

openUTM WebAdmin V7.0



**Web user interface
for administering
openUTM**

1 Contents

1	Contents.....	2
2	Summary of contents.....	6
3	Installation.....	7
3.1	Update installation of Apache Tomcat.....	8
4	What's new.....	9
4.1	...in Version 7.0?.....	9
4.2	...in Version 6.5?.....	9
5	Incompatibilities.....	10
6	General.....	11
6.1	System platforms.....	11
6.2	Resources required.....	11
6.3	Programming language.....	11
6.4	Linking to UTM applications.....	11
6.5	Supported UTM versions.....	13
6.6	Language.....	13
7	Starting and shutting down the web application.....	14
8	Launching the WebAdmin web page in a browser.....	15
9	Configuring the web server.....	16
9.1	Changing the port.....	16
9.2	Using a secure connection.....	16
9.3	Web application as a Windows service.....	16
10	Configuration database.....	17
10.1	Backup.....	17
10.2	Troubleshooting.....	17
11	The web user interface.....	18
11.1	Page types.....	18
11.2	Sidebar (primary navigation).....	19
11.3	Navigation bar (secondary navigation).....	20
11.4	The "forward" and "back" browser buttons.....	21
11.5	Bookmarking pages in the web browser.....	21
11.6	Using tabs in the web browser.....	21
12	WebAdmin objects.....	22

12.1	WebAdmin User	23
12.1.1	Displaying the WebAdmin user list.....	23
12.1.2	Creating a new WebAdmin user.....	23
12.1.3	Deleting a WebAdmin user.....	24
12.2	Hosts	24
12.2.1	Displaying the host list.....	25
12.2.2	Creating a new host	25
12.2.3	Deleting a host.....	25
12.3	UTM applications.....	25
12.3.1	Displaying the application list	27
12.3.2	Generating a new application	27
12.3.3	Deleting an application	27
12.4	UPIC connections.....	28
12.4.1	Displaying the UPIC connection list.....	29
12.4.2	Creating a new UPIC connection	29
12.4.3	Deleting a UPIC connection.....	29
12.5	Collections	30
12.5.1	Displaying the collection list	30
12.5.2	Creating a new collection	30
12.5.3	Deleting a collection	31
12.5.4	Opening a collection.....	31
12.5.5	Closing a collection.....	31
13	Object lists	33
13.1	Displaying object lists with WebAdmin objects	33
13.2	Displaying object lists with UTM objects	33
13.3	Structure of the object list page	34
13.4	Filtering the list objects.....	34
13.5	Sorting objects	35
13.6	Linked objects.....	35
13.7	Object list functions	35
13.8	Object-specific functions	36
14	Defining partial lists	37
14.1.1	Adding a partial list	37
14.1.2	Removing a partial list	38
15	Property pages	39
15.1	Displaying the property page for an object	39
15.2	Modifying object properties	39
15.3	Object-specific functions	40
16	Creating and deleting objects.....	42
17	Functions of a UTM application and UTM cluster.....	43
18	Support for the UTM cluster concept.....	48
18.1	UTM clusters and node applications in the web user interface	48
18.2	Defining a cluster in WebAdmin	48

18.2.1	UPIC users in the UTM cluster	49
18.3	Deactivating cluster nodes	50
18.4	Functions of a UTM cluster	50
18.5	Functions of a node application in the UTM cluster	50
18.6	Property page for a UTM cluster and node application	50
18.7	List of UTM cluster nodes	51
18.8	Object lists of a cluster	51
19	Support for UTM access control	53
19.1	Accessing UTM services – keys and locks	53
19.1.1	"Services"	53
19.1.2	"Users"	54
19.1.3	"Couplers"	54
19.2	The "new" concept – roles and access lists	55
19.3	Roles	56
19.3.1	Roles in object lists	57
19.3.2	Roles in property pages	57
20	WebAdmin message destination – message collectors	59
20.1	Actions on the UTM side	59
20.2	Actions on the WebAdmin side	60
20.3	Properties of message collectors	60
20.4	Message collectors in the cluster	61
20.5	The collected messages list	62
20.6	Tips and notes on generation	62
21	Transaction-oriented administration jobs	64
22	Administration of message queues	65
23	Administering printers and print jobs	69
23.1	Printer control Lterms	69
23.2	Administering printers	69
23.3	Administering print jobs	70
24	Service-controlled queues in WebAdmin	72
24.1	The queue list	72
24.2	Creating and deleting queues	73
25	Statistics	74
25.1	Statistics lists	74
25.1.1	Collector list	74
25.1.2	Statistics chart	77
25.1.3	List of statistics values for a statistic collector	78
25.2	Statistic collectors	79
25.2.1	Creating a statistic collector	79
25.2.2	Deleting a statistic collector	79
25.2.3	Statistic collectors in clusters	79

25.2.4	Functions for selected statistic collectors	79
25.3	Threshold Alert	80
25.4	Saving, deleting, exporting statistics values.....	81
25.5	Supported statistics types	82
26	Security - users and authorizations.....	89
26.1	Passwords and overall privileges	89
27	Other functions	90
27.1	Starting UTM applications.....	90
27.2	Command mode.....	92
27.3	Message dump.....	93
27.4	Actions	95
27.4.1	Saving object properties in a file	96
27.5	Tooltips on property pages	97
27.6	Refreshing data	99
27.7	Online help.....	100
27.8	Trace function.....	100
27.9	Tomcat logging.....	101
27.10	WebAdmin Audit Logging	102
28	WebAdmin as SE Manager add-on.....	103
28.1	Installation	103
28.2	File paths.....	103
28.3	Starting the WebAdmin add-on	103
28.4	Logging in at the WebAdmin add-on.....	104
28.5	Logging out from the WebAdmin add-on.....	104
28.6	Continuing the SEM session	104
28.7	Session timeout.....	104
28.8	Lifetime of messages and statistics values	105
28.9	Conduct of dialogs.....	105
28.10	SE management units.....	106
28.11	Backing up and restoring the configuration data	106
28.12	SEM Audit logging.....	106
29	Appendix	107
29.1	Icons used.....	107
29.1.1	Object icons	107
29.1.2	Other icons.....	109
29.2	User interface terms	111
30	Index.....	118

2 Summary of contents

This document describes the functionality of Version 7.0 of the openUTM WebAdmin program. Like the standalone openUTM WinAdmin program, openUTM WebAdmin can be used to administer UTM applications. In contrast to openUTM WinAdmin however, you can administer applications from a web browser such as Internet Explorer from Microsoft or Mozilla Firefox.

After a single instance of openUTM WebAdmin is installed centrally, it then provides a web application that can be accessed from any client computer on which a web browser is installed. openUTM WebAdmin is supplied together with the Apache Tomcat 7 web server in which the web application is automatically deployed on initial start-up.

The web application uses the back-end classes of openUTM WinAdmin to access the UTM applications to be administered. Connections to these UTM applications are then established at one central point (in the web application). In exactly the same way as in WinAdmin, the configuration data of WebAdmin is stored in a configuration database which is then centrally accessed by the web application. A configuration database created in WinAdmin can be used.

3 Installation

Before installing openUTM WebAdmin, a Java JDK, version Java SE 7 or later, must be installed on the destination computer. This can be downloaded from Oracle's web page (<http://www.oracle.com/technetwork/java/javase/downloads/index.html>) free of charge. If the computer has a 64-bit architecture and is running a 64-bit operating system, then it is also advisable to install a 64-bit version of Java. It is also possible for multiple versions of Java to be installed, for instance by installing 64-bit Java in addition to an existing 32-bit Java installation.

Depending on the version of Java installed, the Apache Tomcat web server (and therefore also the web application) runs either as a 64-bit or as a 32-bit application. The 64-bit version has the benefit of a larger address space, which can be advantageous for applications with lots of objects.

An IzPack installation in the form of the JAR file "WebAdmin70A00.jar" is available.

If the IzPack installation of WebAdmin is chosen then the user is free to select two main directories:

- The basic WebAdmin installation directory (main program directory) which contains programs and procedures for running Apache Tomcat and WebAdmin.
- The WebAdmin main data directory for configuration and log files.

In this way, you are able to separate data and code. However, you can also choose the same main directory for both.

Note:

If you want to use WebAdmin as a Windows service, you have to choose the same directory for data and code.

Installing WebAdmin in systems with a graphical user interface

In systems with a graphical user interface, proceed as follows:

1. When installing on a Unix/Linux platform the environment variable DISPLAY must be set.
2. In the relevant system, open a window for command entry, e.g. shell or DOS prompt.
3. Switch to the directory in which the JAR file is located.
4. Unpack the JAR file using the following command
`„<javapath>/bin/java -jar WebAdmin70A00.jar“.`
`<javapath>` is the directory in which the Java JDK is installed.
5. Follow the instructions output by the graphical installation program and define the main program directory, the main data directory and the JDK directory.
 Attention: A Java JRE is not adequate for a successful installation!

Installing WebAdmin in systems without a graphical user interface

A `WebAdmin-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 `WebAdmin-auto.xml` with a text editor and enter the required installation information as follows:
 - a. Define the main program directory in the `<INSTALL_PATH>` tag and
 - b. the main data directory for configuration and logging data in the `<VAR_INSTALL_PATH>` tag and
 - c. the directory in which the Java JDK is installed in the `<JDK_PATH>` tag.
 Attention: A Java JRE is not adequate for a successful installation!
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 WebAdmin70A00.jar WebAdmin-auto.xml“.
This installs WebAdmin automatically.
Attention: Installing in systems without graphical user interface without using `WebAdmin-auto.xml`, this may lead to unexpected behaviors (e.g. null-pointer-exception).

3.1 Update installation of Apache Tomcat

When you install WebAdmin, a complete Apache Tomcat web server is also installed directly below the main program directory in the directory “apache-tomcat”. Below, this is referred to as the Tomcat main directory (CATALINA_HOME). The Tomcat version that is installed there contains an extension for logging via log4j.

If you want to use a more recent version of Apache Tomcat for WebAdmin, please proceed as follows:

1. Shut down the web application or Tomcat web server.
2. Back up the log4j configuration file for Tomcat with the path
“<Tomcat main-directory>/lib/log4j.properties”
to a location outside of the WebAdmin installation directories.
Caution: An old backup of this file must not be used!
Back up also the log4j.jar file.
3. Delete the Tomcat main directory.
4. Download the following files from the Apache Tomcat Download web site (tomcat.apache.org under “Download”, “Quick Navigation”, “Browse”):
 - a. Under “bin/”, the zip or gz archive containing the Tomcat product files
 - b. Under “bin/extras/”, the files
tomcat-juli.jar (which is larger than the file shipped by default) and
tomcat-juli-adapters.jar
5. Unpack the Tomcat product data archive to the main program directory and rename the new Tomcat main directory
from “apache-tomcat-x.y.z” to “apache-tomcat”.
6. Move the log4j extra files to the Tomcat main directory to
“**bin**/tomcat-juli.jar” and
“**lib**/tomcat-juli-adapters.jar”
and the backed up log4j configuration file back to
“**lib**/log4j.properties” and the backed up log4j.jar back to “**lib**/log4j.jar”.
7. Start the web application or Tomcat web server.

4 What's new...

4.1 ...in Version 7.0?

The most important changes compared to Version 6.5 are:

- WebAdmin supports all the new features of UTM V7.0, e.g. displays configuration data of UTM as a HTTP server.
- New script `instservice.cmd` can be used to set up WebAdmin as a Windows service.

4.2 ...in Version 6.5?

The most important changes compared to Version 6.4 are:

- WebAdmin supports all the new features of UTM V6.5, e.g. long hostnames and passwords of length 16.

5 Incompatibilities

This section lists the incompatibilities and the functions that are no longer present in WebAdmin V7.0 compared to earlier versions.

6 General

The openUTM WebAdmin program is used to administer individual or multiple UTM applications. The applications may be running on several different computers and under different operating systems (BS2000, Unix/Linux, Windows).

The program is a web application and provides a web user interface which can be accessed using a web browser on any client computer.

6.1 System platforms

openUTM WebAdmin runs on Microsoft Windows operating systems and on Unix or Linux platforms. It is a pure Java implementation. More information on the supported platforms can be found in the Release Note.

6.2 Resources required

A full installation of openUTM WebAdmin, including all the libraries and files used, requires approximately 70 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 interruption-free operation.

To run the web user interface of WebAdmin, ideally you need a screen resolution of at least 1024 x 768 pixels. If the resolution is any lower, some of the dialog boxes may not display in full on the screen.

6.3 Programming language

openUTM WebAdmin is written entirely in Java and requires a JRE (Java Runtime Environment), version Java SE 7, to be installed. This can be downloaded from Oracle's web page (<http://www.oracle.com/technetwork/java/javase/downloads/index.html>) free of charge and must already be installed **before** WebAdmin is installed on the destination computer.

6.4 Linking to UTM applications

The product JConnect, which is supplied and installed together with WebAdmin, is used for linking to the UTM applications that are to be administered.

UTM applications from Version 6.4 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 systems)
 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 and does not need to be explicitly linked.

In addition, an access must be generated for the UPIC client WebAdmin, 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 WEBADMIN, PASS=C'XYZ', PERMIT=ADMIN, RESTART=NO
```

A password is not mandatory for the WebAdmin 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 must be specified when defining the UTM application in WebAdmin (see section "UTM applications" on page 25) and must match the corresponding values in the generation of the UTM application.

For UTM applications in BS2000 systems the listener port is always implicitly set to 102.

For Unix/Linux and Windows applications, the listener port is specified in the BCAMAPPL generation statement with the LISTENER-PORT=<port> parameter.

For BS2000 applications, the selector format is always implicitly Transdata.

For Unix/Linux and Windows applications, the selector format can likewise be specified in the BCAMAPPL generation statement (TSEL-FORMAT=<format> parameter). The values T (Transdata), E (EBCDIC), and A (ASCII) are available. If this parameter is omitted in the BCAMAPPL statement, UTM uses the value Transdata if <bcapplname> uses the Transdata character set, otherwise it uses the value EBCDIC. It is advisable to explicitly specify the selector format using the TSEL-FORMAT parameter in the BCAMAPPL statement.

Notes on the UPIC sign-on service:

If a UPIC sign-on service is generated for the UTM application (SIGNON UPIC=YES), it is best to deactivate the sign-on service in the BCAMAPPL statement used for communicating with WebAdmin (BCAMAPPL <bcamapplname>,SIGNON-TAC=*NONE).

If a UPIC sign-on service is nevertheless assigned to the respective BCAMAPPL, then the KDCWADMI TAC must be taken into account when programming this sign-on service.

Access via LTERM / PTERM

As an alternative to access via a terminal pool (TPOOL statement), it is also possible to explicitly generate a connection for WebAdmin:

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

When defining the UPIC connection in WebAdmin, it is necessary to specify the PTERM name <upicpt>. In the case of an explicit UTM generation of a connection for WebAdmin, it is necessary to specify the name of the WebAdmin application host in <upicpro>. See the corresponding paragraph above for information on the listener port and selector format in the PTERM statement.

Advantage of an explicit connection for WebAdmin: Generating

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

makes it possible to minimize the response times of UTM applications under load.

6.5 Supported UTM versions

Version 7.0 of WebAdmin supports the administration of UTM applications from Version 6.4.

6.6 Language

The text of openUTM WebAdmin is in English or German. You can switch between these languages dynamically by clicking a button in WebAdmin's web user interface.

7 Starting and shutting down the web application

The script *startup.sh* (Unix/Linux systems) or *startup.cmd* (Windows systems) is available in the WebAdmin installation directory to start the Apache Tomcat web server (and with it also the web application).

The first time the web server is started, the web application is automatically deployed to the web servers.

The script *shutdown.sh* (Unix/Linux systems) or *shutdown.cmd* (Windows systems) is used to shut down the Apache Tomcat web server (and with it the web application). These scripts are likewise available in the WebAdmin installation directory.

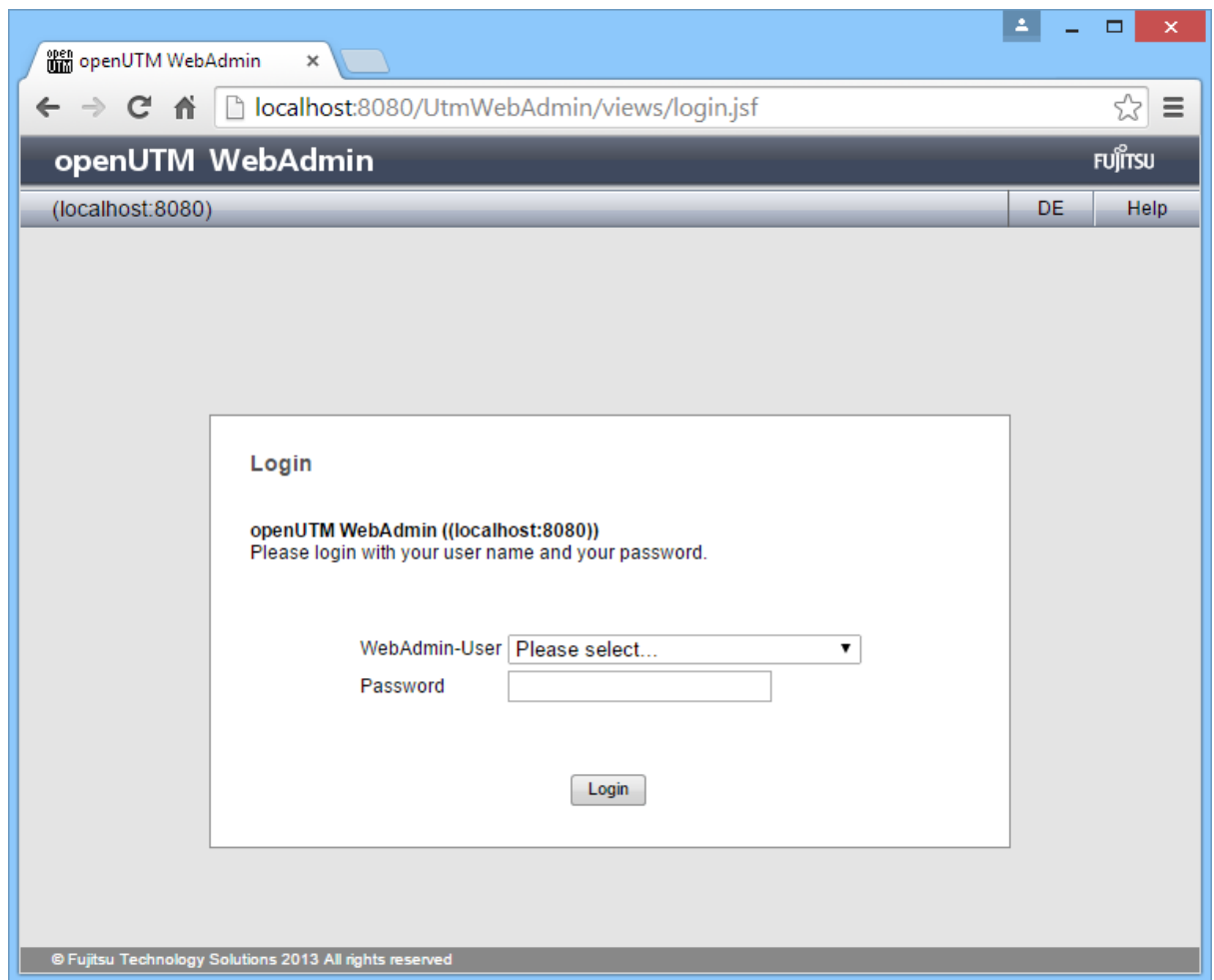
8 Launching the WebAdmin web page in a browser

The WebAdmin administration user interface can be launched from any client computer on which a web browser is installed. Open the web browser (e.g. Microsoft Internet Explorer) and enter the following address in the browser's address bar:

`http://<webadmin-host>:8080/UtmWebAdmin/`

Replace <webadmin-host> with the name of the computer on which the web application has been installed. The web application must of course already be running.

The login screen of openUTM WebAdmin is then displayed:



Choose the WebAdmin user that you want to log in as and enter the corresponding password. Then click the "Login" button to open the actual application itself.

9 Configuring the web server

The Apache Tomcat web server supplied with WebAdmin is ready to use and can be started immediately after installation. By default port 8080 is used, and the HTTP protocol is used for communication.

9.1 Changing the port

To change the port to be used, open the XML file *server.xml* located in *<WebAdmin-Directory>/UtmWebAdmin/conf* with a text editor. Only do this when the web server is not running.

This file contains the node `<Service name="Catalina">`, and within this node the subnode `<Connector port="8080".../>`. You can enter another port number here instead of default port 8080. After making the change, save the file and restart the web server. The WebAdmin application can then be accessed in the web browser at the new port number.

Example:

If you enter port 7070, the web application can then be accessed in the web browser at the address `http://<webadmin-host>:7070/UtmWebAdmin/`

9.2 Using a secure connection

It is also possible to configure the Apache Tomcat web server to use a secure connection between the web client and the web server (HTTPS protocol, SSL connection).

A description of how to do this is provided on the Apache Tomcat web page <http://tomcat.apache.org/tomcat-7.0-doc/ssl-howto.html> for example.

When reference is made in this description to Apache Tomcat configuration files (e.g. *server.xml*), these files are located in the directory *<WebAdmin-InstallationDirectory>/UtmWebAdmin/conf*. The configuration files in the Tomcat subdirectory are not used for the WebAdmin application.

9.3 Web application as a Windows service

If WebAdmin is installed on a Windows system, Apache Tomcat - and therefore launching the web application - can be set up as a service. The web application is then automatically launched when the computer is rebooted.

A description of how to do this is provided on the Apache Tomcat web page <http://tomcat.apache.org/tomcat-7.0-doc/windows-service-howto.html> for example.

To set up WebAdmin as a Windows service the script `instservice.cmd` can be used.

10 Configuration database

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

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

WebAdmin uses an H2 database. H2 is a SQL database written exclusively in Java that can run not only on Windows platforms, but on Unix and Linux platforms as well. It is available as freeware (at <http://www.h2database.com>), however the runtime system for the database is supplied and installed with WebAdmin.

The first time it is started, WebAdmin creates the configuration database in the directory **<WebAdmin-InstallationDirectory>/UtmWebAdmin/app/db**

It is also possible to use a configuration database created and used by openUTM WinAdmin. To do this, before starting the web server, the file **winadmindb.h2.db** of the WinAdmin configuration database must be copied in to the above-mentioned WebAdmin directory as **webadmindb.h2.db**. It is also possible to use (i.e. copy) a WebAdmin configuration database from an old installation.

10.1 Backup

In order to avoid loss of data caused by disk crashes or program errors, backup copies of the configuration database should be made at regular intervals.

To do this, you can simply copy the complete directory containing the H2 database to a safe location. This is best done when the database is not being used.

10.2 Troubleshooting

To avoid or rectify problems during the operation of the configuration database, pay special attention to the following points:

- The database or individual tables generated by WebAdmin must not be write-protected.
- The database may not be opened exclusively by any other process or be fully write-protected.

11 The web user interface

After you log in, the web user interface has the following elements:

The title bar is located at the top of the application window. The name of the currently active user is displayed in the title bar. You can click the "Logout" button to log off the web application again. The login screen is then displayed again. You can use another button on the title bar to toggle the user interface language between English and German. Click the "Help" button to call up the online help system.

The largest part of the application window below the title bar is divided into two areas. Down the left-hand side is the sidebar which is used for primary navigation within the application. At the top to the right of the sidebar is the navigation bar which usually contains several buttons used for secondary navigation. The respective content selected is displayed below this bar. This is generally object lists or the properties of individual objects.

The contents of the sidebar and the navigation bar are dynamic, the elements and buttons displayed differ depending on the type of page being viewed.

11.1 Page types

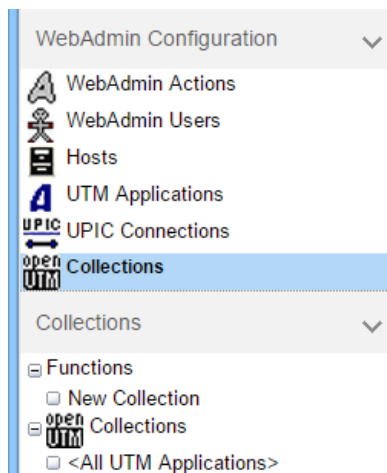
The web application uses the following page types:

- Login page
- Home page
- Object list pages for WebAdmin objects
WebAdmin objects are WebAdmin Users, Hosts, UTM Applications, UPIC Connections and Collections.
- Property pages for individual WebAdmin objects
- Primary page of the opened collection
- Primary page of a UTM application or UTM cluster
- Object list pages for UTM objects
UTM objects belong either to a standalone UTM application, to a UTM cluster, or to a node application of a UTM cluster.
- Property pages for individual UTM objects
- Function pages for a UTM application or UTM cluster
These are special pages for executing a function for a UTM application or UTM cluster, e.g. checking availability or terminating a UTM application.

11.2 Sidebar (primary navigation)

The sidebar consists of a number of individual elements known as "racks". It provides access to the individual object list pages, property pages and function pages. The contents of the sidebar, i.e. the racks currently being displayed, change dynamically depending on what type of page is being displayed. Each individual rack can be expanded and collapsed by clicking on ">" or "v" on the rack title bar.

Example:

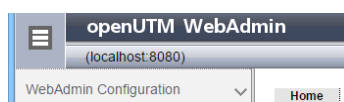


This sidebar appears immediately after you log in. It contains the two racks "WebAdmin Configuration" and "Collections".

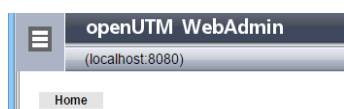
The sidebar can be expanded and collapsed by clicking on the following field at the top left edge of the WebAdmin page.



Sidebar expanded:



Sidebar collapsed:



The following sidebar racks exist:

- WebAdmin Configuration**
 Provides access to the object lists for the WebAdmin objects: WebAdmin actions, WebAdmin users, hosts, UTM applications and UPIC connections.
- Collections**
 Provides access to all the available connections. Clicking on one of the collections opens this collection and closes any other collection that is currently open.

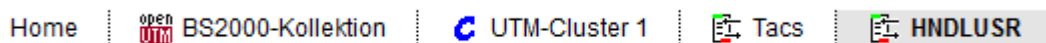
- UTM cluster
Provides access to the property page of the UTM cluster as well as to the cluster functions.
- UTM Application
Provides access to the property page of the UTM application as well as to the application functions.
- UTM Object List
Provides access to functions specific to the particular object list.
- UTM Object
Provides access to functions specific to the particular object.

11.3 Navigation bar (secondary navigation)

The navigation bar runs along the top of the content pane of the individual pages and contains a number of buttons which are used for secondary navigation within the web application. Different buttons appear depending on the type of page being displayed and are arranged hierarchically from left to right in the manner of "breadcrumbs navigation".

One of the buttons is highlighted in color to indicate the currently active page. This is usually the button on the far right.

Example:



Home > openUTM BS2000-Kollektion > UTM-Cluster 1 > Tacs > HNDLUSR

The following types of button exist:

- Home
This button is always present. It leads to the home page of the web application.
- WebAdmin object list
This button is present if a WebAdmin object list or the property page for an individual WebAdmin list object is being displayed. It leads to the respective object list page.
- WebAdmin object
This button is present if the property page for an individual WebAdmin list object is being displayed.
- Opened collection
This page is only displayed if a collection is open. It leads to the primary page of the opened collection.
- UTM cluster
This button is present if a page type that refers to a UTM cluster is being displayed. It leads to the primary page of the UTM cluster.
- UTM Application

This button is present if a page type that refers to a UTM application (standalone application or cluster node application) is being displayed. It leads to the primary page of the UTM application.

- **UTM Object List**

This button is present if a UTM object list or the property page for an individual UTM list object is being displayed. It leads to the respective object list page.

- **UTM Object**

This button is present if the property page for an individual UTM list object is being displayed.

- **Function**

This button is present if the page of an individual UTM function (e.g. Check Availability) is being displayed.

11.4 The "forward" and "back" browser buttons

As well as the sidebar and the secondary navigation buttons, it is also possible to use the browser's own "forward" and "back" buttons for navigating as usual. However, no data is stored on the client computer. This means that the page contents are retrieved from the web server again each time you use the "forward" or "back" button.

11.5 Bookmarking pages in the web browser

The web application is designed to enable individual pages to be bookmarked within the web browser. This allows you to bookmark object lists that you frequently use, for instance.

11.6 Using tabs in the web browser

Most browsers support tabbed browsing, with selection tabs arrayed along the top of the browser window. This makes it possible to quickly switch between different web pages within the browser window using the CTRL+Tab keyboard shortcut. If you hold down the CTRL key, clicking on a web link opens a new tab. The new tab is not yet selected however, i.e. the linked web page is not yet displayed. In this way you can quickly create tabs for several links displayed on one page. If you want to display the linked page on the new tab immediately, press CTRL+Shift while clicking on the link.

You can of course use this standard browser functionality for WebAdmin as well. For example, you can quickly open the properties dialogs of linked objects in an object list on different tabs then subsequently edit the objects one by one.

12 WebAdmin objects

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

You can also create, edit and delete the WebAdmin objects via the WebAdmin user interface.

For each WebAdmin object, there is a property page on which you can edit the properties of an existing object or define the properties of a new object you want to create. In addition, for each WebAdmin object type there is an object list page listing all the currently defined WebAdmin objects. These list pages have the same properties as the object list pages on which UTM objects are displayed (see "Object lists" on page 33).

The following types of WebAdmin objects exist:

- WebAdmin Actions

On the one hand, WebAdmin actions draw attention to a statistics threshold alert. On the other, they permit scheduled saves of the object properties of UTM applications in files on the WebAdmin server. See "Actions".

- WebAdmin User

WebAdmin user objects provide information about user authorizations, for example whether or not the user who is currently signed on to WebAdmin is permitted to edit administration data and objects, or has read access only.

- Hosts

Host objects describe the computers on which the UTM applications to be administered are running.

- UTM applications

These objects describe the properties of the UTM applications to be administered using WebAdmin.

- UPIC connections

These objects describe the properties of the UPIC connections between the WebAdmin application and the running UTM applications. To enable a UTM application to be administered using WebAdmin, a UPIC connection to this application must be defined.

- Collections

Applications that are to be administered jointly can be grouped together in collections. Only one application collection can be open at any one time within a WebAdmin session.

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

12.1 WebAdmin User

Objects of the type "WebAdmin user" are used to administer the authorizations of the individual WebAdmin users. Before working with the WebAdmin application, a user must always sign on by entering their WebAdmin user name. If a password is assigned to this WebAdmin user, then the user must also specify this before he or she can work with WebAdmin.

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 WebAdmin, it may be worthwhile setting up additional "WebAdmin users" and assigning these privileges as appropriate to the tasks they have to perform. See also the "Security - users and authorizations" section on page 89. Only a WebAdmin user with the master privilege is authorized to create, delete or modify a WebAdmin user.

The boxes on the property page:

- 'Name' edit box
You can select any name for a WebAdmin user. However, it has to be unique within the relevant configuration database. The name can be up to 28 characters long.
- '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
A comment relating to the WebAdmin user can be entered here.

12.1.1 Displaying the WebAdmin user list

To view a list of all the WebAdmin users defined in the configuration database, click "WebAdmin Users" in the "WebAdmin Configuration" rack on the sidebar. The "WebAdmin Configuration" rack does not always appear on the sidebar. You can display it however by clicking the "Home" button on the navigation bar.

12.1.2 Creating a new WebAdmin user

To create a new WebAdmin user, it is first necessary to display the WebAdmin user list. The "WebAdmin Users" rack is then displayed on the sidebar. Clicking on "New WebAdmin User" in this rack opens the page for defining the new WebAdmin user.

Once you have specified the properties for the new WebAdmin user, click the "Create Object" button and the new user will be created.

Only WebAdmin users with the master privilege are authorized to create a WebAdmin user.

12.1.3 Deleting a WebAdmin user

To delete a WebAdmin user, it is first necessary to display the property page for the respective user. First display the WebAdmin user list, then click on the name of the WebAdmin user to be deleted.

Then choose one of the following options for deleting the WebAdmin user displayed:

- Click the "Delete Object Immediately" button on the displayed property page.
- Click the "Delete WebAdmin User Immediately" entry in the "WebAdmin-User '<Name>'" rack on the sidebar.

Only WebAdmin users with the master privilege are authorized to delete a WebAdmin user.

It is not possible to delete the WebAdmin user with the name "Master". It is also not possible to delete a WebAdmin user who is currently logged on.

12.2 Hosts

"Host" WebAdmin objects describe the computers on which the UTM applications to be administered are running (the application hosts).

The boxes on the property page:

- 'Name' edit box
You can select any name for the host. However, it has to be unique within WebAdmin (to be more precise, within the current configuration database). The name can be up to 128 characters long.
- 'Name' edit box
Specify the name under which the host will be known in the network. The name can be up to 255 characters long.
- 'UTM Hostname' edit box
Specify the name under which the corresponding host will be known to UTM. This is particularly important for cluster node applications. Here, the name corresponds to the 'Hostname' value specified for the cluster node object.
- 'IP-Address' edit box
Enter the host's IP address here.
- '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
A comment relating to the operating system used can be entered here (e.g. the precise system version number). Maximum length: 255 characters. The entry is optional.
- 'Remark' edit box
A comment relating to the host can be entered here. Maximum length: 255 characters. The entry is optional.

12.2.1 Displaying the host list

To view a list of all the hosts defined in the configuration database, click "Hosts" in the "WebAdmin Configuration" rack on the sidebar. The "WebAdmin Configuration" rack does not always appear on the sidebar. You can display it however by clicking the "Home" button on the navigation bar.

12.2.2 Creating a new host

To create a new host, it is first necessary to display the host list. The "Hosts" rack is then displayed on the sidebar. Clicking on "New Host" in this rack opens the page for defining the new host.

Once you have specified the properties for the new host, click the "Create Object" button and the new host will be created.

Only WebAdmin users with the master privilege are authorized to create a host.

12.2.3 Deleting a host

To delete a host, it is first necessary to display the property page for the respective host. First display the host list, then click on the name of the host to be deleted.

Then choose one of the following options for deleting the host displayed:

- Click the "Delete Object Immediately" button on the displayed property page.
- Click the "Delete Host Immediately" entry in the "Host '<Name>'" rack on the sidebar.

Only WebAdmin users with the master privilege are authorized to delete a host.

If you delete a host then all the UTM applications defined on this host are also deleted (see section "Deleting an application" on page 27).

12.3 UTM applications

WebAdmin objects of the type "UTM application" describe the UTM applications that are to be administered with WebAdmin. A WebAdmin object of the type "Host" that describes 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 UTM application is to be administered with WebAdmin, at least one UPIC connection must also be entered for the UTM application so that WebAdmin can access the application via UPIC (see section "UPIC connections" on page 28).

The boxes on the property page:

- '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, it is advisable to specify a "genuine" UTM application name here (MAX APPLINAME). The name can be up to 128 characters long.
- 'Host' dropdown box
Specifies the computer on which the application is to run (application host). All hosts known to WebAdmin are listed in the dropdown box. The host may need to be defined first (see section "Hosts" on page 24).
- 'Cluster' text box
If the UTM application forms part of a UTM cluster and WebAdmin has already detected this, then the name used for this cluster in WebAdmin is specified here.
This field is not displayed when you generate a new application.
- 'UTM Version' dropdown box
Specifies the version of the UTM application. The dropdown box contains all versions known to WebAdmin.
If necessary, WebAdmin updates the version automatically on first successful contact with the UTM application.
- 'Version Remark' edit box
A comment relating to the version can be entered here (e.g. the precise version name).
Maximum length: 128 characters. The entry is optional.
- 'Standard Port' edit box
Specifies the port through which the UPIC connection is to be established (the default port number is 102). An entry in this field is optional.
The value specified here is the standard port for the UPIC connection. If necessary, it can be overwritten when defining a UPIC connection to this application.
The port specified here (or for the UPIC connection) must match the generation of the UTM application (see "Linking to UTM applications" on page 11).
- 'Standard Selector Format' dropdown box
Specifies the selector format for the UPIC connection. Possible formats are 'Ascii', 'Ebcdic' and 'Transdata'.
The value specified here is the standard value for the UTM application. If necessary, it can be overwritten when defining a UPIC connection to this application.
The selector format specified here (or for the UPIC connection) must match the generation of the UTM application (see "Linking to UTM applications" on page 11).
- 'Keep TA Open' dropdown box
If "Yes" is selected here, 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 respective 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 as a

result of timeouts on the database side. If "No" is selected here, the administration program unit KDCWADMI performs a PEND FI rather than a PEND KP.

- 'Remark' edit box

A comment relating to the UTM application can be entered here. Maximum length: 255 characters. The entry is optional.

12.3.1 Displaying the application list

To view a list of all the UTM applications defined in the configuration database, click "UTM Applications" in the "WebAdmin Configuration" rack on the sidebar. The "WebAdmin Configuration" rack does not always appear on the sidebar. You can display it however by clicking the "Home" button on the navigation bar.

12.3.2 Generating a new application

To generate a new UTM application, it is first necessary to display the application list. The "UTM Applications" rack is then displayed on the sidebar. Clicking on "New UTM Application" in this rack opens the page for defining the new application.

Once you have specified the properties for the new UTM application, click the "Create Object" button and the new application will be generated.

Only WebAdmin users with the master privilege are authorized to generate a UTM application.

12.3.3 Deleting an application

To delete a UTM application, it is first necessary to display the property page for the respective application. First display the application list then click on the name of the UTM application to be deleted.

Then choose one of the following options for deleting the UTM application displayed:

- Click the "Delete Object Immediately" button on the displayed property page.
- Click the "Delete UTM Application Immediately" entry in the "UTM Application '<Name>'" rack on the sidebar.

Only WebAdmin users with the master privilege are authorized to delete a UTM application.

It is not possible to delete a UTM application which is included in a currently open application collection. In this case, either first remove the application from the open collection or close the currently open collection.

When you delete an application, all the objects belonging to this application are also deleted from the configuration database. In particular, this also applies to UPIC connections and message collectors (including all stored messages) defined for this UTM application. The UTM application is also deleted from all the application collections in which it is present.

12.4 UPIC connections

WebAdmin objects of the type "UPIC connections" describe the UPIC connections required by WebAdmin to communicate with the UTM applications if they are to be administered using WebAdmin. When an application collection is opened, WebAdmin searches for a 'matching' UPIC connection to each UTM application in the collection. 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 boxes on the property page:

- 'UTM Application' text box
Specifies the name of the UTM application to which the UPIC connection is to be established.
- 'Bcamappl/Appliname' edit box
Specifies the name of the application via which the UPIC connection is to be established. This will generally be a BCAMAPPL name from the UTM generation.
- 'Pterm' edit box
If the UPIC connection is to be established via a specific PTERM name belonging to the UTM generation, the name of the respective Pterm must be specified here.
If the UPIC connection is to be established via a terminal pool (TPOOL) belonging to the UTM generation, either leave this edit box empty or leave the default value "<Tpool-Coupling>".
- 'Listener-Port' edit box
If you leave this box empty, the standard port specified for the UTM application will be used for communication.
If you wish to use a port other than the standard one, specify it here.
- 'Selector Format' dropdown box
You can select the selector format here.
You can use either the standard selector format specified for the UTM application or another format.
- 'User' edit box
If you wish to use the UPIC security function, specify here the name of the UTM user with which the UPIC connection signs on to UTM. Otherwise you can leave the box empty. User names are case-sensitive.
- 'User Password' edit box
Specify here the password of the UTM user with which the UPIC connection signs on to UTM. If the specified user does not have a password or if no UTM user was specified, leave this edit box empty.
- 'Connection Timeout (sec)' dropdown box
Specify any timeout for the UPIC connection here.
If you choose "<no timeout>", WebAdmin will wait indefinitely for the UTM application to respond.
If you choose "<seconds> (WebAdmin default)", the global timeout value set for WebAdmin will be used.
Otherwise you can also choose a value other than the WebAdmin default here.

If no response is received from the UTM Application within the specified time, WebAdmin terminates the connection and marks the respective UTM application as not available.

- 'Encryption' dropdown box

Specify here whether encryption is to be used on the UPIC connection or not. The encryption level to be used for a connection is negotiated with UTM when the connection is established.

If no encryption is available then the connection is operated without encryption even if "Yes" has been selected here.

- 'Remark' edit box

A comment relating to the UTM connection can be entered here. Maximum length: 255 characters. The entry is optional.

12.4.1 Displaying the UPIC connection list

To view a list of all the UPIC connections defined in the configuration database, click "UPIC Connections" in the "WebAdmin Configuration" rack on the sidebar. The "WebAdmin Configuration" rack does not always appear on the sidebar. You can display it however by clicking the "Home" button on the navigation bar.

12.4.2 Creating a new UPIC connection

To create a new UPIC connection, it is first necessary to display the property page of the UTM application to which the UPIC connection is to be set up. The "UTM Application '<Name>'" rack is then displayed on the sidebar. Clicking on "New UPIC Connection" in this rack opens the page for defining the new UPIC connection.

Once you have specified the properties for the new UPIC connection, click the "Create Object" button and the new UPIC connection will be created.

Only WebAdmin users with the master privilege are authorized to create a UPIC connection.

12.4.3 Deleting a UPIC connection

To delete a UPIC connection, it is first necessary to display the property page for the respective UPIC connection. First display the UPIC connection list then click on the name of the UTM application to which the UPIC connection to be deleted belongs.

Then choose one of the following options for deleting the UPIC connection displayed:

- Click the "Delete Object Immediately" button on the displayed property page.
- Click the "Delete UPIC Connection Immediately" entry in the "UPIC Connection '<Name>'" rack on the sidebar.

Only WebAdmin users with the master privilege are authorized to delete a UPIC connection.

12.5 Collections

Users can create "application collections" (or "collections" for short) of UTM applications in which applications that are to be administered jointly are combined into groups. WebAdmin stores these collections in the configuration database and can then subsequently reopen them at any time.

To administer 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 within a WebAdmin session. If you open a second collection, the first is closed automatically (although it is of course preserved and can be reopened at any stage).

WebAdmin always knows a generic collection called '<All UTM Applications>', which always contains all the UTM applications currently defined in the configuration database. It is not possible to delete or edit this collection.

The boxes on the property page:

- 'Name' edit box

You can select any name for a collection. However, it has to be unique within the configuration database. The name can be up to 64 characters long.

- 'Group' edit box

The group name entered here is used for arranging the individual collections in the collection list on the home page of the web application and on the sidebar.

It is also possible to specify several names separated with the character "/". This is similar to a file path on a Unix/Linux system.

The group name can be up to 64 characters long. The entry is optional.

- 'Remark' edit box

A comment relating to the collection can be entered here. The comment can be up to 255 characters long. The entry is optional.

12.5.1 Displaying the collection list

You can view a list of all the application collections defined in the configuration database in one of the following ways:

- Click the "Home" button on the navigation bar.
- If the "WebAdmin Configuration" rack is displayed on the sidebar, click on "Collections" here.
- If the "Collections" rack is displayed on the sidebar, click on "Collections" here.


12.5.2 Creating a new collection

To create a new application collection, it is first necessary to display the collection list. The "Collections" rack is then displayed on the sidebar. Clicking on "New Collection" in this rack opens the page for defining the new collection.

Once you have specified the properties for the new collection, click the "Create Object" button and the new collection will be created.

Only WebAdmin users with the master privilege are authorized to create an application collection.

12.5.3 Deleting a collection

To delete an application collection, it is first necessary to display the property page for the respective collection. First display the collection list then click on the  icon in the line of the collection to be deleted.

Then click the "Delete Object Immediately" button on the displayed property page and the collection will be deleted.

Only WebAdmin users with the master privilege are authorized to delete an application collection.

You cannot delete a collection that is currently open.

12.5.4 Opening a collection

You can open a collection using one of the methods below:

- Via the collection list on the Home page

The Home page is displayed after you log on to the WebAdmin application. You can also return to this page at any time by clicking the "Home" button on the navigation bar.

All the currently defined collections are displayed in a list on this page. Click on the name of one of the collections in the "Name" list column to open the collection.

- Via the "Collections" sidebar rack

The "Collections" sidebar rack contains an entry for each defined collection. Click on one of the collections to open it.

Note: The "Collections" rack does not always appear on the sidebar. You can display it however by clicking the "Home" button on the navigation bar.

If, when you open a collection, another collection is already open, then this first collection is automatically closed and the selected collection is opened instead.

After opening, the primary page of the collection is displayed. Also, if a collection is open, the corresponding button for it is displayed on the navigation bar.

12.5.5 Closing a collection

You can close the currently open collection using one of the methods below:

- Open another collection

The previously opened collection is then automatically closed.

- Log off the WebAdmin user via the button on the title bar of the web application

- Implicitly by terminating the application session

13 Object lists

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

UTM objects that can occur repeatedly in an application (e.g. User), as well as WebAdmin objects (WebAdmin actions, WebAdmin users, hosts, UTM applications, UPIC connections and collections) are displayed on list pages.

The objects are displayed in tabular form. Each row 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 systems and not for Unix/Linux or Windows systems. In such cases, WebAdmin displays the symbol "<.>", which means "not available".

13.1 Displaying object lists with WebAdmin objects

The object lists for the WebAdmin objects (WebAdmin actions, WebAdmin users, hosts, UTM applications, UPIC connections and collections) can be displayed by selecting the respective entry in the "WebAdmin Configuration" rack on the sidebar.

- 'WebAdmin Actions'
Displays the WebAdmin action list.
- 'WebAdmin Users'
Displays the WebAdmin user list.
- 'Hosts'
Displays the host list.
- 'UTM Applications'
Displays the application list.
- 'UPIC Connections'
Displays the UPIC connection list.
- 'Collections'
Displays the collection list.

The "WebAdmin Configuration" rack is displayed on the sidebar whenever you click the "Home" button on the navigation bar.

13.2 Displaying object lists with UTM objects

To view an object list with UTM objects, it is first necessary to open a collection. On the primary page of the collection choose the respective standalone UTM application or UTM cluster by clicking on the

name of the application or cluster. The primary page of the UTM application or the cluster is then displayed and the names of all the available object lists are shown in the 'Object Lists' section. Click on the name of an object list to open the corresponding object list page.

13.3 Structure of the object list page

Example of an object list page:

Name	Type	Call	Tacclass	PGWT	Api
Filter	All	All	Filter	All	All
KDCAPPL	Dialog	Both	<none>	No	KDCS
KDCAPPLA	Asynchronous	Both	16	No	KDCS
KDCBADTC	Dialog	First	<none>	No	KDCS
KDCDIAG	Dialog	Both	<none>	No	KDCS
KDCDIAGA	Asynchronous	Both	16	No	KDCS
KDCDLETQ	Queue	Both	<none>	No	KDCS
KDCHHELP	Dialog	Both	<none>	No	KDCS

Number of Tacs: 583

The title bar shows the icon for the object list followed by its name, and at the right-hand end the icon for accessing the online help system.

Below that there is a row of buttons for performing actions relating to the displayed object list.

Underneath the buttons is a dropdown box listing all the column sets available for the table. You can use this dropdown box to choose the object properties to be displayed in the table. Click the ">" button to advance to the next available column set. Click the "<" button to return to the last available column set.

The actual object list is then displayed with the corresponding column headers which you can use to sort the objects in ascending or descending order, plus the filter bar which enables simple filtering of the objects displayed.

The individual objects are listed below that, with one object in each row of the table.

13.4 Filtering the list objects

You can use the filter input fields below the column headers to filter the objects to be displayed in an object list. You can specify only one filter criterion at a time in one column. To do this, type the filter text into the filter field and press Enter.

13.5 Sorting objects

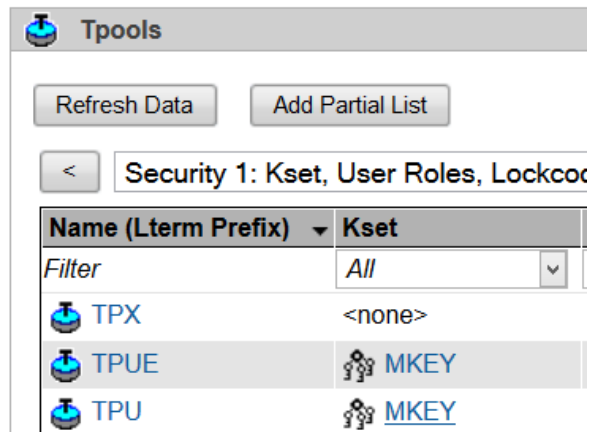
Object lists can be sorted simply and quickly according to various criteria. Simply click on the appropriate column header. The list is then sorted in ascending or descending order on the basis of this column. The column currently being used to sort the list is marked by a triangle symbol in the column header. If you click on a column header that is not marked, the table will first be sorted in its standard sort order. Clicking on a marked column reverses the sort order.

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

13.6 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). These associations are indicated by a link in the corresponding list cells in the object lists. Click on this link to view the property page of the respective linked object.

Example:



In this Tpool object list, the Tpool "TPU" is linked to the Kset "MKEY". If you hover the mouse pointer over the name "MKEY", the browser will present the name "MKEY" as a link, as shown in the screenshot. If you click on this link, the property page of the linked Kset "MKEY" will be displayed.

13.7 Object list functions

Some object lists have functions specific to them. You can call these functions either by clicking on the buttons above the list or via the object list rack on the sidebar.

The following functions are available:

- 'Refresh Data'

This function retrieves all the data in this object list again from the UTM applications or reads it in again from the configuration database. The contents of the object list are then refreshed.
- 'Create <object type>' or 'Create New Object'

This function is only available for object types that can be created dynamically (Con, Kset, Lses, Ltac, Lterm, Pterm, Program, Tac, User). It opens the property page for creating a new object of the specified type.

See also section "Creating and deleting objects" on page 42.

- 'New Host...' / 'New Application...' / 'New UPIC Connection...' / 'New WebAdmin User...'
This function is available in object lists containing WebAdmin objects. It opens the property page to create a new object of the specified type.
- Add Partial List
This function is used to create a new partial list for the object list displayed.
See section "Defining partial lists" on page 37.
- 'Reset All TAC Statistics'
This function is available for a TAC list of a UTM cluster or a standalone UTM application. It can be used to reset the values of certain counters for all the TACs in the list to zero. This refers to the following values for each TAC:
 - 'Used'
 - 'Errors' (number of program unit runs terminated with errors)
 - 'Elapsed Time (msec)' (average 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 time for database calls per program unit run)

These statistical values are displayed on the 'Current' tab of the TACs property page. You can also reset these counters for specific TACs using the 'Reset' check boxes on this tab.

- 'Reset All Wait Times'
This function is available for a TAC class list (Tacclasses) of a UTM cluster or a standalone UTM application. It can be used to reset the values of certain counters for all the TAC classes in the list to zero. This refers to the following values for each TAC class:
 - 'Average Wait Time (msec)'
 - 'Calls'
 - 'Waits'

These statistical values are displayed on the 'Current' tab of the TAC classes property page. You can also reset these counters for specific TAC classes using the 'Reset' check boxes on this tab.

13.8 Object-specific functions

In contrast to WinAdmin, in WebAdmin you cannot access the object-specific functions via the object lists.

You access them instead on the property page of the object. Click on the name of an object in the object list to view its property page.

See section "Object-specific functions" on page 40.

14 Defining partial lists

Some object lists of a UTM application may contain a large number of objects. This is often the case with the User and Lterm lists for example. As a result, it may take a very long time to fetch the data. Filtering objects is implemented in WebAdmin itself, not on the UTM side (post-selection), so it is also not possible to fetch only a certain subset of the objects by pre-selecting them in WebAdmin.

However, the objects are passed from the UTM administration interface in alphabetical order, and it is possible for the interface caller to specify the name of the first object to be transferred. This procedure is utilized in WebAdmin to request partial lists of objects from UTM. An object partial list is defined by one or more name ranges. Only the objects with names within these name ranges are requested from UTM and displayed in the respective list. For instance, it is possible to define a user partial list having the range "K-M" in which only users with names beginning with the letters K, L and M are then displayed (and fetched). By defining a partial list "BR-BR", for example, it would be possible to fetch all users with names beginning with the letters "BR".

The names returned by UTM are listed in alphabetical order. Note however that the order is determined by the underlying character code that WebAdmin uses. Since WebAdmin runs on Unix/Linux or Windows systems the ASCII code is used as the basis for sorting names.

This is the EBCDIC code for UTM applications running on BS2000 systems, and the ASCII code for UTM applications running on Unix/Linux or Windows platforms.

The three character ranges that are most important for object names are uppercase letters (A-Z), lowercase letters (a-z) and digits (0-9). Within these three ranges, the order of the characters is the same in both character codes. However, the three ranges follow in a different order in the two character codes. With ASCII, digits come before uppercase letters, and uppercase letters come before lowercase letters. With EBCDIC, on the other hand, lowercase letters come before uppercase letters, and uppercase letters come before digits.

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

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

This must be taken into account when specifying name ranges.

Example:

If you define a name range "-A", for example, in WebAdmin, then in the case of an application running on Unix/Linux or Windows systems, all objects with names beginning with A plus all objects with names beginning with digits will be returned, as the digit range comes before A in ASCII code. For an application running on BS2000 systems, all objects with names beginning with A will likewise be returned. However, in addition, all objects beginning with lowercase letters will also be returned, as lowercase letters come before A in the EBCDIC code. Objects with names beginning with digits will not however be returned as numbers come after A in the EBCDIC code.

14.1.1 Adding a partial list

In WebAdmin, partial lists are shown below their associated main list as an entry in the "Object Lists" table on the primary page of a UTM application or of a UTM cluster. If you display the main object list to which you want to add a partial list, there is a button "Add Partial List" above the table. Clicking this button takes you to a page where you can define the new partial list (e.g. "A-M"). After you click the "Add Partial List" button, WebAdmin will display the object list page for the new partial list defined.

When defining the new partial list, you can specify several name ranges, separated by commas:

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

These name ranges can be specified 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 and all characters before K in the character code.

M-: All names that start with M and all characters after M in the character code.

TEST: All names that start with TEST.

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

WebAdmin stores the definition of the partial list in the configuration database for each specific application.

14.1.2 Removing a partial list

The "Remove Partial List" button appears above a partial list being displayed. Click this button to remove, after the confirmation prompt, the partial list from the WebAdmin configuration.

15 Property pages

Property pages can be displayed for UTM objects. These property pages always contain all the properties of the corresponding objects supplied by the administration program interface and cannot be parameterized by the user. 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 the user can enter values.

The property pages contain one or more tabs which can be opened individually. Each property page has a "General" tab containing the most important properties of the object, and may have one or more additional tabs with further properties.

In addition, the property name of linked objects acts as link which enables the property page of each linked object to be called up directly.

The property pages have a 'Refresh' button. Click this button to update the contents of the property page, i.e. fetch the current values from the UTM applications again.

15.1 Displaying the property page for an object

Simply click on the name of an object displayed in a window, e.g. an object list. This name will also be displayed as a link in the browser.

Tip:

If you hold down **CTRL** while clicking on the object name link, most common browsers will open the property page of the object on a new browser tab. It is then very easy to switch back to the original tab without having to load it again. This is particularly useful when the original page is an object list with many entries.

15.2 Modifying object properties

Provided openUTM permits modification of an object's properties, these properties can be modified on the WebAdmin property page described above. To do so, first click the 'Modify Object' button on the property page. If a property can be modified, it is displayed as an edit box or as a dropdown box. You can then simply enter or select the new parameter values and click on the 'Apply' button.

In general the syntax of entries in edit boxes must comply with the rules of the KDCDEF generation tool.

Some of the modified parameters are checked locally by WebAdmin; otherwise they are checked by the UTM application(s). If a modification is rejected by WebAdmin, the values last entered in the property page screens 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.

15.3 Object-specific functions

Some objects have functions specific to them. You can call these functions either by clicking on the buttons below the property page, or via the object rack on the sidebar.

The following functions are available:

- 'Refresh' or 'Refresh Data'
This function retrieves all the data for the displayed object again from the UTM applications or reads it in again from the configuration database. The contents of the property page are then refreshed.
- 'Delete <object type>' and 'Delete <object type> Immediately' or 'Delete Object' and 'Delete Object Immediately'
This function is only available for object types that can be deleted dynamically (Con, Kset, Lses, Ltac, Lterm, Pterm, Program, Tac, User). It is used to delete the respective object from the UTM configuration.
See section "Creating and deleting objects" on page 42.
- 'Delete WebAdmin User Immediately...' / 'Delete Host Immediately...' / 'Delete UTM Application Immediately...' / 'Delete UPIC Connection Immediately...'
This function is available in object lists containing WebAdmin objects. You can use this function to delete the displayed WebAdmin object.
- 'Send Message'
This function is only offered for Lterm objects of standalone or node applications in BS2000 systems. On the page that is then displayed you can enter a message to be output on the system status bar of the respective dialog station.
- 'Spoolout'
This function is only offered for printer Lterm objects of standalone or node applications. You can establish a connection to a specific printer using this function.
- 'Update IP-Address'
This function is only available for objects of the type Con, Osi Con and Pterm. It causes openUTM to synchronize the IP address of the selected partner with the hostname database (only available for protocol type TCP-IP-APPLI or also RFC1006 on Unix/Linux or Windows systems).
- 'Confirm Printout'
This function deletes a print job awaiting confirmation. The confirmation mode of the printer must be switched on.
- 'Repeat Printout'
This function repeats a print job awaiting confirmation. The confirmation mode of the printer must be switched on.
- 'Show Messages'
The list of messages of the selected message collector is displayed.
- 'Poll Messages'
The messages of the selected message collector are polled.
- 'Read Content'

The content of the selected GSSB is read and displayed.

- 'Reset Transaction'
This function resets an open transaction with status PTC (prepare to commit).
- 'Explicitly Sign-Off User'
This function signs off a UTM cluster application user.
- 'Abort Bound User Service'
This function aborts the bound service of a UTM cluster application user.
- 'Abort Bound User PTC Service'
This function aborts the bound service with transaction in the PTC state of a UTM cluster application user.

16 Creating and deleting objects

WebAdmin allows the following dynamically administered objects to be created and deleted: 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).

In WebAdmin, it is possible to create new objects as follows:

First of all, display the page containing the object list for the required object type. There is a 'Create Object' button above this table (alternatively you can also access it from the '<object type>' entry in the table rack on the sidebar).

Click this button or rack entry to display a property page which is similar to the one described earlier for existing objects. Here you must specify the name and the parameters for the new object, then click on 'Create Object' to proceed. If there are default values for parameters, these are preset by WebAdmin. Mandatory parameters of the object which have no default value (e.g. the name of the new object) are marked by WebAdmin with the symbol "<=>".

To delete an object, first display the property page of the object to be deleted. This page has the button 'Delete Object' and/or 'Delete Object Immediately'. Alternatively, there are corresponding rack entries for the respective object on the sidebar. When you click on one of the buttons or rack entries, a confirmation prompt will appear. If you confirm, the object will be deleted and the object list to which the object list belonged will be displayed. Objects that are deleted immediately are removed from the object lists directly after their successful deletion.

17 Functions of a UTM application and UTM cluster

The functions of a UTM application or of a UTM cluster are displayed below the "Functions" section on the primary page of the application or cluster. This section is hidden by default, only the title bar is visible. Click on the title bar to make the section visible. In addition, the racks for the UTM application and the UTM cluster on the sidebar also contain options representing functions.

The following application or cluster-related functions are available:

- **'Application Properties' or 'Cluster Properties'**

Displays the property page of the UTM application or UTM cluster.

This function is only available in the rack for the UTM application or UTM cluster on the sidebar.

- **'Check Availability'**

When this function is selected, WebAdmin displays a page with the button 'Check Availability' and the result of the last availability check. If you click the button, WebAdmin checks whether the affected application(s) can be accessed via the UPIC interface. If so (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. The results of the availability check are displayed in detail below the button.

If the function is executed for a UTM cluster, all activated node applications are checked.

- **'Start'**

You use the 'Start' function 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 "Starting UTM applications" on page 90). 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 systems and for Unix/Linux systems, the name of the procedure which WebAdmin 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 it has to be a shell script that can be called using "sh <name>". For Windows systems, the name of a service is required which WebAdmin will then start with the command "net start <name>".

- User ID

Enter the user ID under which the start procedure is available on the host or the service is started. This is particularly important if the 'FTAC Admission' check box is not activated. However, even if this check box is activated, WebAdmin expects to find a user ID here. WebAdmin uses the unique data pair of host/ID to store the specified access data in the configuration database and to use it again subsequently.

- Password

If necessary, specify the password for the user ID (if you are not using a FTAC profile).

- Account or Work Directory

The account number for 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 WebAdmin transfers a job file.

- 'FTAC Admission' check box and edit box

If you check the box on the left, the specified FTAC access authorization is used for file transfer instead of the user ID and password.

If you have checked the 'FTAC Admission' box, you must enter the FTAC access authorization for the FTAC profile defined on the host here.

WebAdmin stores the entries made on this page in the configuration database and automatically proposes them the next time a call is made, with the result that no further user input is required unless there is any change in the access data. The access data is stored together with the above-mentioned unique host/user ID data pair.

The time needed to start the applications differs, depending on the system and the start procedure used. The 'OK' message after you execute the function only indicates that the start procedure has been called up on the host. You can use the 'Check Availability' option to determine whether or not the start was successful.

If this function is called for a UTM cluster, the node application to be started must be selected on the displayed page.

■ 'Shut'

The 'Shut' function is used to terminate UTM applications. After you call this function, a page is displayed which offers the following choice of shutdown modes: SHUT WARN, SHUT GRACEFUL, SHUT NORMAL and SHUT KILL. For SHUT WARN and SHUT GRACEFUL, you can also specify the interval after which the application(s) are closed plus the text of a warning message.

If this function is called for a UTM cluster, you can also specify whether the entire cluster or only one of the node applications is to be terminated.

■ 'Message Dump'

Specifies the conditions under which a message dump is generated.

(See section "Message dump" on page 93.)

This function is not available for a cluster node application.

■ 'Create Diagdump'

Creates a diagnostic dump for the application.

If this function is called for a UTM cluster, you must choose for which cluster node application it is to be executed.

■ 'Inverse KDCDEF'

The 'Inverse KDCDEF' function allows you to initiate an inverse KDCDEF run which generates files, and also library elements in the case of BS2000 systems, with KDCDEF statements for dynamic administration objects on the respective application host. A page is then displayed on which, among other things, you are asked to select the statements to be generated and specify the names of the output files or library elements.

If this function is called for a UTM cluster, you must also choose for which cluster node application it is to be executed.

■ 'Send Message'

The 'Send Message' function allows you to send a message to all active UTM dialog stations for a UTM application on BS2000 systems. A dialog box for entering the message is opened. (Note: Messages to individual dialog stations can be sent via the property page of an Lterm.)

If this function is called for a UTM cluster, you must also choose for which cluster node application it is to be executed.

■ 'Spoolout'

You can use the 'Spoolout...' function to establish connections to all printers of the UTM application(s) for which messages are present and which are not connected.

If this function is called for a UTM cluster, you must choose for which cluster node application it is to be executed.

■ 'Change Application'

The 'Change Application' function enables the application program to be changed while the application is running. On the page then displayed, you are asked whether you want to switch to the next higher or to the previous file generation.

This function is not available for a cluster node application.

■ 'Switch Logging File'

This function enables the logging file to be switched over. On the page then displayed you are offered the following choices:

- 'Uslog'
Switches the user log file(s) to the next file generation group.
- 'Syslog'
Switches the SYSLOG logging 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, can be accessed via the Syslog tab of the property page for a UTM application or UTM cluster.)
- 'Protocol Files'
Switches the application protocol files over.

This function is not available for a cluster node application.

■ 'Encryption'

This function enables you to access the security functions offered by openUTM to encrypt user data during transfer between a UPIC client and a UTM application. The 'Encryption' page of the property dialog for the respective UTM application(s) is displayed.

The displayed property page:

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

This function is not available for a cluster node application.

■ 'Update IP-Addresses'

This function can be used to synchronize the IP addresses stored in the object tables of UTM applications with the IP addresses in the hostname database. The addresses of all the affected communication partners of types Pterm, Con and Osi-Con are updated in the applications.

This function is not available for a cluster node application.

■ 'Show Open Transactions in PTC State'

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

If this function is called for a UTM cluster, you must first choose the node application for which it is to be executed.

■ **'Online Import'**

This function is only available for a UTM cluster or a cluster node application.

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

On the page then displayed, you must select the non-running node application ('source node application') whose data is to be imported into this node application.

If this function is called for a UTM cluster, you must also choose the node application that is to import the data.

■ **'Explicitly Sign-Off Users'**

This function is only available for a UTM cluster or a cluster node application.

This function can only be executed for a node application that WebAdmin does not consider to be currently running. It explicitly signs off from UTM any UTM users who are still signed on to the non-running node application. This function is executed by the UTM node application currently designated "Master" by WebAdmin.

If this function is called for a UTM cluster, you must choose the respective node application.

■ **'Abort Bound Services'**

This function is only available for a UTM cluster or a cluster node application.

This function can only be executed for a node application that WebAdmin does not consider to be currently running. You can find a detailed description of this function in the UTM Administration Manual (KDCADMI operation code KC_LOCK_MGMT, subopcode KC_ABORT_ALL_BOUND_SERVICES). This function is executed by the UTM node application currently designated "Master" by WebAdmin.

If this function is called for a UTM cluster, you must choose the respective node application.

■ **'Repair User File Lock'**

This function is only available for a UTM cluster.

This function allows you to cancel the lock on the UTM user file if this is still set as a result of an aborted KDCDEF run.

■ **'Host'**

This function is only available for a standalone UTM application or a cluster node application.

It displays the property page of the UTM application host,

■ **'Current UPIC Connection'**

This function is only available for a standalone application or a cluster node application.

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

■ **'Remove From Collection'**

Removes the UTM application or UTM cluster from the currently open collection.

This function is only available for a cluster node application.

■ **'Modify Cluster Name'**

This function is only available for a UTM cluster.

It allows you to change the name of a cluster as displayed in WebAdmin.

■ **'Activate/Deactivate Node Application'**

These functions are only available for a deactivated or activated cluster node application respectively. The functions are executed immediately.

18 Support for the UTM cluster concept

UTM supports the creation of clusters which consist of multiple similar UTM applications (node applications). This section describes how WebAdmin supports the UTM cluster concept.

Note:

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

18.1 UTM clusters and node applications in the web user interface

The UTM clusters in the opened collection are displayed in the 'UTM Clusters' section on the primary page of the collection, and the standalone applications are listed in the 'Stand-Alone UTM Applications' section.

Click on the name of a UTM cluster in the 'UTM Clusters' section to open the primary page for this cluster. The page structure is similar to that of the primary page for a standalone application and has three sections:

The 'Functions' section is hidden by default, only the title bar is visible. Click on the title bar to show the functions available for this cluster. These functions are also listed below 'Functions' in the Clusters rack on the sidebar.

The 'Node Applications' section is only available on the primary page of a UTM cluster and contains one entry for each node application of the cluster in each case.

The 'Object Lists' section contains all the object lists available in the cluster.

The functions and object lists displayed on this primary page of the cluster represent a global view of the cluster. For a local view of the node applications of the cluster, there is a similarly structured primary page for each node application which is displayed when you click on the name of a node application in the 'Node Applications' section.

The following basic rule applies:

Functions executed on the primary page of the cluster are global cluster functions, while those executed on the primary page of a node application are local application (node-local) actions. The same applies to the data displayed: Data displayed on the primary page of the cluster (and in the associated lists) is global cluster data, whereas data that is displayed on the primary page of a node application (and in the associated lists) is node-local data.

Example:

The 'Shut' function on the primary page of the cluster enables the entire cluster to be terminated, whereas the same function on the primary page of a node application terminates only this one node application.

18.2 Defining a cluster in WebAdmin

No explicit definition of a cluster object is required in WebAdmin. You declare a cluster node application in WebAdmin in the usual manner (host object, application object, UPIC connection). The

application is then initially listed as a standalone application in an open collection. The first time that contact is successfully established between WebAdmin and the application (e.g. by explicitly checking availability), WebAdmin identifies that the application is a cluster node application. The application is then removed from the standalone section and displayed in the UTM Clusters section instead. WebAdmin 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 WebAdmin objects of the other node applications belonging to the cluster (i.e. hosts, applications and UPIC connections). These objects are automatically written to the configuration database and are therefore already present when the next program session begins. In addition, all the cluster node applications are automatically transferred into the currently open collection and are displayed in the 'Node Applications' section of the cluster's primary page.

It is therefore advisable to explicitly define the WebAdmin objects (host object, application object, UPIC connection) of **only one** cluster node application. The respective WebAdmin objects of the other cluster node applications are then automatically generated by WebAdmin as described above.

WebAdmin 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 are running) 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 WebAdmin and do not have to be explicitly added by the user.

18.2.1 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 to communicate with WebAdmin. Otherwise an attempt to establish a connection from WebAdmin to one of the node applications could be rejected because the respective UPIC user that WebAdmin wants to use to sign on to UTM is still signed on to 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 long. <nn> is a two-digit number in the range between '01' and the total number of generated cluster nodes.

Example:

If six cluster nodes have been generated, then the following six USERS could be generated for communication with WebAdmin:

```
WEBADM01  
WEBADM02  
WEBADM03  
WEBADM04  
WEBADM05  
WEBADM06
```

When contact has been successfully established with a node application belonging to a cluster, WebAdmin checks whether the UPIC user specified for the UPIC connection to the node application complies with the above name convention. In particular, it checks the number indicated in the last two characters of the name. If this part of the name matches the index of the respective node application, when importing the other node applications WebAdmin assigns the UPIC users of these connections in accordance with the convention.

The index of the cluster node can be determined indirectly from the KDCDEF input used for cluster


generation. The first cluster node specified ('CLUSTER-NODE' statement) has the index 1, the second has the index 2, etc.

Example :

If WebAdmin successfully establishes contact with the node application with index 3 and if the UPIC user for the connection has the name 'WEBADM03', then WebAdmin assigns the UPIC users 'WEBADM01', 'WEBADM02', 'WEBADM04' etc. for the remaining node applications to be imported.

As an alternative to this automated method of assigning UPIC users, they can of course also be modified manually after the node applications have been imported by WebAdmin. To do this, choose the 'Current UPIC Connection' function of a node application and change the UPIC user accordingly.

18.3 Deactivating cluster nodes

Cluster nodes that have been generated for future use in UTM, but for which the corresponding applications are not (yet) currently being run, can be deactivated in WebAdmin. This is done by executing the 'Deactivate Node Application' function for a node application. Any node application deactivated using this function is indicated by a special icon () in the user interface. The primary page of a node application that has been deactivated in this way contains only the 'Activate Node Application' function. It is therefore not possible to access the objects and properties of these node applications. WebAdmin does not take account of deactivated node applications for cluster functions (check availability, fetch objects, etc.).

Deactivated node applications can be reactivated using the 'Activate Node Application' function.

18.4 Functions of a UTM cluster

UTM cluster functions are also listed in the 'Functions' section of the cluster's primary page, and also below 'Functions' in the Clusters rack on the sidebar.

The individual functions are described in detail in section "Functions of a UTM application and UTM cluster" on page 43.

18.5 Functions of a node application in the UTM cluster

The functions of a node application are listed in the 'Functions' section of its primary page, and also below 'Functions' in the Node Application rack on the sidebar.

The individual functions are described in detail in section "Functions of a UTM application and UTM cluster" on page 43.

18.6 Property page for a UTM cluster and node application

The property page for a UTM cluster and for a node application has additional tabs named 'Cluster' and 'Cluster Current' which contain the general and the current properties of the UTM cluster respectively.

The following values can be changed on the 'Cluster' tab for a UTM cluster on this property page (this is not possible on a node application's property page because changes there apply globally to the clusters):

- 'Alive Check Timer (sec)'
- 'Communication Retry'
- 'Reply Timer (sec)'
- 'Restart Timer (sec)'
- 'File Lock Timer (sec)'
- 'File Lock Retry'
- 'Deadlock Prevention'

18.7 List of UTM cluster nodes

UTM clusters and node applications have a further object list with the name 'Cluster Nodes' which contains the properties of the individual cluster nodes belonging to the UTM cluster. Like all other object lists, this list can be accessed via the 'Object Lists' section on the primary page of the UTM cluster.

It is not possible to modify any values on the property page of a cluster node displayed via a local, node-level list, whereas the following (global cluster) values can be edited on the property page of a cluster node displayed via the global cluster-level list:

- '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!

18.8 Object lists of a cluster

The available object lists of a cluster are shown in the 'Object Lists' section of the primary page for a UTM cluster. Click on the name of an object list to display the list of objects of that type. 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 property pages 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 are

generated in the same way. However, it should be noted that WebAdmin is not able to retrieve data from all the cluster applications simultaneously and that this type of <mix> status may consequently arise in the case of values that can change dynamically. 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 lists are also listed in the 'Object Lists' section on the primary page of a node application. Here, however, they simply refer to the relevant node application (local node-level lists).

19 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 mark 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).

19.1 Accessing UTM services – keys and locks

Basically, when "users" use the services 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 by particular users, both services and couplers can have locks in UTM. Users can thus only access a 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 for 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.

19.1.1 "Services"

The services of a UTM application are represented by objects of the 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 statements).

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

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).

19.1.2 "Users"

Users are represented by UTM objects of the 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 Kset of the user (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.

19.1.3 "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 creates 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 that 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 creates a user. This implicitly generated user "inherits" the Lterm's Kset.

If the coupler is a Tpool, openUTM implicitly creates 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 established, 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 creates 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 creates 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, the specified user is assigned to the Osi-Lpap instead.

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).

19.2 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 possible), 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 hand, at the time of UTM generation it is necessary to decide which services particular users are to be permitted to access. However, it is frequently easier to view things from the opposite direction: which users should be allowed to access this service?

The "new" access control concept takes this as its starting point.

This concept introduces the idea of "roles". It is possible to assign one or more roles to each user (e.g. "manager", "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 permitted to access the service. For example, the access list for the "Payroll" service might contain the roles "manager" and "payroll clerk" since all department managers and payroll clerks are to be allowed to use this service, whereas caretakers are not.

This concept is largely based on 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 Lac object types have been extended to include 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, it is advisable not to assign any lockcodes to couplers. Although this means that all the couplers are then 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.

This concept is supported by WebAdmin as described in the following sections.

19.3 Roles

In WebAdmin roles are represented as "normal" UTM objects, for example as Ksets. However, the role objects are known only to WebAdmin 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 less transparent 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 "department manager" or "clerk", and can be edited as required. Wherever roles are displayed in WebAdmin, for example in object lists and property pages, the alias of the role is specified instead of the name.

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-relevant 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)

Object Type	Type	Generation parameter	Role type
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

19.3.1 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 listed separated by commas. If no role object is assigned to a lockcode/keycode then the role is represented by its printable code instead.

Example:

An access list contains codes 1, 2 and 5. Code 1 is the role with the alias "Manager" and code 5 is 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 "Manager,Clerk,2".

19.3.2 Roles in property pages

The property dialogs for objects that may reference Ksets or access lists have (at least) one further dialog page 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 page also includes the LOCK parameter so that all the parameters relating to access rights are grouped together 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 are displayed in the form of a list of comma-separated roles or codes in the same way as in object lists.

Tpool and Osi-Lpap dialogs both contain two user role pages (for Kset and User-Kset or Ass-Kset). Tac dialogs contain two access role pages (for read and write access to the queue) for Tac queues

(TYPE=Q). In user dialogs there are three role pages: one for the user roles and two with the access roles for read and write access to the user queue.

20 WebAdmin message destination – message collectors

openUTM has so-called "service-controlled queues" (see also section "Service-controlled queues in WebAdmin" on page 72). 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 were sent to the queue. As well as the ability to create and use such queues on a temporary basis (during a program unit run), it is also possible to use static Tac and user queues.

From the point of view of WebAdmin, an interesting aspect of the Tac and user queues is that they are also permitted message destinations for UTM messages. This queue property has been used to make WebAdmin into a kind of "message destination" for UTM messages. WebAdmin is able to fetch messages from the addressed UTM queues either cyclically or at the request of the user, output these in lists, and if necessary save them in the configuration database.

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

20.1 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 then be retrieved by WebAdmin. This requires the following steps:

- Define messages with the message destination USER-DEST-1...4
Specify the (additional) message destination USER-DEST-1 (or USER-DEST-2...4) for all messages that WebAdmin 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
```

20.2 Actions on the WebAdmin side

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

There is also an entry "Message Collectors" in the 'Object Lists' section on the primary page of a standalone application or of a cluster node application. Click on this entry to view the list of currently defined message collectors. This object list page has a 'Create New Object' button which you can click to create a new message collector.

20.3 Properties of message collectors

The property page for message collectors contains the following controls:

- 'UTM Queue Object' section

This specifies 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 only be used as queues if the generated QLEV is greater than zero. 'Queue Name' specifies the name of the UTM object (i.e. either the name of a Tac or a user queue).
- 'Message Polling' dropdown box

Specifies when WebAdmin is to fetch the messages from openUTM. Here you can either set 'On Demand' or choose a defined regular interval at which WebAdmin automatically fetches the messages. You can choose intervals between 30 seconds and two hours.

If you choose 'On Demand' then there are two different ways of initiating message retrieval. You can either click the 'Poll Messages' button on the property page of the message collector, or you can display the message list ('Show Messages' button) and click the "Poll Messages" button on the object list page which is then displayed with the messages.
- 'Message Type' dropdown box

Specifies the type of message that WebAdmin can expect in the queue.

 - 'UTM message'

This is a message generated by openUTM in PRINT format.

- 'Printable Text'
The messages can also be printed but they might 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, WebAdmin converts the data at the UTM end of the connection into a printable, hexadecimal format and displays the data in the message lists in this format.
- 'Message Lifetime' dropdown box
This property specifies how long WebAdmin keeps the collected messages.
 - 'Don't store'
The collected messages are not stored in the configuration database. If you close the current collection then the messages are lost forever.
 - 'Delete on Demand'
This is the opposite of 'Don't store'. In this case, the messages are stored in the configuration database and are not deleted until this is explicitly requested. You can do this by choosing the 'Delete Message Immediately' function in the rack of a collected message on the sidebar.
 - 'Keep Last <n>'
In this case, the last 100, 500, 1000 or 5000 collected messages are stored in the database. If the database contains more messages than the number specified then the oldest messages are deleted.
 - 'Keep 1 Day / Keep 2 Days / Keep 1 Week / Keep 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 WebAdmin fetches the messages, not the time the messages were generated in openUTM.
 - 'Current Application Run'
Only the messages from the current application run of the respective UTM application are stored in the database. The application start time is taken from the structure `kc_curr_par_str` of the KDCADMI interface (fields `start_date_year`, `start_date_month`, `start_date_day`, `start_time_hour`, `start_time_min`, `start_time_sec`). This is the (cold) start time of the UTM application (system time of the application host).
If the message is of the type 'UTM message', the time the message was sent by UTM (system time of the application host) is taken as the message time. Otherwise the time at which WebAdmin fetched the message from UTM (system time of the WebAdmin host) is taken as the message time.
- 'Consume Message' dropdown box
Here you can specify whether the messages are to be immediately consumed on reading (this is the default) or not. If the messages are not to be consumed, they will remain in the respective queue. This makes it possible, for instance, to view the contents of messages in the dead letter queue.

20.4 Message collectors in the cluster

Message collectors are also local node objects in a UTM cluster. WebAdmin ensures that the message collectors are "synchronized" in a UTM cluster. That is to say, if a message collector is created, modified or deleted in a node application, then this action is automatically performed in all activated node applications so that the same message collectors with the same settings are present in all activated node applications.

The list of message collectors can be accessed via the 'Object Lists' section on the primary page of a node application.

20.5 The collected messages list

You can activate the collected messages lists via the associated message collectors. Click the 'Show Messages' button on the property page for a message collector. The object list then displayed contains all the messages for this message collector that are stored in the database and/or have been fetched by WebAdmin during the current session.

The columns have the following meaning:

Column	Meaning
Sequence No.	Sequential number of the message. Each message collector has 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 the sequence has not been changed via message queue administration (DADM)).
Polled at	Time when WebAdmin fetched 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).
Send at	Specifies the time at which a UTM message was generated in UTM (system time of the application host) 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 here. If a message consists of multiple message segments then the individual segments are displayed in sequence here (separated by ":-").

20.6 Tips and notes on generation

If WebAdmin is to be used as a "message destination", you should make sure that the UTM application is able to meet the resulting resource requirements. You must set suitable generation 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 a queue via the QLEV parameter. This prevents a page pool overload. It is also advisable 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 specified queue level is reached.

- WebAdmin 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 communication 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 WebAdmin as a message destination for messages generated by the program units themselves. However, 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 WebAdmin. 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 WebAdmin and openUTM, you should remember that the messages that WebAdmin fetches from the UTM queues actually disappear from the queues if the default value 'Consume Message=Yes' is set for the message collector. WebAdmin should therefore only poll queues with 'Consume Message=Yes' that are intended precisely for this purpose, i.e. whose messages have been sent to the queue specifically for WebAdmin. If any other queues are polled by WebAdmin then this is sure to create havoc with the logic of the UTM application since important messages will suddenly disappear.

Additional note:

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

21 Transaction-oriented administration jobs

In UTM, administration jobs are executed at different times. There are three fundamentally different execution times:

- Immediately (**I**)

Some jobs are executed immediately within the administration program unit and are not subject to transaction management (deferred delivery). These jobs are referred to as type **I** in the Administration Manual. The I stands for "immediate".

- At the end of the transaction (**P**)

Other jobs only take effect once the transaction within which the administration program unit was called has finished. As long as the transaction has not been completed it can still be reset, with the result that the administration job is then not executed. This type is referred to as **P** in the Administration Manual. The P stands for "PEND" (program end).

- Independently (**A**)

With this type, UTM initially only creates a job for executing the action, which is then executed by UTM at a subsequent point in time. Administration jobs of this type are not subject to transaction management but are executed by UTM separately from the administration program unit. This type is referred to as **A** in the Administration Manual. The A stands for "announced" or also "Action".

The openUTM manual "Administering Applications" indicates which of these three types each administration job is. In the case of some objects for which several properties can be modified, the individual changes may come under different types. For example, changing the status of a Tac takes immediate effect (**I**), but changing the lockcode is transaction-dependent and belongs to the second type (**P**).

It is also possible to find out about the modification characteristics of individual object properties in WebAdmin as well. They are also displayed with the tooltips on property pages (see section "Tooltips on property pages" on page 97). You can also view these tooltips on property pages by hovering the mouse pointer over the name of an object property.

22 Administration of message queues

WebAdmin 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 WebAdmin graphic user interface.

You can:

- Display information about asynchronous jobs that are buffered in a message queue (including any user information).
WebAdmin requests the user information with a maximum length of 1024 bytes from the UTM applications.
- 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 the asynchronous jobs for service-controlled queues
Service-controlled queues are temporary queues, Tac queues and User queues (see also section "Service-controlled queues in WebAdmin" on page 74).
WebAdmin requests the messages with a maximum length of 1024 bytes from the UTM applications.

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

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

The properties dialog for these objects in the WebAdmin administration section contains the button 'Display Message Queue'. If you click this button, WebAdmin displays all the asynchronous jobs in the message queue for the object in a new object list, the message queue list. It should be noted that in UTM clusters, this button is only present in object properties dialogs 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.
Job Submitter	UTM user who initiated the asynchronous job.

Column	Meaning
Job Submitter Type	The Job Submitter of the asynchronous job can be a user, a LU6.1 session or a OSI-TP association.
Generation Time	Time at which the job was submitted in the form "nnn / hh:mm:ss", where nn 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 message destination.
Positive Confirmation Job	Specifies whether there is a positive confirmation job for the asynchronous job.
Negative Confirmation 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 WebAdmin user. 'incomplete': The user information has been requested and read but has not been retrieved in full because of its length. WebAdmin requests the user information with a maximum length of 1024 bytes from the UTM applications. 'complete': The user information has been requested and read and is available in full.
User Information	User information for the asynchronous job with a maximum length of 1024 bytes. The following outputs are possible: '<not requested>': The user information has not yet been requested by the WebAdmin 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 about the status of the message for an asynchronous job in a service-controlled queue. The following outputs are possible: 'not requested':

Column	Meaning
	<p>The message has not yet been requested by the WebAdmin user.</p> <p>'incomplete': The message has been requested and read but has not been retrieved in full because of its length. WebAdmin requests the message with a maximum length of 1024 bytes from the UTM applications.</p> <p>'complete': The message has been requested and read and is available in full.</p>
Message	<p>Message for asynchronous job in a service-controlled queue with a maximum length of 1024 bytes.</p> <p>The following outputs are possible:</p> <p>'<not requested>': The message has not yet been requested by the WebAdmin user.</p> <p>other values: Specify the requested and read message either in printable or hexadecimal, printable format.</p>
Redelivery Counter	<p>Counter indicating how often the message has already been delivered. Redelivery Counter is only displayed for service-controlled message queues provided that the message has already been read.</p>
Data Read at...	<p>Time when the data of the asynchronous job was retrieved from the UTM application (system time of the WebAdmin computer).</p>

The following actions can be performed for the list of message queue entries:

- 'Request All User Info'
Retrieves the user information or the messages corresponding to the individual message queue entries from UTM.
- 'Delete All Jobs'
Deletes all the jobs in the message queue.

The following additional action can be performed for the list of message queue entries in a service-controlled queue:

- 'Request All Messages'
Retrieves the messages corresponding to the individual message queue entries from UTM.

The following additional action can be performed for the list of message queue entries in the dead letter queue:

- 'Move All Jobs'
Moves all the jobs in the dead letter queue to the corresponding original destination or to another destination.

Click on the job ID of a job in the list of message queue entries to display the job's property page. This property page contains buttons for performing the following actions:

■ 'Prioritize Job'

This command is available if the job is not already at the head of the message queue (position 1). You use this action 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 job from the message queue.

■ 'Move Job'

This command is only available for the dead letter queue.

Moves the job from the dead letter queue to the original destination or to another destination. You can specify the destination in the dialog box that is then displayed.

The destinations can be asynchronous Tacs or Tac queues with the exception of the Tacs "KDCDLETQ" and "KDCMSGTC". Also Lpaps or Osi-Lpaps can be selected. But the new destination must be of the same type as the original destination.

You can use the 'Printable' and 'Hexa Printable' radio buttons to define the way in which the user information or messages are displayed in a job's property dialog.

23 Administering printers and print jobs

WebAdmin allows you to administer the printers of a UTM application together with the associated print jobs. The entire functionality of the PADM call of the KDCS program interface is available via the user-friendly WebAdmin user interface.

You can:

- display information about all printers assigned to a printer control Lterm
- lock and release printers again
- establish and terminate a 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 print jobs to be confirmed
- confirm or repeat print jobs

23.1 Printer control Lterms

To make it easy to identify printer control Lterms (Cterms), the Lterm object lists now contain the column 'Printer Control Lterm' which can have the values 'Yes' or 'No'.

23.2 Administering printers

WebAdmin provides a list of all printers for UTM applications. You can access this list via the administration section of the navigation tree (in the same way as other UTM object lists).

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 confirmation mode is active or inactive for this printer.

Column	Meaning
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 property dialog also to display any queued print jobs (for which there is as yet no confirmation) for this printer. You can call up this dialog directly from the printer list by 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 fetched from the UTM application (as in all administration object lists).

Click on a printer ID in the printer list to view the property page for this printer where you can activate or deactivate confirmation mode for the printer by clicking the 'Modify Object' button.

Note that if you deactivate confirmation mode, you will need to individually confirm any print jobs still in the list of print jobs to be confirmed. Otherwise new print jobs for the printer will not be processed.

23.3 Administering print jobs

WebAdmin also provides a list of the print jobs of UTM applications requiring confirmation. This list can likewise be accessed via an entry in the 'Object Lists' section on the primary page of a standalone application or a cluster node application.

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 host).
Start Time	Desired start time for printing in the case of time-controlled jobs (DPUTs) in the

Column	Meaning
	same format (system time of the application host).
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 fetched from the UTM application (as in all administration object lists).

Click on a Job ID in this object list to view the property page of the print job to be confirmed. You can either confirm or repeat the print job to be confirmed by clicking the 'Confirm Printout' and 'Repeat Printout' buttons on this page respectively.

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 assigned to the printer (not the printer control Lterm of the printer!).

24 Service-controlled queues in WebAdmin

UTM makes use of so-called "service-controlled queues" (frequently referred to as "queues" for short). The respective openUTM manuals provide a detailed description of precisely what a service-controlled queue is, where it can be used and how it can be used.

There are three types of service-controlled queue. These differ primarily in the way they are created, but there is generally no difference in their function or mode of use. The three types of queue 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 therefore represented by a special variant of an UTM object of the type TAC.
- User-Queues
Every UTM user simultaneously represents a potentially exploitable user queue unless the QLEV property is explicitly set to zero. A "User" type object is created either with a KDCDEF "USER" statement or dynamically by means of a KDCADMI program interface call.

WebAdmin also uses service-controlled queues for the "WebAdmin message destination" functionality (see section "WebAdmin message destination – message collectors" on page 59).

The sections below describe how WebAdmin supports service-controlled queues and indicates where you will encounter them.

24.1 The queue list

There is an entry called 'Temporary Queues' in the 'Object Lists' section on the primary page of a standalone application or a cluster node application. Click on this entry to view an object list containing all the temporary queues for the respective UTM application. The following properties are listed:

Column	Meaning
Name	Name of the service-controlled queue.
Queue Level	Queue level of the queue. Specifies the maximum number of messages that may be present in the queue. Queue Level 32767 means that the number of messages is unlimited.
Queue Length	Number of messages present in the queue at the time the data was fetched from the UTM application.

Column	Meaning
Queue Mode	Queue mode, i.e. either 'Standard' or 'Wrap-around'. Standard means that no new messages are accepted once 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.
Data Read at...	Specifies the time at which the displayed data was fetched from the UTM application (as in all administration object lists).

24.2 Creating and deleting queues

You create Tac queues in the same way as "normal" Tacs. In the case of Tac queues, you must set the 'Type' property to 'Queue'. Every object of the type User implicitly has its own user queue. In both cases, you must of course pay attention to the queue-specific properties 'Queue Level' and 'Queue Mode'. In addition, you can restrict access to the queue you want to create on the 'Queue Read Access' and 'Queue Write Access' dialog pages.

WebAdmin is also able to create temporary queues. Click on the 'Create New Object' button above the object list containing the temporary queues.

All queues can be deleted again using the 'Delete Object' button on the respective property page (provided that the object in question is currently available for deletion).

25 Statistics

WebAdmin offers functions for collecting statistical data on the UTM applications. The collected data can be displayed in a graphic or in a list, copied to the clipboard, and then presented in Microsoft Excel, for example, or it can be saved to a (text) file.

The collected data is automatically stored in the configuration database.

So-called "statistic collectors" are responsible for collecting the statistics 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). This makes it possible, for example, to monitor the "In-Queue" of a specific Tac from within WebAdmin.

In addition, it is possible to define an upper and lower threshold for each statistic collector which causes WebAdmin to trigger a 'threshold alert' if exceeded.

From WebAdmin's perspective, the statistic collectors are objects just like any other UTM objects. They are stored in the configuration database. They are therefore available to other WebAdmin instances that work with the same database.

After a collection has been opened, defined and activated statistic collectors automatically start to collect values provided that the initial contact with the UTM application has been successful.


25.1 Statistics lists








The collection of statistics data is controlled via the statistics collector list. The object list of a UTM application or UTM cluster therefore contain the entry 'Statistic Collectors'.

25.1.1 Collector list

This list can be handled in exactly the same way as a UTM object list (see section "Object lists" on page 33).

The list contains the following columns:

Column	Meaning
1st column without title	Selection of up to three statistic collectors for the graphical display. You select a collector for graphical display by checking the corresponding box. Once the list has been refreshed, the collector list can be sorted on the selected collectors. Instead of a filter, the number of currently selected collectors is displayed.
Type	Icon and type of the statistic collector, as well as a specification of the unit of measurement (# for number, % for percent). Possible icons:
	 Collector is inactive. It can be activated via its property dialog.

Column	Meaning
	 <p>Collector sleeping (see 'State' column).</p>
	 <p>Collector is active.</p>
	 <p>Collector is active and is currently requesting a statistics value.</p>
	 <p>Active collector in which the last collected statistics value exceeded the defined upper threshold.</p>
	 <p>Active collector in which the last collected statistics value exceeded the defined lower threshold.</p>
	 <p>Active collector whose assigned "collecting" object does not exist (no longer exists) in the corresponding UTM application.</p>
	 <p>Active collector in which the last attempt to collect a statistics value failed. Possible cause: The statistics value that is to be collected is not available in the operating system version or variant of the corresponding UTM application.</p>
Object Type Object Name	Type and name of the "collecting" object that is assigned to the statistic collector.
Host	Name of the host on which the UTM application to which the collector belongs is running. This is not displayed for standalone applications.
Activated	Specifies whether the collector should collect statistic values.
State	<p>State of the statistic collector Possible values:</p> <p>Inactive The collector is 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 has been activated but no statistic values are being collected at present because the application has not yet been contacted or is not (no longer) available.</p> <p>Without object The "collecting" object assigned to the collector is not (no longer) available in the UTM application.</p>
Interval	Specifies the period between two pollings of statistic values.
Automatic Wake-Up	Specifies whether the sleeping collector should be woken up and start collecting data when the UTM application is available.

Column	Meaning
	Caution: If automatic wake-up is selected for one or more selectors for a UTM application then WebAdmin checks the availability of this UTM application at regular intervals.
Statistic Values Lifetime	<p>Limits the volume of collected statistic values stored in the configuration database.</p> <p>Don't Save The collected values are not saved in the configuration database and are lost when the current collection is closed.</p> <p>Delete on Demand All the collected values are saved in the configuration database without restriction. They are only deleted if the user explicitly requests this.</p> <p>Keep Last <n> The collected values are saved in the database up to the selected maximum number. If this number is exceeded then the oldest values exceeding the permitted quantity are deleted.</p> <p>Keep 1 day/2 days/1 week/1 month The values are stored in the database for the specified period and are then deleted when this period expires.</p> <p>Current Application Run Only the values from the current application run of the respective UTM application are stored in the database. The start time of the application is taken from the structure kc_curr_par_str in the KDCADMI interface.</p>
Last Value Minimum Maximum	Indicates the last value or the smallest or largest value so far identified in the current WebAdmin session. '<.>' is displayed for each of these values until a successful polling operation has taken place.
Next Sample At	Specifies the time at which the statistic value will next be polled. '<.>' is displayed here until a polling operation has taken place.
Number of All Values	Specifies the number of all available values.
Values	Specifies the number of statistic values for the collector polled in the current WebAdmin session.
Upper Threshold	Upper threshold set for the collector or '<none>' if no upper threshold is defined.
Lower Threshold	Lower threshold set for the collector or '<none>' if no lower threshold is defined.
Upper threshold exceeded	Specifies whether the last polled value exceeds any upper threshold that may have been defined.
Lower threshold exceeded	Specifies whether the last polled value exceeds any lower threshold

Column	Meaning
	that may have been defined.
threshold exceeded	Specifies whether the last polled value exceeds any lower or upper threshold that may have been defined.
Upper Threshold Alert	Type of alert issued if the upper threshold is exceeded.
Lower Threshold Alert	Type of alert issued if the lower threshold is exceeded.
Remark	Comment concerning the statistic collector.

25.1.2 Statistics chart

Caution:

The graphical output of statistics values in Microsoft Internet Explorer is only possible as of version 10.

WebAdmin can display up to three statistic collectors in a chart. To do this, proceed as follows:

- In the property dialog of a statistic collector, choose the action "Show Chart" to display the values of the collector in a chart.
- Using the checkboxes in the first column, select up to three statistic collectors from the list of statistic collectors and then choose the action "Show Chart" in order to display the values for these collectors in a chart.

Before displaying the chart, you can also define the following parameters:

Parameter	Meaning
Time Range	Choose a time range from 1 hour through to 1 year or "All" for the statistic values that are to be displayed. Caution: The graphical display of thousands of values can take a few seconds and may possibly cause the termination of Java Script routines in the browser. If this is the case then you should choose a smaller time range.
Show multiple value axis	This option is only available if you are displaying multiple collectors in the chart. It permits the optimum scaling of the curves of collectors whose values have different units, e.g. load in percent compared to CPU time in microseconds.
Poll statistic values	This option is of use if a running UTM application or UTM cluster application is to be monitored.

Now choose the action "Refresh Data" to display the chart.

Each polled statistic value appears as a node in the graphic; neighboring nodes are connected by a line. The x-axis (horizontal) of the graph corresponds to the time, while the y-axis (vertical) corresponds to the scaled values of the statistic collector.

Below the chart you will see a key describing the curves displayed in the chart.

If the curves of more than one statistic collector are displayed in the graph then the individual curves are displayed in different colors (red, green and blue in the sequence in which the collectors were selected) and the nodes in the graph are also displayed differently. The key below the graph indicates which curves correspond to which statistic collectors.

If a threshold for a collector displayed in the chart is exceeded then the icon corresponding to the collector in question above the chart is displayed with a red or blue bar.

Zooming in the chart:

- Zooming out:
 - Right-click the magnifier icon at the top right of the chart to display the entire time range.
 - With the left mouse button held down, drag the right or left slider in the gray bar above the chart to enlarge the displayed section
- Zooming in / displaying a section of the chart
 - Left-click in the chart, keep the mouse button held down and select a section that will then be displayed.
 - With the left mouse button held down, drag the right or left slider in the gray bar above the chart to reduce the displayed section

Moving the displayed time range:

- Left-click in the light area between the two sliders in the gray bar above the chart, keep the button held down and move the mouse to the right or left.
- Briefly right-click in the gray bar above the chart outside the light area between the two sliders in order to select the midpoint of the displayed time range.

25.1.3 List of statistics values for a statistic collector

WebAdmin can output all the statistics values that are stored for a statistic collector in the configuration database in a list. This list can be handled in exactly the same way as a UTM object list (see section "Object lists" on page 33).

The list contains the following columns:

Column	Meaning
Sequence No.	Sequential number of the statistics value.
Value	Individual collected statistics value. The unit of measurement (number or "%" for percent) depends on the type of statistic collector.
Polled at	Time (time and date) at which the statistics value was collected.

You can use the mouse to select entries in the list and then copy them to the clipboard.

Caution: The list must be refreshed manually.

25.2 Statistic collectors

25.2.1 Creating a statistic collector

A statistic collector can be created in the same way as other UTM objects, i.e. by clicking the "Create New Object" button in the object list or by means of the sidebar function "Create Statistic Collector".

25.2.2 Deleting a statistic collector

A statistic collector (like other UTM objects) can be deleted via its property dialog ("Delete Object Immediately" button) or via the sidebar function "Delete Statistic Collector Immediately".

When you delete a collector, all the statistic values belonging to this collector are also deleted from the configuration database.

25.2.3 Statistic collectors in clusters

In a UTM cluster, statistic collectors are also objects that are local to the node. WebAdmin ensures that the statistic collectors are "synchronized" in a UTM cluster. That is to say, if a statistic collector is created, modified or deleted in a node application, then this action is automatically performed in all activated node applications so that the same statistic collectors with the same settings are present in all 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. This list can be used, for example, to display the values of a given statistic collector for 3 node applications in a chart.

25.2.4 Functions for selected statistic collectors

All these functions can be performed simultaneously for all in statistic collectors table selected collectors.

Function	Meaning
Activate Deactivate	Activates or deactivates the selected collectors.
Automatic Wake Up via action "Modify Properties"	If automatic wake-up is selected for one or more selectors for a UTM application then WebAdmin checks the availability of this UTM application at regular intervals and may collect data for all active collectors.
Show Values	Shows all statistic values collected by the selected collectors in a window. A simple click selects these values. Then they can be copied in HTML format to the clipboard.
Save Values	Saves all statistic values collected by the selected collectors as text to a file.
Delete Values	Deletes the statistic values of the selected collectors from the configuration database. Various delete options are offered.

Function	Meaning
Set Sample Interval via action "Modify Properties"	Allows the sample interval of the selected collectors to be set.
Set Statistic Values Lifetime via action "Modify Properties"	Allows the statistic values lifetime of the selected collectors to be set.

25.3 Threshold Alert

On the 'Threshold' page of the property dialog, it is possible to assign each individual statistic collector an upper and a lower threshold. The thresholds defined in this way are also defined in the list of statistic collectors.

If these thresholds are exceeded, WebAdmin triggers a threshold alert.

This takes the following form:

- Collectors whose last collected value is above the upper threshold value are indicated by this



- Collectors whose last collected value is below the lower threshold value are indicated by this icon:



If an alert has been assigned to the threshold in question (this is also done on the 'Threshold' page of the property dialog), then the specified action is executed by WebAdmin **the first time** the upper or lower threshold is exceeded.

Any action that has already been defined can be selected for the alert. For a description of the actions that can be defined and how to define them, see section "Actions" on page 95.

If an action of type 'Call Script' has been defined as threshold alert then WebAdmin passes the following arguments to the specified script:

- /T:<Time>
 <Time>: The time at which the threshold alert occurred in the format: YYYYMMDDhhmmss.
 YYYY: Year (always 4 digits)
 MM: Month (01 to 12)
 DD: Day in 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 WebAdmin.
- /H:<Host>
 <Application>: Name of the host as defined in WebAdmin.

- /O:<Object type>
<Object type>: Type of "collecting" UTM object. 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. In the case of the object type "APPLICATION-PARAMETER", <Object name> is empty.
- /S:<Statistics type>
<Statistics type>: Type of statistics value.
In the case of the type "Connected Users (#)", this is, for example, "CONNECTEDUSERS".
The values can be taken from the table in section "Supported statistics types" on page 82.
- /V:<Value>
<Value>: Collected value that triggered the threshold alert. In the case of a percentage value, the percentage sign is also specified.

The arguments are passed in the order in which they are given above.

25.4 Saving, deleting, exporting statistics values

The values collected by the statistic collectors are automatically saved in the configuration database for the defined period (lifetime). They are then also available in later WebAdmin sessions and for other WebAdmin instances. Values present in the database can be deleted via the property dialog of the associated collector or via the table of statistic collectors. When you select the action 'Delete Statistics Values...', you see a dialog box in which you can define the deletion mode. You can choose between the following modes:

- Delete all values
- Keep newest <n> values
- Delete values older than <...>

It is also possible to specify that the values should be backed up to a (text) file before being deleted from the database.

The collected values of the selected statistic collectors can either be displayed and copied to the clipboard or backed up to a file. The values are copied in HTML format or saved in text format. The statistics values copied or saved in this way can then be inserted as plain text or loaded in other applications (e.g. Microsoft Excel).

25.5 Supported statistics types

The table below contains all the statistics types supported by WebAdmin, the object type of the associated "collecting" object and the "source field" from which the information was obtained. Unless indicated to the contrary, the name of the structure and the field name within the structure at the administration program interface are specified for the source field.

For the precise meaning, please consult the relevant UTM manual.

The "Type" column lists both the name that is output in the WebAdmin user interface and the value that is passed as the argument "/S" to an executing procedure in the event of a threshold alert.

Type	Object Type	source
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

Type	Object Type	source
Open Asynchronous Services (#) OPENASYNSERVICES	Application	kc_curr_par_str open_asyn_services
Dialog Transactions / sec (#) DIALTAPERSEC	Application	kc_curr_par_str dial_ta_per_100sec
Asynchronous Transactions / sec (#) ASYNTAPERSEC	Application	kc_curr_par_str asyn_ta_per_100sec
Dialogsteps / 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. Dialogsteps / sec (#) DIALSTEPPERSEC	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
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

Type	Object Type	source
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
Page 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 (#) USLOGFGG	Application	kc_curr_par_str 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
Cache Requests (#) NRCACHERQS	Application	kc_curr_par_str nr_cache_rqs
Cache Searches (#) NRCACHESEARCHES	Application	kc_curr_par_str nr_cache_searches
Resource Requests (#) NRRESRQS	Application	kc_curr_par_str nr_res_rqs
Resource Requests For Max. (#) NRRESRQSFORMAX	Application	kc_curr_par_str nr_res_rqs_for_max
System Resource Requests (#) NRSYSRESRQS	Application	kc_curr_par_str nr_sys_res_rqs
System Resource Requests For Max. (#) NRSYSRESRQSFORMAX	Application	kc_curr_par_str nr_sys_res_rqs_for_max

Type	Object Type	source
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: Letters (total) (#) CON_LETTERS	Con	kc_con_str letters
Con: Conbad (#) 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
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

Type	Object Type	source
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 (#) NRFPMPAGES	Pagepool	kc_pagepool_str fpmmpages
Pages for Asyn Messages (#) NRFPUTPAGES	Pagepool	kc_pagepool_str fput_pages
Pages for MSGTAC Messages (#) NRMSGTACPAGES	Pagepool	kc_pagepool_str msgtac_pages
LPUT Pages (#) NRLPUTPAGES	Pagepool	kc_pagepool_str lput_pages
Pages for Physical Output Messages (#)	Pagepool	kc_pagepool_str

Type	Object Type	source
NRPHYSMSGPAGES		phys_msg_pages
Pages for OSI-TP Log Records (#) NRLOGRECPAGES	Pagepool	kc_pagepool_str log_rec_pages
Pages for Reset Messages (#) NRRESETMSGPAGES	Pagepool	kc_pagepool_str reset_msg_pages
Other Pages (#) NROTHERPAGES	Pagepool	kc_pagepool_str other_pages
Pterm: Conbad (#) 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: Avg. CPU Time (msec) (#) TAC_TACCPUMSEC	Tac	kc_tac_str taccpu_msec
Tac: Avg. 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: Tac Units (#) TAC_USED	Tac	kc_tac_str used
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: Number of Transaction Commits (#) TAC_NBR_TACOMMITS	Tac	kc_tac_str nbr_ta_commits

Type	Object Type	source
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
Tacclass: Calls (#) TACCLASS_NRCALLS	Tacclass	kc_tacclass_str nr_calls
Tacclass: Waits (#) TACCLASS_NRWAITS	Tacclass	kc_tacclass_str nr_waits
Tpool: 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

26 Security - users and authorizations

Access to WebAdmin and the type of actions that can be executed are controlled by a user and authorization concept. In WebAdmin, various users can therefore be defined and passwords and privileges assigned. WebAdmin users must not be confused with UTM users which are used for communication with UTM applications.

26.1 Passwords and overall privileges

Passwords and overall privileges are assigned to users on the WebAdmin User property page. This is described in detail in section "WebAdmin User" on page 23.

If a password has been assigned to a WebAdmin user, then the user can only log on to the WebAdmin application if they enter the correct password on the login page. When defining a new user, a user with the master privilege may define a password for this user directly. It is possible to change the password of the current user at any time, or to delete it by entering a blank password, on the property page for this WebAdmin user.

The overall privileges that can be assigned to a WebAdmin user on the associated property page cover the so-called master privilege as well as read and write authorizations. "Overall" means that any authorization assigned here applies to all UTM applications and their associated objects.

Only users with write privileges can make administrative changes to openUTM applications via WebAdmin (create or delete objects, modify objects, establish connections, etc.). Users without this privilege can only view the data and objects of UTM applications. If a user does not have write privileges, the write actions are deactivated in the menus and are therefore unavailable.

The following overall privileges exist:

- Read authorization for administration
A user with this privilege may view but not modify the administration data of all UTM applications and UTM 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 UTM objects.
- Master privilege
A user with this privilege has all the above privileges and can also create, modify or delete hosts, UTM applications, UPIC connections, Collections and WebAdmin users in the configuration database.

27 Other functions

27.1 Starting UTM applications

It is also possible to use WebAdmin to start UTM applications. The prerequisite for this is the file transfer product openFT (Version 10 or higher).

To be able to use the file transfer functions, it is necessary to make certain settings in WebAdmin. These are specified in the file *openft_config.cmd* or *openft_config.sh* which can be found in the directory *<WebAdmin-InstallationDirectory>/UtmWebAdmin/app/config*.

If a WinAdmin installation is already present on the same computer on which WebAdmin is installed and if this WinAdmin configuration is set up to use openFT, you can also copy the file *openft_config.cmd* or *openft_config.sh* respectively from the WinAdmin directory *<WinAdmin-InstallationDirectory>/config* into the above WebAdmin directory.

Otherwise you will need to open the file in a text editor and edit it accordingly. WebAdmin basically uses the Java interface of openFT. For this WebAdmin needs to know the location of the two JAR libraries *openFTapi.jar* and *openFT.jar* of openFT in the file system.

The following entries must be specified:

- **OPENFT_JAVA_PATH**
Specify the full path of the *java* subdirectory in the openFT installation directory.
- **OPENFT_API_JAR_PATH**
Specify the full path to the *openFTapi.jar* openFT file.
This file is usually located in the *java* subdirectory of the openFT installation directory.
- **OPENFT_JAR_PATH**
Specify the full path to the *openFT.jar* openFT file.
This file is usually located in the *java* subdirectory of the openFT installation directory.

openFT must be installed both on the WebAdmin host and on the server running the UTM application. In addition, openFT must be configured on both sides in such a way that the partner computer is known on the opposite side. For further details, please refer to the openFT documentation.

WebAdmin 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, a start procedure is initiated. For this to be possible, openFT must allow follow-up processing on all servers. The 'Inbound follow-up processing allowed' option must be activated in the admission set for this.

The dummy files cannot be deleted on servers with Unix/Linux or Windows systems. WebAdmin generates the following files in the work directories listed below:

WinAdminStartApplication<n> (Unix/Linux systems) or
WinAdminStartApplication<n>.cmd (Windows systems)

where <n> is the ID number of the corresponding UTM application.

WebAdmin may address the file transfer 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 web pages contain a 'Use FTAC profile' check box 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 this variant supports access with a user ID and 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. Flexible grading of the access rights linked to an FTAC profile is also possible.

The following operating system-dependent syntax rules and restrictions apply to the access data:

System	Parameter	Description
BS2000	User ID	1 - 8 alphanumeric characters WebAdmin automatically converts these to uppercase. Specified without leading '\$'.
	Account	0 - 8 alphanumeric characters WebAdmin 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 systems then BS2000 system automatically converts it to lowercase , e.g. XXX becomes xxx. Similarly, WebAdmin also converts a transfer admission entered in the form <admission> to lowercase.
Unix	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

System	Parameter	Description
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

27.2 Command mode

WebAdmin offers a simple command mode in order to call Tacs in UTM applications directly. When you activate command mode, you will see a Page 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 two ways:

- Click on 'Command Mode' in the rack "UTM Application '<Name>'" in the sidebar.
- Click on 'Command Mode' below the "Functions" section on the primary page of the UTM application.

The command mode page contains the following fields:

- 'Convert to Upper Case' check box

If you check this box then WebAdmin 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 have sent data successfully in command mode, this option is automatically stored as an application-specific entry in the configuration database and is re-used the next time you activate command mode.
- 'Automatic KDC Prefix' check box

If this box is checked, WebAdmin 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 Upper Case' option, the action is not performed if the first character is a period.

When you have sent data successfully in command mode, this option is automatically stored as a user-specific entry in the configuration database and is re-used the next time you activate command mode.
- 'Always Delete Output' check box

If this box is checked, WebAdmin clears the output text window before any new output.

When you have sent data successfully in command mode, this option is automatically stored as a user-specific entry in the configuration database and is re-used the next time you activate command mode.

- 'Saved Commands' dropdown box

The commands that have been used the most are stored in the dropdown box and can be selected if required. The commands that are located at the top of the dropdown box are the ones that have been used most frequently. When you have sent data successfully in command mode, these commands are also stored WebAdmin user-specific in the configuration database together with their use counters and are loaded again when you next activate command mode.

The commands that have been send using the 'Send Last Output Line' button will not be considered here.
- 'Command' edit box

You enter the command that is to be sent to UTM in the edit box.
- '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.
- '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, WebAdmin uses the last line in the output text window as a command and sends this to UTM.
- 'Delete Output' button

A click on this button clears the output text window.
- 'Delete Saved Commands' button

A click on this button clears all most frequently used commands of the WebAdmin user stored in the configuration database .
- Output text window

The entered commands and UTM responses are displayed in this 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. WebAdmin aborts multi-step transactions after the first program unit run.

27.3 Message dump

OpenUTM offers a message dump interface which is also supported by WebAdmin. There is therefore a tab on the property page of a UTM application or cluster which can be used to activate and control this function. This tab can be displayed via the 'Message Dump' function in the 'Functions' section on the primary page of a UTM application or UTM cluster.

The interface allows you to specify up to three different events which will trigger UTM to perform a diagnostic dump. The following events can be monitored:

- Occurrence of a particular UTM message

In this case up to three different message inserts can be defined which the function compares with predetermined values.

- Occurrence of a particular sign-on return code
- Occurrence of a particular RCCC code
- Occurrence of a particular RCDC code

The tab has the following input fields:

- 'Event'
Combo box offering a choice of three possible dump events.
The dump event selected here is displayed in the following input fields.
- 'Event Type'
The type of event to be defined is specified here. The following types are available:
 - '<none>'
This value indicates that the dump event is not defined and no further input is possible on the dialog page.
 - 'Message'
A specific UTM message that is to trigger a diagnostic dump is defined.
If this is selected, up to three message inserts can be defined in the bottom section.
 - 'Sign-Code'
A specific sign-on return code that is to trigger a diagnostic dump is defined.
 - 'RCCC-Code'
A specific RCCC code that is to trigger a diagnostic dump is defined.
 - 'RCDC-Code'
A specific RCDC code that is to trigger a diagnostic dump is defined.
- 'Event'
This field defines the event itself, i.e. either a specific UTM message ID (e.g. 'K065'), a sign-on return code (e.g. 'U04'), an RCCC code or an RCDC code.
You can enter a maximum of four characters.
- 'Insert 1', 'Insert 2' and 'Insert 3'
If the event type is 'Message', you can define here up to three different message inserts with which the generation of a diagnostic dump can be controlled more precisely when the specified UTM message occurs. The inserts must be specified in the format [<Index>] <Comparison> <Value> where the three parts signify the following:
 - <Index>
This specifies the index of the respective message insert. The first insert has the index 1. The highest index that can be used is 20.
 - <Comparison>
This specifies whether the insert must contain the value entered (equality) or not (inequality).
The values 'EQ', '==' or '=' are permitted for equality, and 'NE', '!=', '<>' or '><' are permitted for inequality.
 - <Value>
This is the value of the message insert to be monitored.

You can enter as insert value either a number, a printable value or a hexadecimal value in the form X'<hexvalue>'. If a number entered is not to be interpreted as a number but as a printable value, then it must be entered in the form C'<value>'. Values may be a maximum of 32 characters long, and hexadecimal values can be up to 64 characters long (two characters correspond to one byte).

If nothing is entered for an insert, then this insert is not defined.

27.4 Actions

WebAdmin allows you to define and call so-called "actions". The properties of defined actions are stored in the configuration database and are therefore available to all the WebAdmin instances that work with this database.

On the one hand, actions draw attention to a statistics threshold alert (see section "Threshold Alert" on page 80). On the other, they permit scheduled saves of the object properties of UTM applications in files on the WebAdmin server.

The following types of action exist:

- **Save Object Properties in File**
See section "Saving object properties in a file" on page 96.
- **Call Script**
A specified script, i.e. a text file containing commands which generally have the suffix '.cmd' or '.bat' on Windows systems or '.sh' on Unix/Linux systems, is called on the WebAdmin server.
- **Execute Program**
A specified program is executed on the WebAdmin server.

Administration and activities based on actions are initiated by means of the sidebar.

You work with the action list in the same way as with other object lists (see also section "Object lists" on page 33).

The property dialogs for actions contain the following functions:

Command	Description
Delete Action / Object Immediately	The action is deleted.
Test Action / Object	The action is tested, i.e. an attempt is made to execute the action. The result of this attempt is then displayed in text form in a dialog box. The text is usually output in English.
Modify Object	The modifiable properties of the action can be modified.

In the case of actions that refer to files, it should be noted that the specified file must be present at the same location on all WebAdmin server computers that use the same configuration database if it is to be possible for the action to be executed without error in all cases.

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 long.
- 'Action' text box
Specifies the type of action.
- Remark
You can enter a comment of your choice here. The comment can be up to 128 characters long.
- 'File' edit box
This box is not present for the type "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.

27.4.1 Saving object properties in a file

Actions of type "Save Object Properties in File" make it possible to save the content of a given object list in the form of a text file on the WebAdmin server. The format of the text file has been chosen in such a way 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.

The application parameters of UTM applications can also be saved to 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".

WebAdmin can also save object properties automatically at scheduled times. The interval can be set to discrete values between one and 96 hours.

In addition to the above fields, a property dialog of this type also contains the following fields:

- 'Application' edit box
The UTM application to which the objects belong.
- 'Object Type' dropdown box
Type of objects to be saved.
- 'Name Prefix'
Name prefix of the file to be created.
The following file is generated when this action is executed:

"<Destination-directory>\<Name-prefix>.<Date>.<Time>.<Action-ID>.txt".

Here, <Date> has the format "YYYYMMDD" and <Time> has the format "hhmm". <Action-ID> is a number that specifies the internal ID of the action.

Example: "C:\WebAdmin\statistics\WadmUserListe.20110519.1005.23.txt"

■ 'Automatic Save' section

In this section, you can specify whether and at what intervals the object properties are to be saved automatically by WebAdmin.

- '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 WebAdmin 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, WebAdmin 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.

For security reasons, the destination directory on the WebAdmin server is predefined.

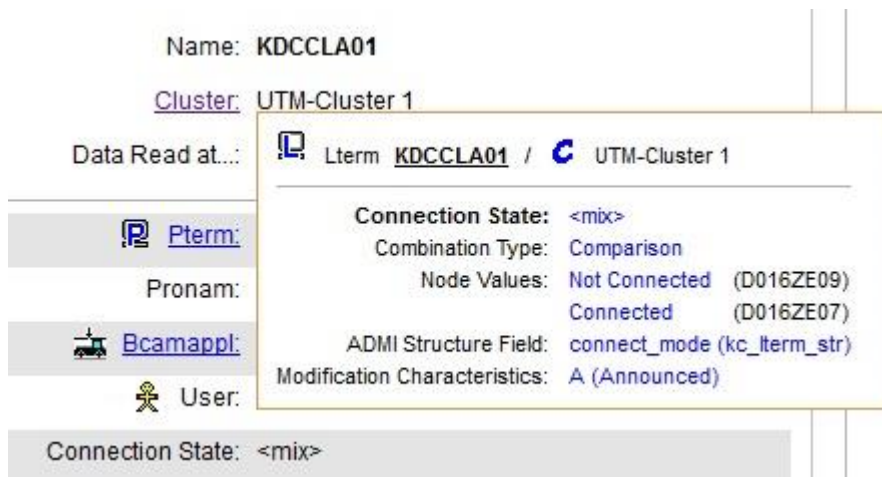
It should be noted that WebAdmin must of course be started if the action is to be executed at the desired time. In addition, the relevant UTM application must also of course be available for WebAdmin at the required time.

27.5 Tooltips on property pages

Tooltips that provide more information about the displayed object properties can pop up on the property pages of UTM objects. This is particularly helpful for global cluster property pages because the property values of the individual node applications are also shown in the tooltips.

Since WebAdmin combines the property values of the individual activated node applications into one value on the global cluster property pages, this can result in a "<mix>" status, depending on the property in question. The individual values of the node applications are also shown in the tooltip for such a "<mix>" value, so that the individual values making up the value can be immediately seen. As a result, properties that have been combined by adding together or averaging can be interpreted more easily. Besides the individual values of the node applications, the "Combination Type" is also output which indicates how WebAdmin has combined the individual values.

Here is an example of a tooltip for an Lterm object property:



Explanation:

- Header

The tooltip header shows the type and name of the respective object (including the object's icon) plus the name of the respective cluster (in this case) or UTM application to which the object belongs.

- Property name and value

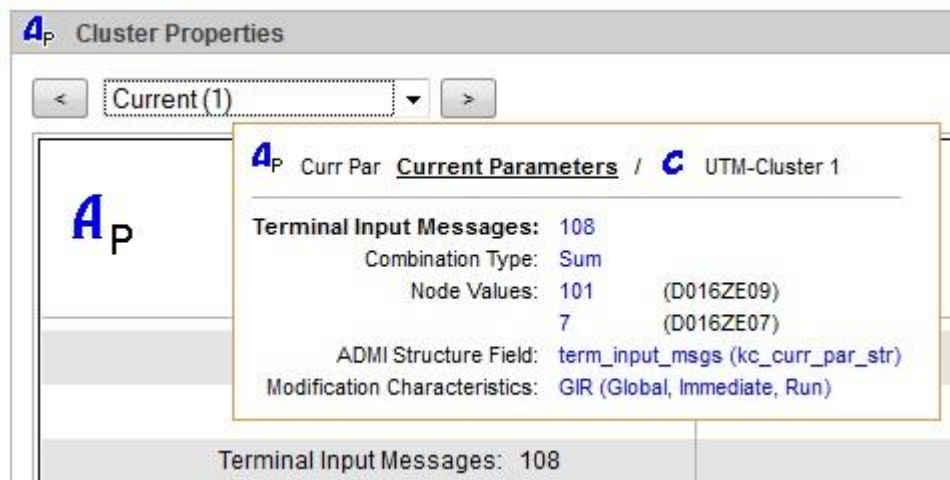
The name of the property being shown on the tooltip is in bold (in this case "Connection State") and next to it its value (in this case "<mix>").

- Combination type and node values

The combination type "Comparison" indicates that the value "<mix>" has been determined by comparing the underlying property values of the node applications. If the values are not the same (as in this case), the individual node values are displayed. The host name of the respective cluster node is shown in brackets following the node values.

The individual node values are displayed if the combination type is not "Comparison" or if the combined value is "<mix>".

Here is an example of the combination type "Sum" from the property page of a UTM cluster application.



- **ADMI Structure Field**

If the property was read from a data structure of the UTM administration interface KDCADMI (this is the case for most properties), then the name of the field within the structure followed by the structure name in brackets is shown next to "ADMI Structure Field". If necessary, this information allows you to look up the exact meaning of the values in the UTM Administration Manual.

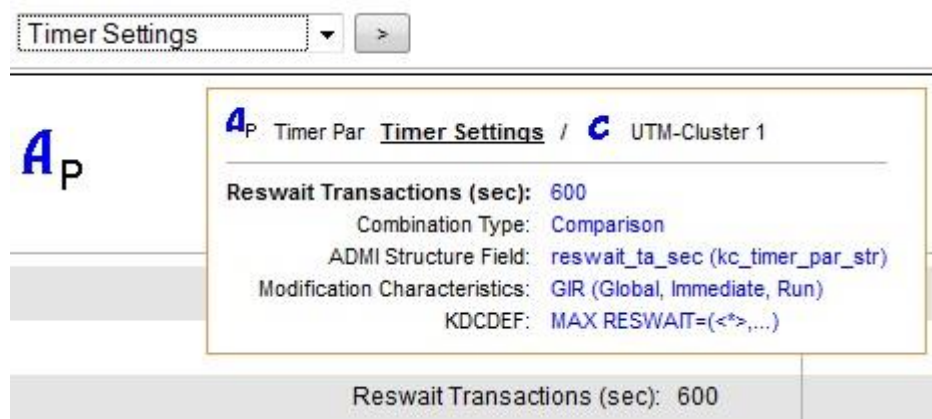
- **Modification Characteristics**

If the object property can be modified via the UTM administration interface, then how and when a change takes effect in UTM is indicated next to "Modification Characteristics". You can likewise look up the meaning of the modification characteristics in the UTM Administration Manual.

- **KDCDEF**

If the object property relates to a KDCDEF generation statement, the name of the generation statement and, if possible, the name of the respective parameter, is shown next to "KDCDEF".

Here is an example of a tooltip from the property page of a UTM cluster application:



Note: In this example, the combination type is also "Comparison", but the value is not "<mix>". The individual values of the node applications are therefore all the same and are not shown individually.

You can view tooltips on the property pages of UTM objects by hovering the mouse pointer over the name of a property.

27.6 Refreshing data

WebAdmin works with local data. When required, the requested data is polled once from the relevant UTM applications and is stored locally on the WebAdmin computer. During further processing, WebAdmin 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 can be updated by clicking the 'Refresh' button on a property page. On object list pages, click the 'Refresh Data' button.

27.7 Online help

You can call up the online help system at any time by clicking 'Help' at the top of the web pages. There is a question mark icon at the far right of the title bars of the object lists and property pages. Click on this icon to view a context-sensitive help page.

This is usually opened on a new tab in the web browser.

For more detailed information about administration and generation, see the openUTM manuals "Administering Applications" and "Generating Applications".

27.8 Trace function

For diagnostic purposes, WebAdmin allows trace information on the program run to be recorded in files. WebAdmin uses the Java Logging Framework Log4j for the logging functions. The configuration settings of the Log4j framework are stored in the file *log4j.properties.xml* which is located in the *UtmWebAdmin/app/config* subdirectory of the WebAdmin installation directory.

WebAdmin uses a mechanism that generates multiple logging files for storing the trace records. The main logging file, the number of backup files and the maximum size of a logging file are specified.

Mode of operation:

The mechanism generates and requires the two management files *<File>.lck* and *<File>.gen*, i.e. the file path will usually be *UtmWebAdmin/app/logs/logging.txt.lck* and *UtmWebAdmin/app/logs/logging.txt.gen*.

It always writes to the logging file *<File>*. On switching the file *<File>* is copied into a backup file called *<FilePrefix>.<NextFileGeneration>.<FileSuffix>*, and the file *<File>* is now written from the beginning again. If there is subsequently more backup files than specified in *<MaxNbrBackupFiles>*, and if *<MaxNbrBackupFiles>* is greater than 0, then the oldest backup files are deleted on switchover.

Example for *<MaxNbrBackupFiles>=3*

Situation before switching:

<i>logging.txt</i>	(in use)
<i>logging.13.txt</i>	(oldest backup file)
<i>logging.14.txt</i>	
<i>logging.15.txt</i>	(newest 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 switching:

<i>logging.txt</i>	(in use)
<i>logging.14.txt</i>	(oldest backup file)
<i>logging.15.txt</i>	
<i>logging.16.txt</i>	(newest backup file, previously <i>logging.txt</i>)

In this case, the <FilePrefix> is "logging" and the <FileSuffix> is "txt". A file suffix always consists of three characters and is separated from the file prefix by a dot.

WebAdmin is supplied with three different logging configurations. These are in the directory <WebAdmin-InstallationDirectory>/UtmWebAdmin/app/config and are named *log4j.properties_default.xml*, *log4j.properties_error.xml* and *log4j.properties_debug.xml*. The configuration being used is in the file *log4j.properties.xml*. To use a different logging configuration, simply delete the file *log4j.properties.xml*, copy the desired configuration file and rename it *log4j.properties.xml*. Only do this when the web server is not running.

The individual configuration files in detail:

■ ***log4j.properties_default.xml***

Following installation, the contents of this file is the same as that of the file *log4j.properties.xml*. This logging configuration records the least trace information.

■ ***log4j.properties_errort.xml***

This logging configuration records a medium amount of trace information and can be activated to diagnose errors in the web application.

■ ***log4j.properties_debug.xml***

This logging configuration records a large amount of trace information. Among other things, it activates the logging of all UPIC traces which are received from the UTM applications. It should only be activated in certain situations for error diagnostics.

Please note:

WebAdmin as a Windows service writes logging into directory apache-tomcat\logs

27.9 Tomcat logging

The version of Apache Tomcat shipped with WebAdmin also offers logging with log4j as a trace function.

The Tomcat log4j configuration file "lib/log4j.properties" in the Tomcat main directory(CATALINA_HOME) contains the complete file path (File), the maximum size (MaxFile Size) and number (MaxBackupIndex) of log files for each individual RollingFileAppender and additional a ConsoleAppender.

The log files and console output file are located in the WebAdmin main data directory under "UtmWebAdmin/logs":

UtmWebAdmin	(in use)
UtmWebAdmin.N	(Backup file, N natural number)
catalina	(in use)
catalina.N	(Backup file, N natural number)
localhost	(in use)
localhost.N	(Backup file, N natural number)
catalina.out	(Console log file)

27.10 WebAdmin Audit Logging

This functionality can be used to log WebAdmin user log-in and log-out and modifications of the WebAdmin configuration to a file.

To switch WebAdmin audit logging on proceed as follows:

1. Shut down web server.
2. Change the value of the property "allow-audit-logging" from „false“ to „true“ in file `webadmin.properties.xml` in the WebAdmin main data directory under „UtmWebAdmin/app/config“.
3. Start the web server.

You can find the WebAdmin audit logging file „audit-logging.txt“ in the WebAdmin main data directory under „UtmWebAdmin/app/audit-logging“.

28 WebAdmin as SE Manager add-on

This section brings together all the special characteristics that must be considered when running WebAdmin as an add-on in SE Manager (SEM for short).

28.1 Installation

The WebAdmin add-on is installed via the SE Manager user interface. During the installation, an `openutm` group and `openutm` ID are set up if they do not already exist on the system. WebAdmin therefore always runs under the ID `openutm`. The add-on is also logged in at the SE Manager. To permit web access, a Catalina Tomcat service – named `openutm` – is installed and automatically started on Linux systems.

When the WebAdmin add-on is uninstalled, the service is terminated and uninstalled. The add-on is also logged out at the SE Manager. The configuration data, and in particular the configuration database, are retained. This makes it easier when upgrading to a new version of the WebAdmin add-on.

28.2 File paths

WebAdmin keeps selected files which contain the data of the web application in accordance with the Linux Filesystem Hierarchy Standard in a main directory which is separate from the code:
`/var/opt/fujitsu/utmwebadmin/UtmWebAdmin/`

- `app/db`
replaces `<WebAdmin-installation-directory>/UtmWebAdmin/app/db` and contains the WebAdmin configuration database
- `app/config`
replaces `<WebAdmin-installation-directory>/UtmWebAdmin/app/config` and contains the WebAdmin-specific configuration file for logging with `log4j`
- `app/logs`
replaces `<WebAdmin-installation-directory>/UtmWebAdmin/app/logs` and contains the WebAdmin `log4j` log files
- `logs`
replaces `<WebAdmin-installation-directory>/UtmWebAdmin/logs` and contains the Tomcat `log4j` log files

28.3 Starting the WebAdmin add-on

Tomcat and the WebAdmin Web application are started automatically on the completion of installation.

28.4 Logging in at the WebAdmin add-on

The WebAdmin add-on is accessed using a web browser by means of a link in the SEM's web GUI. This implicitly logs the user on at the WebAdmin add-on with the ID (account) of the SEM user. If this user ID does not yet exist in the WebAdmin add-on then it is created and saved in the configuration database. When this is done, the privilege of the new WebAdmin user ID is derived from the roles of the SEM ID as follows:

- SEM Administrator role: User ID gets the WebAdmin Master privilege
- SEM UTM Administrator role: User ID gets the WebAdmin Write privilege
- Other: User ID gets the WebAdmin Read privilege

If the SEM user has a real name (account name) then this is also displayed in the WebAdmin add-on:

- As the currently logged-in user in the title bar
- As the "Real Name" in the list of WebAdmin users and associated property dialogs. If a user is logged in multiple times then multiple real names may also be displayed.

Otherwise, the SEM user ID is displayed in the title bar and the real name remains empty.

In the case of an implicit login to the WebAdmin add-on, the language (German or English) of the SEM session is taken over and is displayed in the title bar. This can be modified for the duration of the WebAdmin session.

Following an implicit login, the WebAdmin add-on displays the list of collections to the user.

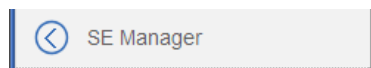
An earlier WebAdmin session is not continued!

28.5 Logging out from the WebAdmin add-on

On logout via the WebAdmin title bar, the user is logged out and the SEM login page is displayed.

28.6 Continuing the SEM session

To continue the SEM session with the last page before the WebAdmin add-on was called, it is necessary to click the topmost link "SE Manager":



This implicitly logs the user out at the WebAdmin add-on.

The SE Manager sidebar link is always visible in the WebAdmin user interface when the sidebar is expanded.

28.7 Session timeout

If the user of the WebAdmin add-on does not perform any actions requiring a new web page to be loaded or a WebAdmin action to be performed for longer than the set session timeout then, for

security reasons, this user is logged out of the WebAdmin add-on and SEM when loading the next web page or executing a WebAdmin action. The timed out WebAdmin session is also not continued when the user logs in again!

User actions in which no new web page is loaded and no WebAdmin action is executed include, for example, the sorting or filtering of lists.

The automatic refreshing of statistical charts does not count as a user action.

If automatic refreshes are deactivated for the user in the SE Manager then this user can monitor the statistic collector charts for as long as desired even without performing any user action. Once the SE Manager session timeout has expired, the user is not logged out until he or she calls a new web page.

All users can set the session timeout value and the refresh cycle individually in the SE Manager.

28.8 Lifetime of messages and statistics values

In order to restrict the data volume of collected messages and statistics values in the configuration database, only limited values can be set for the message lifetime of message collectors and the lifetime of the statistic values collected by statistic collectors, namely:

- for message collectors:
 - Do not store
 - Keep last 100
 - Keep last 500
- Additionally for statistic collectors:
 - Keep last 1000
 - Keep last 5000
 - Keep for 1 day

28.9 Conduct of dialogs

The special characteristics relating to the conduct of dialogs when WebAdmin is used as an SEM add-on are summarized again here:

- No log-in dialog but, instead an implicit login with the SEM ID and possibly other user names. The language is taken over from the SEM session. A list of all the collections is displayed. No previous WebAdmin session is continued.
- SEM user name (not the user ID!) is displayed as the real name in the title bar and in the list of WebAdmin users.
- Return to SEM via the link "SE-Manager" in the sidebar.
- Logging out from the WebAdmin add-on also results in a logout from SEM.
- Session timeout has the same effect as a logout.

- Restricted data lifetime for message and statistic collectors
- The function to start a UTM application is not available because openFT is not installed on the server.

28.10 SE management units

The SE Manager and the associated add-ons are installed separately for each management unit of the SE Server. The management units do not have any shared disk storage. As a result, the WebAdmin add-on works at each management unit with a separate local configuration database. No data synchronization is performed. It is therefore advisable to always call the WebAdmin add-on from the same management unit.

The name of the management unit on which the WebAdmin add-on is running is displayed in the WebAdmin title bar.

28.11 Backing up and restoring the configuration data

The WebAdmin add-on uses the SE Manager's **C**onfiguration **S**ave and **R**estore (CSR) functionality.

A configuration backup can be executed or restored in the SE Manager. When this is done, the configuration data of the WebAdmin add-on is backed up or restored respectively in a way that ensures data consistency and the Catalina Tomcat service openutm is briefly terminated to permit this.

The WebAdmin add-on must be re-installed if the operating system on the management unit is updated. The installation operation recognizes that the WebAdmin configuration data is missing and automatically restores this on the basis of the last backup of the configuration data. If this backup is on the management unit not yet available, the WebAdmin add-on is working with an empty configuration database and the configuration data must be restored in the SE Manager.

28.12 SEM Audit logging















The WebAdmin add-on uses the SE Manager's audit logging functionality to log the modifications of the WebAdmin Configuration.




















In the SE Manager you can find the audit logging by clicking on the entry "Audit logging" under "Logging" in the sidebar. Please select "OPENUTM" in the column filter of column "Component" to see just the logging entries created by the WebAdmin add-on only.













29 Appendix

29.1 Icons used




29.1.1 Object icons



















Icon	Object Type
	Abstract syntax
	Access point
	Application context
	Bcamappl
	Character Set
	Cluster node
	Con
	Printer
	Print job
	Edit
	Gssb
	HTTP Descriptor
	IP Subnet
	Kset
	Kset (deleted)



Icon	Object Type
	Load module/shared object
	Lpap
	Lses
	Lses (deleted)
	Ltac
	Ltac (deleted)
	Lterm
	Lterm (deleted)
	Message (fetched from a UTM queue)
	Message collector
	Message module
	Message queue entry
	Mux
	Osi association (created)
	Osi association (generated)
	Osi Con
	Osi Lpap
	Program
	Program (deleted)

Icon	Object Type
PTC	Open transaction in the PTC state
	Pterm
	Pterm (deleted)
	Queue (temporary)
	Queue (temporary, deleted)
	Role
	Sfunc
	Tac
	Tac (deleted)
	Tacclass
	Tpool
	Transfer syntax
	User
	User (deleted)

29.1.2 Other icons

Icon	Meaning
	Application collection
	UTM cluster for which at least one of the activated node applications was available when last contacted
	UTM cluster that has not yet been contacted

Icon	Meaning
	UTM cluster for which no activated node application was available when last contacted
	UTM application If the icon describes a single administration application, this application was available when last contacted.
	UTM application that has not yet been contacted
	UTM application that was not available when last contacted
	Deactivated UTM cluster node application
	Application parameter
	Command Mode
	Host (WebAdmin object)
	UTM application (WebAdmin object)
	UPIC connection (WebAdmin object)
	WebAdmin user (WebAdmin object)
	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 last collected statistics value exceeded the defined upper threshold (Statistics)
	Statistic collector in which the last collected statistics value exceeded the defined lower threshold (Statistics)
	Statistic collector whose assigned "collecting" object does not exist (no longer exists) in the corresponding UTM application (Statistics)

Icon	Meaning
	Statistic collector in which the last attempt to collect a statistics value failed (Statistics). Possible cause: The statistics value that is to be collected is not available in the operating system version or variant of the corresponding UTM application.
	Action

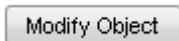
29.2 User interface terms

As the various user interface controls are often given different names, the terms used in this document are listed briefly below and described and/or illustrated.

■ Button

Clicking on a button generally causes an action to be performed. The text on the button briefly describes what will happen when the button is clicked.

Example:



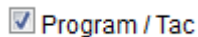
■ Insertion point

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.

■ Check box

Check boxes indicate options which can be activated or deactivated by clicking on the check box or associated text.

Example:



■ Cursor (or mouse cursor)

The cursor indicates the current position of the mouse 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.

■ Dropdown box

A dropdown box is basically a list box that is normally "closed". Only the entry that is currently selected in the picklist is visible. However, if you click on the down arrow or somewhere in the closed dropdown box, the entire picklist is displayed. Clicking on one of the entries in the picklist

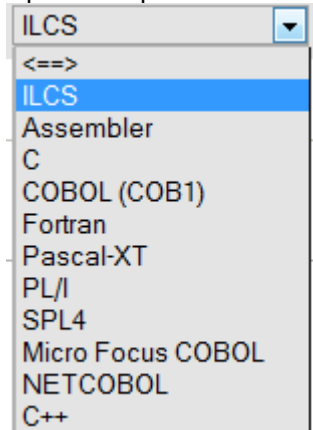
selects the item and closes the box again. Clicking on the arrow button again or pressing the **Esc** key will also close the box again.

Example:

Closed dropdown box:



Opened dropdown box:

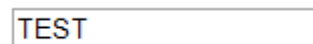


■ Edit box

An edit box enables you to enter a value (number or text).

In general, values in the clipboard can also be pasted into an edit box, or selected values in an edit box can be copied to the clipboard.

Example:



■ Property page

A property page generally allows you to view and to modify object properties. It consists of one or more sheets or pages that are arranged like tabs overlapping one another. Each individual sheet has a tab indicating the sheet's name. Only one sheet is fully visible at any given time. To view a sheet that is not fully visible, click on its tab. The sheet is then displayed (activated, foregrounded).

Below the sheets you will generally find the 'Refresh' and 'Modify Object' buttons, and sometimes also 'Delete Object'. There may also be other buttons for accessing object-specific functions.

Example:

The screenshot shows the configuration page for a Tac object named 'PUBSUBA'. The interface includes a header with the object name and a help icon. Below the header is a tabbed menu with 'General' selected. The main content area displays the following configuration details:

- Name: **PUBSUBA**
- Application/Host: sample2 / MC122
- Data Read at...: 2013-04-19 16:56:32
- Program: pubsuba
- Exit: <none>
- Tacclass: 16
- Type: Asynchronous
- Status: On
- Admin: No
- Call: Both
- Api: KDCS
- Queue Level: 32767
- Queue Mode: Standard
- PGWT: No
- Encryption Level: None

At the bottom right of the configuration area, the text 'Tac 'PUBSUBA'' is displayed. Below the configuration area are three buttons: 'Refresh', 'Modify Object', and 'Delete Object'.

Clicking the 'Modify Object' button switches to the Edit mode in which all properties that can be edited are presented in text boxes or dropdown boxes. Properties that cannot be edited are grayed out. The property page then looks something like this:

The screenshot displays the configuration page for a Tac object named 'PUBSUBA'. The interface includes a top navigation bar with tabs for 'General', 'Specific', 'Current', 'Access Control', 'Queue Read Access', and 'Queue Write Access'. The 'General' tab is active. The configuration details are as follows:

- Name: PUBSUBA
- Application/Host: sample2 / MC122
- Data Read at...: 2013-04-19 16:56:32
- Program: pubsuba
- Exit: <none>
- Tacclass: 16
- Type: Asynchronous
- Status: On (dropdown menu)
- Admin: No
- Call: Both
- Api: KDCS
- Queue Level: 32767
- Queue Mode: Standard
- PGWT: No
- Encryption Level: None

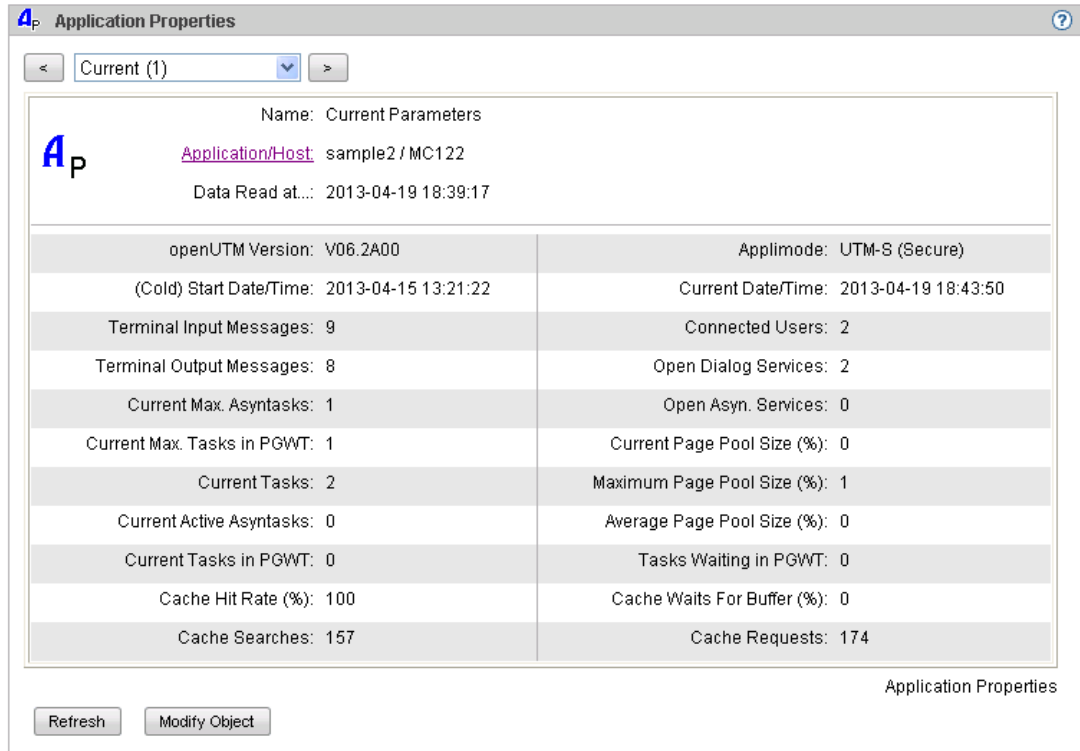
At the bottom of the configuration area, there are three buttons: 'Refresh', 'Apply', and 'Cancel'. The text 'Tac 'PUBSUBA'' is visible in the bottom right corner of the configuration area.

In Edit mode, only the 'Refresh', 'Apply' and 'Cancel' buttons appear.

Clicking the 'Cancel' button switches Edit mode off again. If you click the 'Apply' button, the changes you have made are saved. If saving is successful, Edit mode is exited. If saving is not successful, the reason will be indicated below the 'Apply' button.

Clicking the 'Refresh' button updates all the properties with the values specified for the object. Any changes you have made, but not yet saved, will be overwritten.

The tabs on the individual sheets may also be listed in a dropdown box. In WebAdmin this is the case for the property pages of UTM applications or UTM clusters:



■ **Property page section**

Controls that form a logical unit are grouped together in property page sections.

Example:



■ **Click (or mouse click)**

This generally means pressing the **left** mouse key. If the **right** mouse key is to be pressed, this will be 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 (right-clicking) often opens the general context menu for the web browser, not a specific WebAdmin function.

Note: The function of the left and the right mouse keys can be swapped in the Windows Control Panel.

■ **Navigation bar (secondary navigation)**

The navigation bar is displayed along the top of every page and indicates the position of the currently displayed page in the respective hierarchy. The currently displayed page is highlighted in color. Clicking on one of the other entries displays the page represented by this entry.

Example:

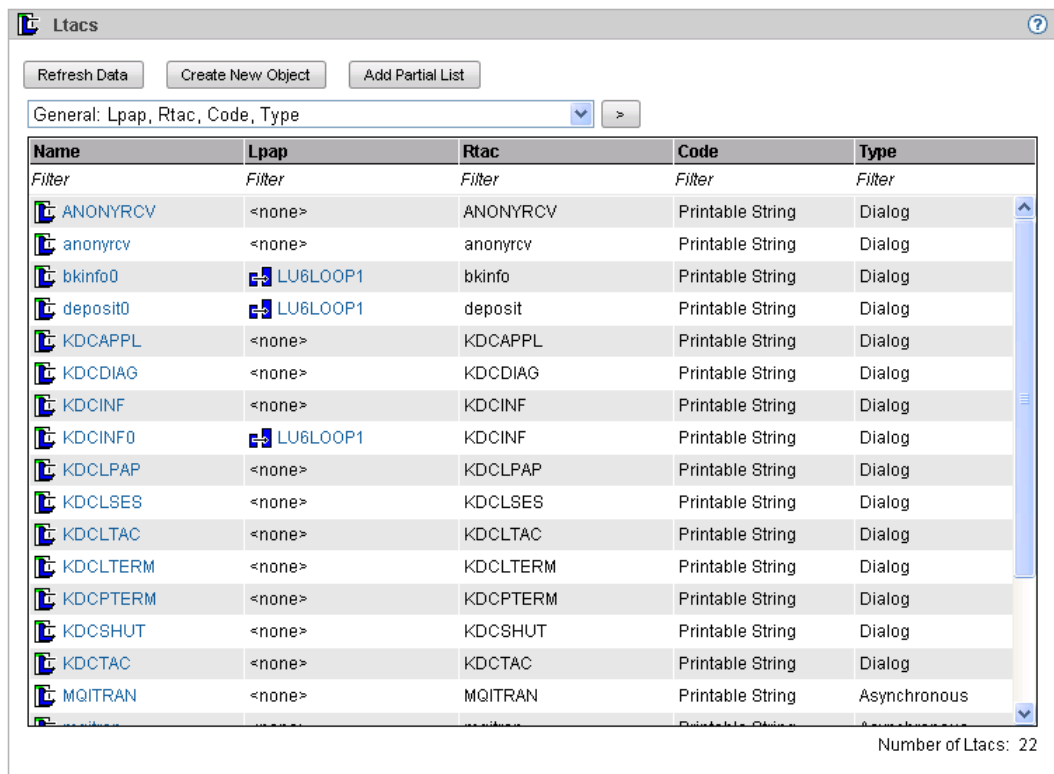
Home |  BS2000-Kollektion |  UTM-Cluster 1 |  Ltacs |  TAC4




















In this example, clicking on 'Ltacs' would display the object list with the Ltacs for the cluster 'UTM-Cluster 1', and clicking on 'UTM-Cluster 1' would display the primary page for the UTM cluster 'UTM-Cluster 1'.

■ Object list page

An object list page contains all the objects of a particular type in the form of a table.

Example:



Name	Lpap	Rtac	Code	Type
<i>Filter</i>	<i>Filter</i>	<i>Filter</i>	<i>Filter</i>	<i>Filter</i>
 ANONYRCV	<none>	ANONYRCV	Printable String	Dialog
 anonyrcv	<none>	anonyrcv	Printable String	Dialog
 bkinfo0	 LU6LOOP1	bkinfo	Printable String	Dialog
 deposit0	 LU6LOOP1	deposit	Printable String	Dialog
 KDCAPPL	<none>	KDCAPPL	Printable String	Dialog
 KDCDIAG	<none>	KDCDIAG	Printable String	Dialog
 KDCINF	<none>	KDCINF	Printable String	Dialog
 KDCINF0	 LU6LOOP1	KDCINF	Printable String	Dialog
 KDCLPAP	<none>	KDCLPAP	Printable String	Dialog
 KDCLSES	<none>	KDCLSES	Printable String	Dialog
 KDCLTAC	<none>	KDCLTAC	Printable String	Dialog
 KDCLTERM	<none>	KDCLTERM	Printable String	Dialog
 KDCPTERM	<none>	KDCPTERM	Printable String	Dialog
 KDCSHUT	<none>	KDCSHUT	Printable String	Dialog
 KDCTAC	<none>	KDCTAC	Printable String	Dialog
 MQITRAN	<none>	MQITRAN	Printable String	Asynchronous

Number of Ltacs: 22

■ Radio button

There are generally at least two radio buttons. They allow you to select just one of several options offered and are often grouped together in one 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:

- Shut Warn
- Shut Graceful
- Shut Normal
- Shut Kill

■ Scroll bar

You can use the scroll bar to change the displayed part of a long list, dropdown box, etc.

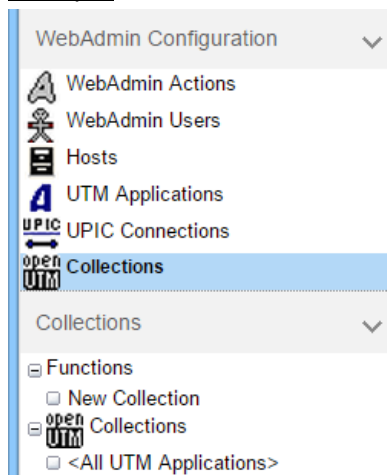
Example:



■ Sidebar (primary navigation)

The sidebar is displayed down the left-hand side of every page and allows you to access certain objects and object functions.

Example:



30 Index

A

Abort Bound Services.....	46
access lists.....	55
action.....	95
application collection.....	30
application functions.....	43
asynchronous job.....	65
authorizations.....	89

B

button.....	111
-------------	-----

C

check box	111
click	115
cluster.....	48
cluster functions.....	43
collection.....	30
command mode.....	92
configuration	
web server.....	16
configuration database.....	17
troubleshoot.....	17

D

dead letter queue.....	68
diagnostic dump.....	44
diagnostic trace.....	100
dropdown box	111

E

edit box	112
Encryption.....	45
Explicitly Sign-Off User.....	46

F

filtering.....	34
FTAC profile.....	91

H

hosts.....	24
------------	----

I

icons.....	107
for objects.....	107
other.....	109
incompatibilities.....	10
insertion point	111
installation.....	7

L

linked objects.....	35
logging.....	100

M

message	
for asynchronous job.....	67
message collectors.....	59
message dump.....	44
Message dump.....	93
message queues.....	65
mouse click	115
mouse pointer	see pointer

N

navigation	
primary.....	19, 117
secondary.....	20, 115
navigation bar.....	20, 115

O

object list	
functions.....	35
partial list.....	37
object list page.....	34, 116
object lists.....	33
objects	
create.....	42

- delete42
- filter.....34
- functions.....36, 40
- icons.....107
- linked.....35
- properties.....39
- save in file.....96
- sort.....35
- online help.....100
- online import.....46
- openFT.....90

P

- partial list37
 - add.....37
 - remove.....38
- pointer**111
- primary navigation19, 117
- print job.....69, 70
- printer69
- printer control Lterm.....69
- privileges
 - overall89
- property page**.....112
- property page section115
- property pages.....39
- property tooltips97

Q

- queue
 - service-controlled72

R

- rack.....19
- radio button**.....116
- RCCC.....94
- RCDC.....94
- refresh.....99
- roles.....55
- RSA keys45

S

- scroll bar**.....117
- secondary navigation.....20, 115
- security.....23, 89
- service-controlled queue72

- Show Open Transactions in PTC State45
- shutdown14
- sidebar19, 117
 - rack.....19
- sign-on service12
- sort.....35
- start14
- statistics74
 - collector icons74
 - collector list.....74
 - collector state75
 - create collector.....79
 - delete collector79
 - of TACs36
 - standard types82
 - threshold alert80
 - types.....82
- Statistik
 - Kollektoren79
- symbols107
- system platforms11

T

- tooltips97
- transaction-oriented administration jobs.....64

U

- UPIC
 - sign-on service.....12
- UPIC connections28
- user information
 - for asynchronous job66
- UTM access control53
- UTM applications25
- UTM cluster48
- UTM cluster application48

W

- web server
 - configuration.....16
 - HTTPS16
 - port16
 - SSL.....16
 - Windows service.....16
- WebAdmin objects.....22
- WebAdmin user23