, after this operation, the number of backup files exceeds the value specified in **<MaxNbrBackupFiles>** and if **<MaxNbrBackupFiles>** is greater than 0 then the oldest files in excess of the specified number are deleted on changeover.

Example for <MaxNbrBackupFiles>=3

Situation before changeover:

logging.txt	(in use)
logging.13.txt	(oldest backup file)
logging.14.txt	
logging.15.txt	(most recent backup file)

On changeover, **logging.txt** is copied to **logging.16.txt**, **logging.13.txt** is deleted and **logging.txt** is written again.

Situation after changeover:

logging.txt	(in use)
logging.14.txt	(oldest backup file)
logging.15.txt	
logging.16.txt	(most recent backup file, previously logging.txt)

In this case, the **<FilePrefix>** is "logging" and the **<FileSuffix>** is "txt". A file suffix always consists of exactly three characters and is always separated by a dot from the file prefix.

29.14 Specifying the WinAdmin user at the start

As of Version 6.0, it is possible to pass the WinAdmin user to be signed on and the corresponding password to WinAdmin on start-up. In this case, the dialog box for user selection and password entry is not displayed on start-up, thus making it possible to start WinAdmin without any manual intervention by the user.

This may be of use, for example, if WinAdmin is simply to be used for the automatic collection of statistics values and an automatic start-up is therefore desirable when the WinAdmin computer is powered up.

The WinAdmin start procedure, which is located in the "shsc" subdirectory of the WinAdmin installation directory, has been written to cope with this requirement. The start procedure on Windows systems is named "startwinadmin.cmd" and on Unix/Linux systems "startwinadmin.sh". These files

can be opened and edited with a text editor (in Windows, for example, with Notepad, and on Unix/Linux systems with vi).

The files contain the following lines on Windows systems:

```
rem -- the auto-sign WinAdmin user and the corresponding password (if any)
set AUTOSIGNUSER=
set AUTOSIGNUSERPW=
```

and on Unix/Linux systems the lines:

```
# the auto-sign WinAdmin user and the corresponding password (if any)
AUTOSIGNUSER=
AUTOSIGNUSERPW=
```

To activate the auto-sign mechanism, simply enter the name of the WinAdmin user who is to be signed on after `AUTOSIGNUSER=` and the corresponding password after `AUTOSIGNUSERPW=`.

WinAdmin evaluates these specifications on start-up. If the specified WinAdmin user is known in the configuration database and if the specified password is valid then the user is automatically signed-on to WinAdmin. Otherwise, WinAdmin outputs a message informing you of this in the protocol window. In this case, WinAdmin cannot be used and can simply be shut down via the File menu.

29.14.1 Creating a WinAdmin program link (Windows systems)

To start WinAdmin automatically on Windows systems when the Windows user logs in, you can create a WinAdmin program link and copy this to the "Autostart" program group. You can do this (for example) as follows:

First of all, open Windows Explorer, navigate to the WinAdmin installation folder and display this in the right-hand part of the Explorer window. Now double-click the "shsc" subfolder so that its content is displayed on the right of the Explorer. Search for and right-click on the file "startwinadmin.cmd". Choose "Create Link" from the context menu. The Windows system now creates a WinAdmin program link in the same folder.

In order to modify the name of the program link, right-click the generated link and choose "Properties" from the context menu. You can edit the name of the shortcut in the "General" sheet of the property dialog that is now displayed.

In the dialog's "Link" sheet, you can also modify the icon displayed for the program link by means of the "Change Icon" button. When you click this button, you are first informed that the file "startwinadmin.cmd" does not have an icon. Exit this message window with "OK" and then click "Browse...". You can now navigate to the "shsc" subdirectory of the WinAdmin installation directory in the Explorer window that is displayed. Here you will find the file "winadmin.ico" which contains the WinAdmin icon which you can select for use.

You now close the properties sheet with "OK".

You can now copy the created program link to a folder of your choice (e.g. with Ctrl+C (Copy) and Ctrl+V (Paste) to the target folder). This may be, for example, the "Autostart" folder or the desktop. Double-clicking the link then starts WinAdmin.

To copy the link to the Windows "Autostart" folder, open this folder via the Windows start icon in the task bar. To do this, right-click on the start icon and choose "Open" or "Open All Users" from the context menu. Next, double-click the "Programs" folder in the Explorer window that is now displayed. This folder usually contains the "Autostart" folder which can in turn be opened by a double-click. If the "Programs" folder does not yet contain an "Autostart" folder then you can create it by right-clicking in a free space in the Explorer window, selecting "New"/"Folder" in the context menu and renaming the newly created folder to "Autostart".

30 Settings

The 'Settings...' command in the Extras menu offers various options for defining the behavior of WinAdmin.

Choosing this command opens a property dialog comprising several sheets which allows you to define the settings and actions described below.

30.1 The 'General' sheet

This sheet of the property dialog covers settings of a general nature.

- 'Language' dropdown box
Here you can specify the language to be used in the WinAdmin user interface. You are able to choose between German and English. Any change to the user interface language takes effect the next time WinAdmin is restarted.
- 'Look&Feel' dropdown box
Here you can modify the Java Look&Feel of the user interface.
The entries 'Cross Platform L&F' (Java standard), 'System L&F' (system-specific) and 'Nimbus' are always available.
There is also the entry 'Current L&F'. If this option is selected then the Look&Feel set on WinAdmin start-up is used.
In addition, there is an entry for all the other Look&Feel possibilities available on the platform.
Any change you make here applies only to any new interface items that are generated. It does not take full effect until you restart WinAdmin.
You can click the 'Other...' button to specify other Java Look&Feel classes. These must be present in the WinAdmin start path.
- 'Clipboard Textformat' dropdown box
Object data can be copied to the clipboard either in a list-oriented format or in a format suitable for representing a single object.
Default: list-oriented
- 'Dialog Type' dropdown box
This allows you to choose whether only modal dialogs ('Use Modal Dialogs') or (where possible) "panel" dialogs ('Use Panel Dialogs If Possible') should be used when displaying property dialogs for objects.
Unlike modal dialogs, multiple "panel" dialogs can be open at the same time. These are listed and are available for selection in the Window menu and below the 'Open Windows' node in the navigation tree.
Default: 'Use Modal Dialogs'
- 'Layout of the Application Subtree' dropdown box
Here you can define the layout of the Application subtrees.
See section "Layout of the Application and Cluster subtrees" on page 45.

-
- 'Layout of the Cluster Subtree' dropdown box
Here you can define the layout of the Cluster subtrees.
See section "Layout of the Application and Cluster subtrees" on page 45.
 - 'Tooltips in Tables' dropdown box
The object lists for UTM objects provide tooltips which supply you with more detailed information about the content of the table cell over which the mouse pointer is currently positioned (see section "Tooltips in lists and property dialogs" on page 159).
Here you can specify when the tooltips are to be displayed:
 - Never
Tooltips are never displayed.
 - When Alt-key is pressed
Tooltips are displayed if the mouse pointer is located over an object list which currently has the input focus and the **Alt** key is pressed. In this case, the tooltips are hidden when the **Alt** key is released.
 - Standard.
Tooltips are handled in the normal way, i.e. they appear if the mouse pointer is left for a certain time in the same position over an object list which has the input focus and are then hidden again automatically after a certain time.
Default: 'Standard'
 - '**Max. Stored Communication Requests**' dropdown box
Here you can specify the number of terminated communication requests that are stored in WinAdmin.
See section "Communication requests" on page 25.
Default: 20
 - '**Default UPIC Connection Timeout (sec)**' dropdown box
This dropdown box can be used to set the maximum time that WinAdmin waits for a response in the case of connections to a UTM application. If the set interval is exceeded, the operation is terminated with an error return code and WinAdmin marks the relevant application as 'not available'.
The value set here is the WinAdmin default value. It is also possible to define specific values for individual UPIC connections. These connection-specific values take precedence over the default value.
Default: 60 seconds
 - 'Allow UPIC Data Compression' check box
Here you can specify whether or not the data sent from the UTM application to WinAdmin may be compressed.
While data compression reduces the transferred data volume (reduced network load), it is of course necessary for the UTM application to perform data compression, thus resulting in a higher CPU load at the application server (increased server load).
Default: activated
-

30.2 The 'File Transfer' sheet

This sheet of the property dialog contains settings that apply to the file transfer functions in general and, more specifically, to the openFT configuration.

If openFT is installed on the WinAdmin host then you need to make certain settings in WinAdmin if WinAdmin is to be able to use openFT on this computer.

WinAdmin primarily uses the openFT Java interface. To make this possible, both JAR libraries – "openFTapi.jar" and "openFT.jar" – must be entered in the WinAdmin class path. You can do this easily in this dialog sheet. The changes then take effect the next time WinAdmin is restarted.

The dialog sheet has the following boxes:

- 'Default Directory for File Transfer' edit box
Here you can specify a directory that is to be used as the default directory when retrieving files. You can click the button with the three dots at the right if you want to choose the directory via a selection dialog.
- 'ftexec Directory' edit box
You specify the directory containing the openFT file "ftexec.exe" (Windows systems) or "ftexec" (Unix/Linux systems) here. You can click the button with the three dots at the right if you want to choose the file via a file selection dialog.
The file is usually present in the "bin" subdirectory of the openFT installation directory.
- 'openFTapi.jar Path' edit box
You enter the complete path to the openFT file "openFTapi.jar". You can click the button with the three dots at the right if you want to choose the file via a file selection dialog.
The file is usually present in the "java" subdirectory of the openFT installation directory.
- 'openFT.jar Path' edit box
You enter the complete path to the openFT file "openFT.jar". You can click the button with the three dots at the right if you want to choose the file via a file selection dialog.
The file is usually present in the "java" subdirectory of the openFT installation directory.

30.3 The 'Logging / Traces' sheet

You configure the WinAdmin logging/trace functions in this sheet. The mode of operation is described in more detail in the section "Trace function" on page 161.

The dialog sheet has the following boxes:

'Logging / Traces' section

- 'Level' dropdown box
Allows you to set the trace level.

- 'File' combo box

Here you can specify the file to which WinAdmin is to write the trace records. By default, the "logs" directory below the WinAdmin installation directory is intended for the logging files. You can use the button with the three dots on the right to select a logging file. The file does not already have to exist. If necessary, WinAdmin creates a new file.

Alternatively, logging information can also be sent to "System.out". You can set this option via the combo box's dropdown box. This contains the corresponding entry "Log to System.out". On Windows systems, trace records that are sent to "System.out" are lost.

- 'Max. Nbr. of Files' dropdown box

This combo box allows you to select the maximum number of logging files.

- 'Max. File Size (KB)' dropdown box

Here you select the maximum size for a logging file in kilobytes. If the logging file grows beyond this size then it is renamed and the current logging file is cleared and written from the beginning again.

'UPIC Trace' section

The UPIC traces can be configured in this section. The trace records are written to the same file as has been specified in the "Logging /Traces" section.

- 'UPIC Trace' check box

Here you activate or deactivate the general UPIC trace.

- 'RFC1006 Trace' check box

Here you activate or deactivate the UPIC RFC1006 trace.

- 'UPIC Data Trace' check box

Here you activate or deactivate the UPIC data trace.

This trace option generates a very large number of trace records and should only be used in certain circumstances for diagnostic purposes.

- 'RFC1006 Data Trace' check box

Here you activate or deactivate the UPIC RFC1006 data trace.

This trace option generates a very large number of trace records and should only be used in certain circumstances for diagnostic purposes.

- 'Encoding Trace' check box

Here you activate or deactivate the UPIC encoding trace.

This trace option generates a very large number of trace records and should only be used in certain circumstances for diagnostic purposes.

31 Online help

Online help is available in any situation by pressing the F1 key. It is mostly context sensitive, i.e. in general a help text suitable for the current task will be provided. For example, if the F1 key is pressed in a dialog, the section of the online help concerning this dialog box will be displayed. The help basically refers to WinAdmin program handling, but it also contains descriptions of the objects and parameters required for the administration of UTM applications. Refer to the manuals "Administering Applications" and "Generating and Handling Applications" for more detailed information on the subjects of administration and configuration.

The content of the online help is available by choosing 'Help Topics' from the '?' menu. The dialog box that is then opened also contains the tabs 'Index' and 'Find' which you use to search the online help capability by means of keywords. If you are not yet familiar with using a Windows help system, the item 'Content' from the 'Help' menu will provide information.

32 Appendix

32.1 The menu bar

32.1.1 File menu

- **New Collection... (Ctrl+N)**
Allows you to define a new collection.
- **Open Collection... (Ctrl+O)**
Allows you to open an existing collection.
- **Import Collection File (*.utm)**
Allows you to import a WinAdmin V6.0 collection file.
- **Close Collection**
Closes the collection that is currently open.
- **Edit Collection**
Allows you to edit the collection that is currently open.
- **WinAdmin Users**
Displays the list of WinAdmin users.
- **Hosts**
Displays the list of hosts.
- **UTM Applications**
Displays the list of UTM applications.
- **UPIC Connections**
Displays the list of UPIC connections.
- **Collections**
Displays the list of defined collections.
- **Actions**
Displays the list of actions.
- **Import / BeanConnect container.properties file...**
Allows you to import the host, the UTM application and the UPIC connection stored in a BeanConnect container.properties file.
See section Importing the BeanConnect container.properties file on page 143.
- **Send File...**
Allows you to send a file to a remote system.

- **Get File...**
Allows you to get a file from a remote system.
- **Show File...**
Allows you to select and display a file located in the local system. The file is output in 'dump mode'.
- **Show File in Dump Mode...**
Allows you to select and display a text file located in the local system.
- **1 <collection-name1>
2 <collection-name2> etc.**
Opens a collection that was open in an earlier program session.
- **Exit**
Exits the WinAdmin session.

32.1.2 View menu

- **With Navigation Tree and Protocol Window**
Displays the navigation area and protocol window.
- **Navigation Tree**
Shows or hides the navigation area.
- **Protocol Window**
Shows or hides the protocol window.
- **Without Navigation Tree and Protocol Window**
Hides the navigation area and protocol window.
- **Focus Navigation Tree**
Sets the focus to the navigation tree.
- **Focus Work Area**
Sets the focus to the work area (if at least one work area window is open).
- **Focus Protocol Window**
Sets the focus to the protocol window.

32.1.3 Extras menu

- **Minimize and Lock**
Minimizes the WinAdmin program window and locks it. To restore the window, it is necessary to enter the password for the current WinAdmin user.
- **Settings...**
Opens the settings dialog box.
See section Settings on page 164.

32.1.4 Window menu

- Close All Windows
Closes all the open work area windows.
- Cascade
Arranges all the work area windows so that they overlap from the top left to the bottom right.
- Tile Horizontally
Arranges the work area windows in sequence from the top to bottom of the screen. This command is not available if too many windows are open.
- Next Window / Previous Window
If more than one window is open then the next or previous window is moved to the foreground.

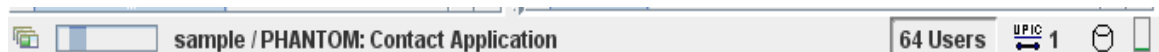
32.1.5 Help menu

- Content
Displays the contents of the online help.
- About
- Opens the dialog box 'About openUTM WinAdmin'..

32.2 The status bar

The status bar displays information concerning the current status of WinAdmin and the keyboard. In the case of a number of time-consuming operations a progress indicator is also displayed in the status bar.

The controls in the status bar:



- (1) Background activity indicator
This icon is displayed if a background activity is currently being executed.
- (2) Progress indicator
This is displayed for many time-consuming actions in order to display the progress of the action.
- (3) Text box
Contains general information, e.g. the action that is currently being carried out or an explanation of a chosen command.


■ (4) Text box

Contains information about the active window, e.g. number of objects, position of the insertion point. In the case of object lists, and, if there is an active display filter and at least one object has been excluded, the number of objects are displayed in the list.

Therefore, for example: "23 (1544) Users" means:


- The list contains a total of 1544 objects
Currently only 23 of the 1544 objects are displayed (because a filter is active).

■ (5) Communication requests

If open communication requests are present then the corresponding number is displayed next to the  icon.



See section "Communication requests" on page 25.

■ (6) Open transaction indicator

If at least one UTM application in the currently opened collection has an open transaction with an administration job that has not yet been terminated then the icon  is displayed here.

See section "Transaction-oriented administration jobs" Transaction-oriented administration jobs on page 97.

■ (7) Configuration database activity indicator

If the  icon is displayed here then this means that a database call is active. If no database call is active then the  icon is displayed.

■ (8) Memory utilization indicator

Indicates the program's current memory utilization.

The height of the column displayed in the rectangle indicates the memory utilization. As long as the memory utilization is below 50% of the maximum value, the column is displayed in green; between 50 and 75% it is yellow and above 75% it is red.

The tooltip associated with the icon indicates the memory utilization as both a percentage and absolute value. The absolute specifications are output in MiB (mebibytes, 1 MiB = 2²⁰ bytes = 1024 * 1024 bytes).

32.3 Complex filter masks

In various situations (when filtering the content of object lists or searching for text in files), WinAdmin allows you to enter a "complex" filter mask to select information.

This type of filter mask can consist of one or more tokens. The tokens are separated from one another by spaces. If a token is to consist of several words and/or if the case of the token is to be taken into account during the search then the token must be surrounded by double quotes.

The following types of token exist:

■ Plus token

If there is a plus sign in front of the token then the text of the token **must** be present in the information if it is to be selected.

■ Minus token

If there is a minus sign in front of the token then the text of the token **must not** be present in the information if it is to be selected.

■ Or token

Tokens that are preceded by neither a plus nor a minus sign are Or tokens. **At least one** of the specified Or tokens **must** be present in the information if it is to be selected.

A filter mask consisting of only one Or token is identical to a mask with a plus sign and the same text.

Example:

The filter mask







either or +must +"EXACTLY" -"must not"


















consists of five tokens, namely the Or tokens "either" and "or", the Plus token "must", the case-sensitive Plus token "EXACTLY" and the case-sensitive Minus token "must not".
















Information is selected if it contains either the text "either" or the text "or" irrespective of case. It must contain the text "must" and, once again, case is not considered. It must also contain the text "EXACTLY" in uppercase characters. Finally, it must not contain the lowercase text "must not".







32.4 Displayed icons

32.4.1 Object icons









Administration	Object type
	Abstract Syntax
	Access Point
	Application Context
none	Area
	Bcamappl
	Cluster Nodes
	Con
none	Create Control Statements
none	Database




















Administration	Object type
	Printer
	Print job
	Edit
none	Entry
none	Exit
	Gssb
	IP Subnet
	Kset
	Kset (deleted)
	Load Module / Shared Object
	Lpap
	Lses
	Lses (deleted)
	Ltac
	Ltac (deleted)
	Lterm
	Lterm (deleted)
	Message (fetched from the UTM queue)
	Message collectors


















Administration	Object type
	Message module
	Message queue entry
none	Modules
none	Mpool
	Mux
OSI 	Osi Association (created)
OSI 	Osi Association (generated)
OSI 	Osi Con
OSI 	Osi Lpap
	Program
	Program (deleted)
PTC	Open Transaction with state PTC
	Pterm
	Pterm (deleted)
	Queue (temporary)
	Queue (temporary, deleted)
none	Rmxa
none	Role
none	Satsel
	Sfunc
	Tac






Administration	Object type
	Tac (deleted)
	Tacclass
none	Tcbentry
none	Tls
	Tpool
	Transfer syntax
none	Uls
	User
	User (deleted)

32.4.2 Other icons

Icon	Meaning
	Application collection
	WinAdmin Objects subtree (collection window)
	Open Windows subtree (collection window)
	Administration subtree (collection window)
	Configuration subtree (collection window)
	Subtree containing the UTM clusters for the open collection (Administration)
	Subtree containing the UTM clusters for the open collection (Configuration)
	Subtree containing the standalone applications for the open collection (Administration) In addition, application list in the 'Standalone' subtree (Administration) and 'Cluster Applications' in a cluster's subtree (Administration).

Icon	Meaning
	Subtree containing the standalone applications for the open collection (Configuration) In addition, application list in the 'Standalone' subtree (Configuration) and 'Cluster Applications' in a cluster's subtree (Configuration).
	UTM cluster (Administration) at which at least one of the activated node applications was available the last time contact was established
	UTM cluster (Administration) which has not yet been contacted
	UTM cluster (Administration) There is a transaction-oriented administration job that has not yet been concluded on at least one activated node application in the cluster.
	UTM cluster (Administration) at which no activated node application was available the last time contact was established
	UTM application (administration) If the icon describes a single administration application, the application was available when last contacted.
	UTM application (administration) A transaction-oriented administration job has not yet been concluded for this application.
	UTM application (administration) that has not yet been contacted.
	UTM application (administration) that was not available when last contacted.
	Application list (administration)
	Deactivated UTM cluster node application (Administration)
	Application parameter (administration)
	Object selection (administration)
	Command mode
	Security view (administration)
	Coupler (Security view)
	No object found (Security view)
	Security resources (Security view)
	Privileged object (Security view)

Icon	Meaning
	Non-privileged object (Security view)
	Protected object (Security view)
	Unprotected object (Security view)
	Host (administration)
	UTM application (WinAdmin object)
	UPIC connection (WinAdmin object)
	WinAdmin user (WinAdmin object)
	WinAdmin user with overall privilege (WinAdmin Users' subtree in the collection window)
	WinAdmin user without overall privilege but with at least one application-specific privilege (WinAdmin Users' subtree in the collection window)
	WinAdmin user without overall privilege or any application-specific privilege but with at least one object type-specific privilege (WinAdmin Users' subtree in the collection window)
	WinAdmin user without overall privilege, application-specific or object-specific privilege (WinAdmin Users' subtree in the collection window)
	Active statistic collector (Statistics)
	Active statistic collector currently requesting a statistic value (Statistics)
	Sleeping statistic collector (Statistics)
	Inactive statistic collector (Statistics)
	Statistic collector in which the statistic value last collected exceeds the defined upper threshold (Statistics)
	Statistic collector in which the statistic value last collected exceeds the defined lower threshold (Statistics)
	Statistic collector whose assigned "collecting" object no longer exists in the relevant UTM application (Statistics)

Icon	Meaning
	Statistic collector in which the last attempt to collect a statistic value failed (Statistics). Possible cause: The statistic value to be collected is not available in the operating system version or variant of the relevant UTM application.
	List column
	Displayed text file
	Action
	Category node in the subtree of a cluster or a UTM application.

32.5 Shortcuts

Key combination	Meaning
General functions	
F1	Calls online help.
Ctrl+Alt+M	Minimizes and locks WinAdmin.
Ctrl+F4	Closes the active window (view).
Ctrl+F6, Alt+F9	Switches to the next open window (View).
Alt+F10	Returns to the last open work area window.
CTRL+Shift+F 2	Shows or hides the navigation window.
F3	Displays the object selection of the open collection.
Alt+0	Places the input focus in the collection window.
Shift+F2	
Shift+F3	Sets the input focus to the work area (if at least one work area window is open).
Shift+F4	Sets the input focus to the protocol window.
Alt+1	Places the input focus in the collection window, opens the 'WinAdmin Objects' subtree and closes the other subtrees.
Alt+2	Places the input focus in the collection window, opens the 'Display Filters' subtree and closes the other subtrees.
Alt+3	Places the input focus in the collection window, opens the 'Open Windows' subtree

Key combination	Meaning
	and closes the other subtrees.
Alt+4	Places the input focus in the collection window, opens the 'Collection...' subtree and closes the other subtrees.
Alt+5	Places the input focus in the collection window, opens the 'Actions' subtree and closes the other subtrees.
Ctrl+N	Creates a new application collection.
Ctrl+O	Opens an application collection.
Selecting, copying, pasting, deleting	
Ctrl+A	Selects all the objects in the active object list.
Ctrl+C Ctrl+Ins	Copies the objects selected in the active object list to the clipboard.
Ctrl+V Shift+Ins	Pastes objects from the clipboard.
Del	Removes the selected objects from an object selection or an object group.
Text file	
Ctrl+F	Opens the dialog box that allows you to search for texts or set bookmarks.
Alt+F	Opens the dialog box in which you can find particular areas of text and set bookmarks.
Statistics	
Ctrl+U	Toggles the display mode if the Statistics window is active.
Other	
F5	Refreshes the active window.
Alt+Enter	Displays the property dialog for the selected object or objects.
Ctrl+L	Moves to the next column set if an object list with multiple column sets is active.
F3	Displays the object selection (administration objects).
Shift+F3	Displays the object selection (configuration objects).

32.6 Graphical interface terminology

As the various Windows controls are often given different names, the terms used in this manual are listed briefly below and described and/or illustrated.

■ Work area window

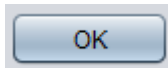
The individual windows that are displayed in the WinAdmin work area are known as *work area windows*.

You can switch between the individual open work area windows using the keyboard shortcut **Ctrl+F6**. The keyboard shortcut **Ctrl+F4** closes the active view. In addition, you can use the shortcuts **Alt+F9** and **Alt+F10** to switch to the "previous" or "next" (if present) work area window (similarly to the "History" buttons in a browser).

■ Button

A click on a button generally causes an action to be carried out. The text on the button describes what will happen when the button is clicked.

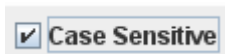
Example:



■ Check box

A check box is activated or deactivated (checked or unchecked) with a mouse click.

Example:



■ Click (or mouse click)

This generally means pressing the left mouse button. If the right mouse button is to be pressed, then that is expressly mentioned. You click with the left mouse button, for example, to activate windows, select entries in lists, move the input focus to a control etc.

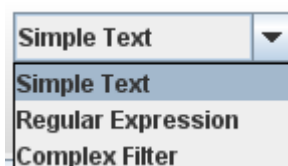
Clicking with the right mouse button often opens a context menu for the object at which the mouse pointer is pointing.

Note: The function of the left and the right mouse keys can be swapped in the Windows Control Panel.

■ Combo box

A combo box is a combination of an edit box and a dropdown box. You can either enter a value in the edit box or select an entry from the dropdown box. The selected value is then inserted in the edit box.

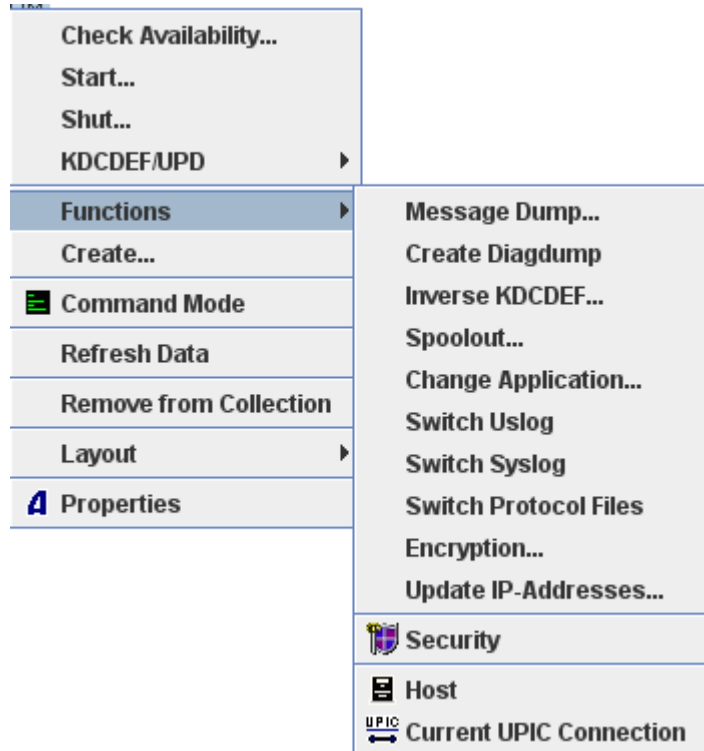
Example:



■ Context menu (or pop-up menu)

Context menus may be displayed when you click with the right mouse button when the mouse is pointing at an object. Context menus always offer commands that can be applied to the current object. Context menus are hidden again when you click on and choose a command or click anywhere outside of the context menu. The context menu can also be closed by pressing the Esc key.

Example:



■ Dialog box

A dialog box contains controls (operating elements) of various types and is generally used to make settings or carry out actions.

Most dialog boxes are "modal", i.e. while they are displayed all the other controls belonging to the application (menus, etc.) are unavailable. You must close such dialog boxes before you can use the other controls again.

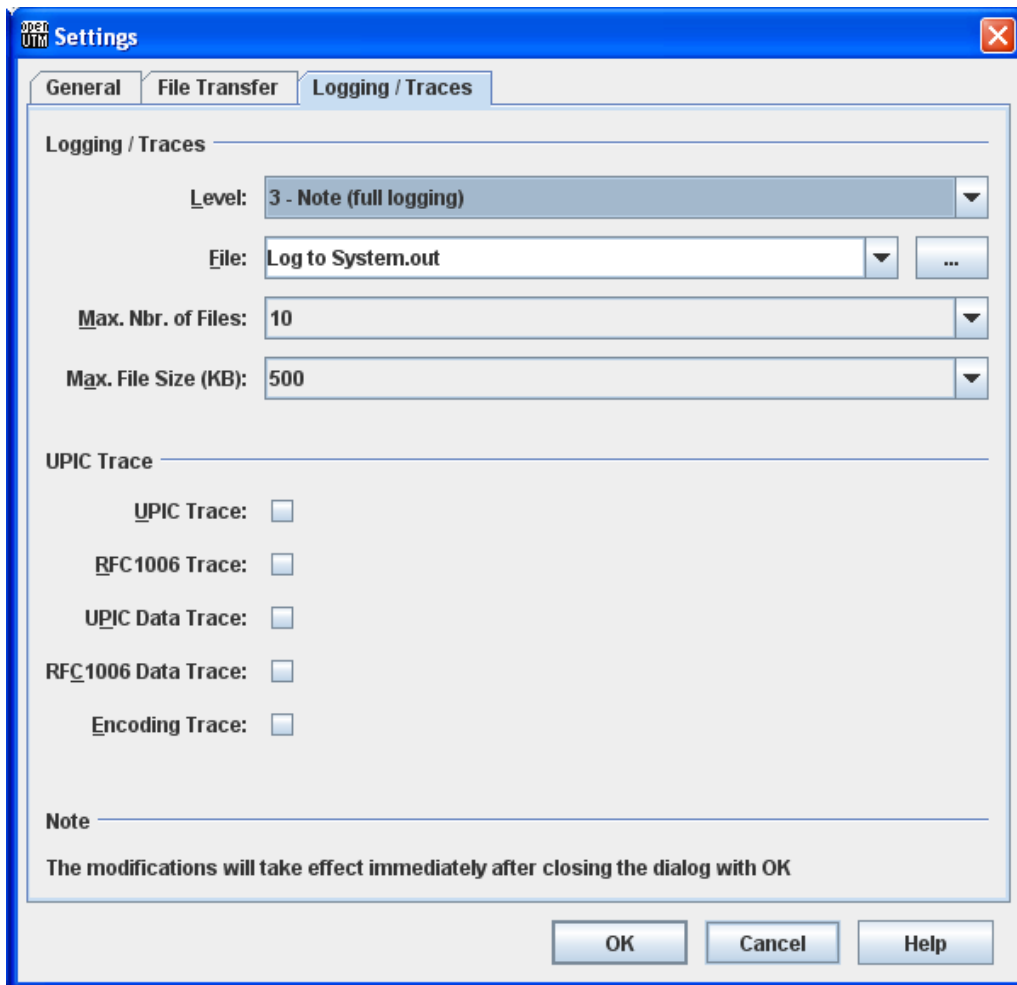
Dialog boxes generally have an 'OK' and a 'Cancel' button. Both of these buttons close the dialog box. If the 'OK' button is clicked, any settings you have made are accepted. However, if you click the 'Cancel' button the original state is restored. Pressing the Esc key has the same effect as clicking the 'Cancel' button.

The emphasized button is the default button (in this example the OK button). If the Enter key is pressed in a dialog box, the effect is the same as clicking the default button.

If the input focus is on a button (cancel button in this example), then you can simply press the space bar to activate the button.

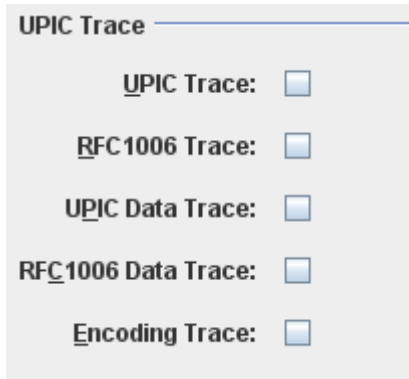
The input focus can be switched from one operating element to another by pressing the TAB key. The order is defined by the application ("TAB sequence"). You can press Shift+Tab to move through the controls in the opposite direction.

Example:



■ **Dialog section**

Controls that form a logical unit are grouped together in a dialog section.



■ Dropdown box

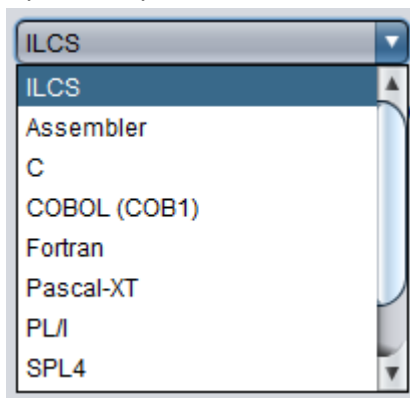
Basically, a dropdown box is a selection list that is "closed" in normal state. Only the entry that is currently selected in the selection box is visible. However, if you click on the arrow button (downwards or anywhere in the closed box) the entire selection box is displayed. A click on one of the entries in the dropdown box selects the item and closes the box again. Clicking on the arrow button or pressing the **Esc** key will also close the list box again.

Example:

Closed dropdown box



Opened dropdown box

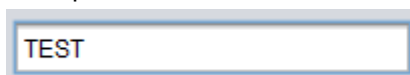


■ Edit box

An edit box enables you to enter a value (number or text).

In general, values in the clipboard can also be copied into an edit box or selected values in an edit box can be copied to the clipboard.

Example:



■ Insertion point

The insertion point is not to be confused with the mouse pointer. It specifies the point at which you can enter text in edit boxes or text boxes. You can usually set the insertion point by clicking with the mouse. It appears, for example, as a blinking vertical line or an inverted blinking rectangle. The blink frequency can be set in the Windows Control Panel.

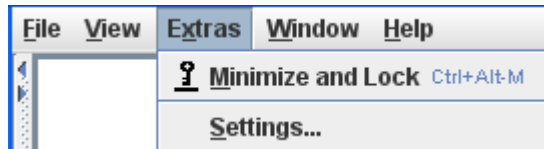
■ Insertion point

The insertion point should not be confused with the mouse pointer. It indicates the position at which new material is inserted in edit boxes or text windows. The insertion point can usually be set by clicking with the mouse. It is represented, for example, by a flashing vertical line or a flashing inverted rectangle.

■ Menu

The menu can generally be found below the title bar of the main window of a Windows application. You can select and execute actions from the menus and, in some cases, make settings.

Example:



■ Pointer (or mouse pointer)

The pointer indicates the current mouse position on the screen. If it is pointing at a specific object it may change its appearance. This generally means that a particular operation can be carried out by clicking or dragging with the mouse.

■ Pointer (or mouse pointer)

The pointer indicates the current position of the mouse on the screen. If the pointer points to specific objects then its appearance may change. This usually indicates that specific operations can be performed at these positions by clicking or "dragging" the mouse.

■ Property dialog

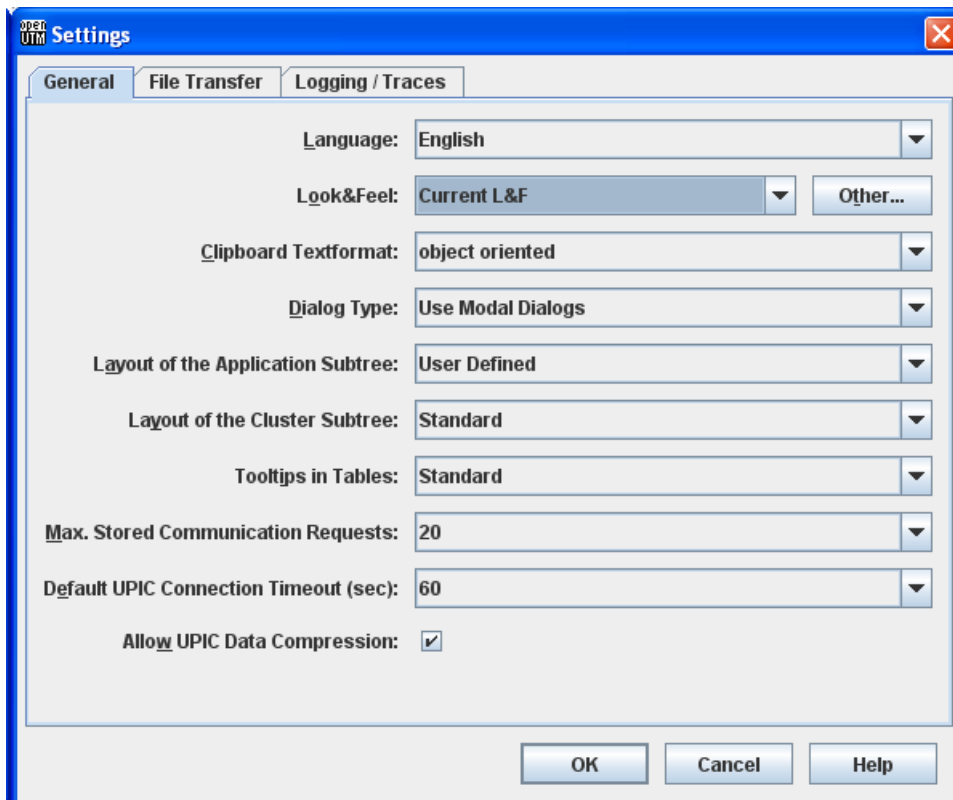
A property dialog or property sheet allows you to view and to modify object properties. A property dialog consists of one or several sheets or pages that are arranged like tabs overlapping one another. Each sheet has a tab with the sheet's name. Only one sheet is fully visible at any given time. To view a sheet that is not fully visible, you must click the tab. The sheet is then displayed (foregrounded). You can also move through the sheets using the arrow keys (left/right or up/down) if one of the tabs has the input focus.

Basically, each tab sheet represents a dialog box in its own right.

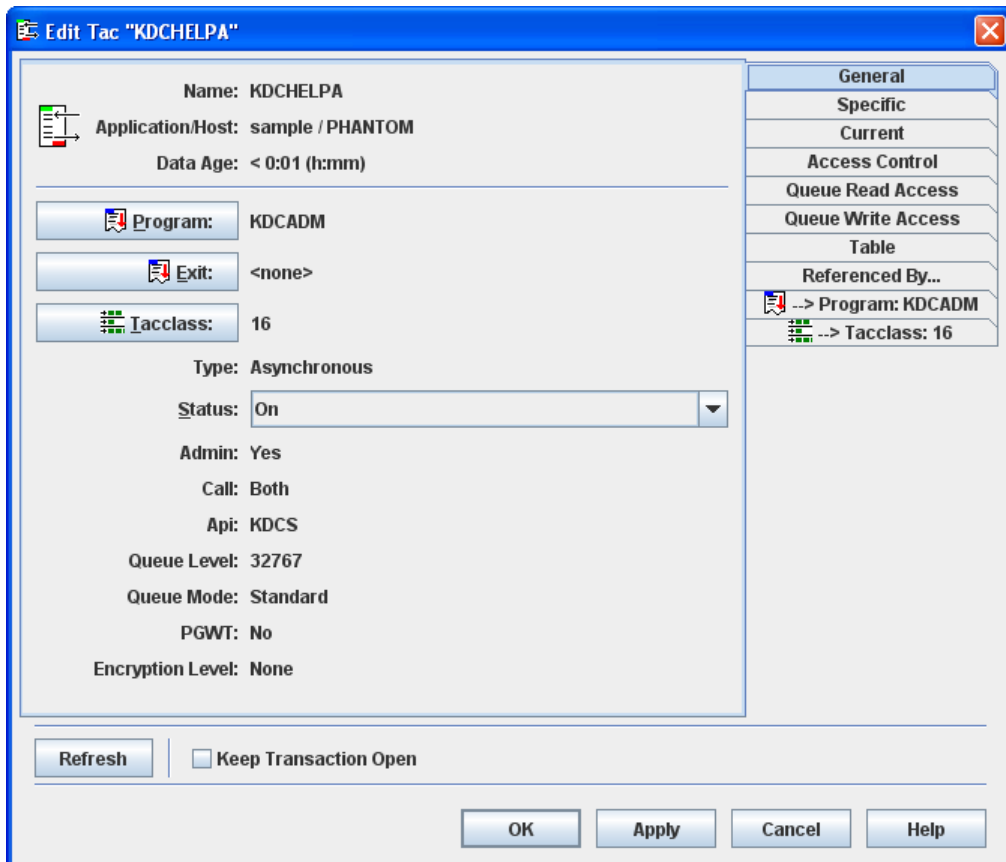
Below the sheets you will generally find the 'OK', 'Cancel', 'Apply' and 'Help' buttons.

A click on the 'OK' button will accept any settings made (on all of the sheets) and close the property dialog. A click on the 'Apply' button will also accept any settings made, but it does not close the dialog

Example:



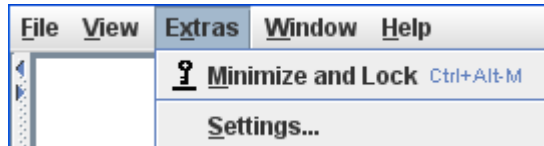
The tabs in the individual dialog sheets can also be arranged along the right-hand edge of the dialog. In WinAdmin, they are arranged in this way in UTM object property dialogs:



■ Menu

The menu can generally be found below the title bar of the main window of a Windows application. You can select and execute actions from the menus and, in some cases, make settings.

Example:



■ Radio button

If radio buttons occur then there are generally at least two of them. They allow you to select just one of the options offered and they are normally grouped together in a dialog section. If you click on a radio button (or on its text) in a group then that button is selected. Any previously selected radio button in the group is deselected (like the station keys on an old car radio: if one is pressed, the other pops out again).

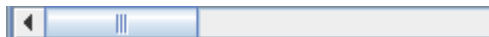
Example:



■ Scroll bar

You use the scroll bar to change the displayed section of a long list, a text window, a dropdown box, etc.

Example:



33 Index**A**

Abort Bound Services.....	73
access lists.....	83
actions.....	118, 139
actions infobox.....	156
Application subtree	
layout.....	45
asynchronous job.....	94, 96, 100, 109, 110
authorizations.....	29, 127

B

button.....	181
-------------	-----

C

check box.....	181
Cluster subtree	
layout.....	45
clusters.....	76
collection	
opening automatically.....	41
column condition	
defining.....	58
suggestion.....	58
column sets	
defining.....	58
suggestion.....	59
combo box.....	181
command mode.....	74, 136
communication requests.....	25
complex display filters.....	53, 54
complex filters.....	173
configuration database.....	22
troubleshooting.....	23
context menu.....	182

D

Dead Letter Queue.....	102, 135
diagnostic trace.....	161, 166
dialog box.....	182
display filters.....	43, 53, 58
complex.....	53, 54, 173
filter criterion.....	55
dropdown box.....	184

E

edit box.....	184
encryption.....	72
Explicitly Sign-Off User.....	73

F

file	
get.....	144, 145
send.....	145
show.....	137, 145
file transfer.....	31, 69, 70, 129, 144
configuration.....	166
filter criterion.....	55
filter mask	
complex.....	172
filters	
complex.....	173
FTAC profile.....	130
FTP.....	129

H

hosts.....	30, 33
------------	--------

I

icons.....	173, 176
import BeanConnect container.properties file.....	143, 169
incompatibilities.....	17
insertion point.....	184
installation.....	12

K

keyboard shortcuts.....	179
-------------------------	-----

L

layout	
of the Application and Cluster subtrees.....	45
linked objects.....	63
list columns	
column sets.....	58

condition.....	58
list generation.....	141
logging.....	166

M

menu	185, 187
message	
of asynchronous job	102
message collectors	91, 108
Message Dump	71, 138
message queues	94, 96, 100, 109, 110
mouse pointer	see pointer, see pointer

O

object list	
partial list.....	64
object lists.....	47
property dialog.....	57
object selection	75
objects	
context menus.....	62
copy	156
creating	68, 74
deleting	68, 74
icons.....	173
linked.....	63
properties.....	66
selection.....	75
online help.....	168
Online Import	73
openFT.....	129
configuration	166

P

partial list	64
adding.....	65
removing.....	65
pointer	185
pop-up menu.....	182
print jobs.....	104, 106
printer control Lterm.....	104
printers	104, 105
privileges	
application-specific	42, 128
object-specific.....	42, 128
overall	127
property dialogs	66, 185
Protocol files	
switch	72

Q

queue	
service-controlled	91, 100, 108

R

radio button.....	185, 187
RCCC	138
RCDC	138
refresh all application data	74
refresh data	161
roles	83
RSA keys.....	72

S

scroll bar	187
security	29, 127
security view	87, 110
selection.....	75
service-controlled queue.....	91, 100, 108
settings	164, 170
shortcuts	179
Show Open Transactions in PTC State	73
shut-down.....	21
sign-on service	19
sort.....	61
start	21
statistic	
types.....	119, 120
statistics	111
collector icons	112
collector list.....	112
collectors	117
copying/saving values	120
creating statistic collectors	116
deleting collectors	117
state of a collector	113
threshold alert	118, 139
status bar	171
symbols	173, 176
system platforms	18

T

table tooltips	159
TAC statistics.....	62
text file	
highlighting identical words.....	147, 154, 155
searching.....	149
selecting	147
tooltips	159
trace	166
transaction-oriented administration jobs.....	74, 97, 172

U

UPIC	
sign-on service.....	19
UPIC connections	36
user information	
for asynchronous job	101

UTM access control.....	73, 81
UTM applications.....	33
UTM clusters.....	76

W

WinAdmin objects	28
WinAdmin users.....	29, 127
work area window	181
work area windows	24