

FUJITSU Software openUTM Enterprise Edition (Unix, Linux and Windows systems)

Version 6.3A  
January 2015

Release Notice

All rights reserved, including intellectual property rights.  
Technical data subject to modifications and delivery subject to availability. Any liability that the data and illustrations are complete, actual or correct is excluded. Designations may be trademarks and/or copyrights of the respective manufacturer, the use of which by third parties for their own purposes may infringe the rights of such owner.

© 2015 Fujitsu Technology Solutions GmbH

Fujitsu and the Fujitsu logo are trademarks or registered trademarks of Fujitsu Limited in Japan and other countries. BS2000 is a trademark of Fujitsu Technology Solutions GmbH in Germany and other countries.

<b>1</b>	<b>General</b>	<b>1</b>
1.1	Ordering	2
1.1.1	Licenses	2
1.1.2	Licenses for encryption	3
1.2	Delivery	3
1.2.1	Software on data media	3
1.2.2	Software for downloading	3
1.2.3	Software with encryption functions	3
1.2.4	XML for openUTM	3
1.2.5	Software for WS4UTM and BizXML2COBOL	4
1.3	Documentation	4
1.3.1	Availability	4
1.3.2	Documentation of the Open Group interfaces	4
<b>2</b>	<b>Software extensions</b>	<b>5</b>
2.1	New functions in openUTM (Unix, Linux and Windows systems) V6.3A	5
2.1.1	Additional system processes for internal tasks	5
2.1.2	Higher resolution of the consumed CPU time	5
2.1.3	New trace functions	5
2.1.4	Performance improvement	6
2.1.5	Other changes	6
2.1.6	Last simulation with "Workload capture & replay"	7
2.1.7	New and modified functions for openUTM WinAdmin	7
2.1.8	New functions for openUTM WebAdmin	7
2.2	Implemented change requests (CR)	8
<b>3</b>	<b>Technical information</b>	<b>8</b>
3.1	Resource requirements	8
3.2	Software configurations	8
3.3	Product installation	10
3.3.1	Installation on Unix/Linux platforms	10
3.3.2	Installation on Windows platforms	11
3.3.3	Installation together with C++ environment on Unix/Linux platforms	11
3.3.4	Installation logs	12
3.3.5	Installing openUTM-CRYPT	12
3.3.6	Notes: Kernel parameters for openUTM	12
3.4	Product use	14
3.4.1	Notes for upgrading to openUTM V6.3A	14
3.4.2	Converting a stand-alone application to a UTM-Cluster application	14
3.4.3	Client programs with a local connection to openUTM	14
3.4.4	Linking UTM OSI-TP applications on Unix and Linux platforms:	14
3.4.5	Notes on improving performance	14
3.4.6	Notes on a sample application for UTM-Cluster applications	15
3.5	Functions to be discontinued	15
3.6	Incompatibilities	15
3.7	Restrictions	16
3.7.1	Restrictions in UTM-Cluster applications	16
3.7.2	General restrictions	17
3.8	Procedure in the event of errors	17
<b>4</b>	<b>Hardware support</b>	<b>18</b>

# 1 General

The open and universal transaction monitor openUTM supports the creation and operation of transactional applications under the FUJITSU operating system BS2000 as well as under commonly available Unix systems (Solaris, HP-UX, AIX) or Linux and Windows systems.

FUJITSU Software openUTM lets you implement client/server architectures and create distributed transaction applications in homogenous and heterogeneous IT infrastructures (multi-tier architecture).

FUJITSU Software openUTM Client is a product which allows client/server communication with openUTM server applications and is available as a separate product for the Fujitsu operating system BS2000 (see separate Release Notice) as well as for Unix, Linux and Windows systems. There is a separate Release Notice for openUTM-Client.

WS4UTM is a tool which offers a convenient method for making program units of a UTM application available as web services. It is possible to call a UTM-Service as a web service from a web application implemented in Java. This allows UTM to be integrated into SOA solutions.  
(SOA = Service Oriented Architecture)  
There is a separate Release Notice for WS4UTM.

"XML for openUTM" provides openUTM with a convenient interface for creating and processing data in the form of XML documents.

A separate Release Notice is available for UTM-XML for openUTM.

The product FUJITSU Software BeanConnect enables UTM applications to connect to Java EE Application Servers. BeanConnect also includes a component for creating Java clients for openUTM. A separate Release Notice is available for BeanConnect.

The add-on product FUJITSU Software openUTM LU62 (Unix/Windows) V5.1 supports connections to CICS/IMS applications on IBM systems via SNA LU6.2 (see separate Release Notice).

BizXML2Cobol permits the creation of Cobol data structures from existing service definitions (as a WSDL description or XML schema) and code, which can be integrated in existing transactional Cobol applications so that these implement the predefined service. Thus, the top-down approach (from the business-relevant definition to implementation) in SOA projects is also supported for the inclusion of existing program logic.

A separate Release Notice is available for BizXML2Cobol.

Encrypted transfer of messages between the UTM application and the client is possible as a result of the additional encryption software from openUTM and openUTM Client with the carrier system UPIC. There is also additional licensing and special coding software for openUTM(Unix, Linux and Windows systems).

More information about the listed products and the data sheets is available under:

<http://de.ts.fujitsu.com/openUTM>

openUTM is part of the comprehensive middleware suite openSEAS. The openSEAS middleware suite enables the simple integration of individual applications and the interaction of existing and new applications. openSEAS perfectly complements the Fusion middleware of our strategic partner Oracle. Oracle Fusion middleware is a market-leading family of pre-integrated products, which ranges from application infrastructure development applications through to business intelligence.

Together this provides the platform for:

- Individual business-critical IT solutions
- Company portals
- Integration solutions for service-oriented architectures
- Web applications and portals
- Business processes

This Release Notice is a summary of the major extensions, dependencies and operating information with respect to openUTM(Unix, Linux and Windows systems) V6.3A and openUTM-CRYPT (Unix, Linux and Windows systems) V6.3A under the operating systems Solaris, HP-UX, AIX, Linux and Windows and all the information necessary for migrating from the predecessor version openUTM (Unix, Linux and Windows systems) V6.2A. Changes which have become known subsequently are updated and provided in this file.

The contents correspond to release status V6.3A of January 2015.

openUTM (Unix, Linux and Windows systems) is the server component of the client/server package openUTM Enterprise Edition (Unix, Linux and Windows systems). This package also includes: openUTM-Client (client/server communication), openUTM-WinAdmin and openUTM-WebAdmin (graphical administration interfaces).

Not all products are available on the various platforms.

Separate Release Notices are available for the individual products.

The UPIC carrier system for UTM-Client applications with local connections (UPIC local) is part of openUTM.

In the case of openUTM clients on Unix, Linux and Windows platforms connected remotely over a network, a choice can be made between two carrier systems (UPIC, OpenCPIC), each of which delivers a different functionality.

The Release Notices are supplied with the products in the form of README files. Any changes that came too late to be included in the printed documentation are noted in this file and published as part of a correction release.

If one or more predecessor versions are skipped when this current product version is in use, it may be necessary to also take account of the information from the Release Notices (and/or README files) of the predecessor versions.

All Unix, Linux and Windows product Release Notices including this one are available on the Internet under the following URL:  
<http://manuals.ts.fujitsu.com>

The use of names, trademarks, etc. in this Release Notice does not entitle readers to assume that these names/designations may be used without restriction by anyone; often the names/designations are protected by law or contract, even if this is not indicated here.

## **1.1 Ordering**

openUTM Enterprise Edition (Unix, Linux and Windows systems) V6.3A can be ordered from your local distributors.

openUTM Enterprise Edition (Unix, Linux and Windows systems) V6.3A is subject to the general terms and conditions of the software product purchase, maintenance, use and service agreement.

### **1.1.1 Licenses**

The licenses are part of the contract that you entered into with Fujitsu Technology Solutions when you purchased openUTM Enterprise Edition.

openUTM Enterprise Edition (Unix/Linux/Windows) is a product for which a license must be purchased. Licenses are required in order to use the product. These are provided as a basic license for each system or partition plus licenses for each user.

For distributed transaction processing, there are also additional licenses in 5 different variants depending on the number of parallel connections between the applications.

For development purposes, development packages are available with one basic license for each system or partition and 2 users for developing UTM applications. Licenses for additional developers can be purchased separately.

The user licenses for openUTM-Enterprise Edition include client and server users.

Licenses for UTM-Cluster applications:

There is no special cluster license for the use of UTM-Cluster functions.

You have to license openUTM on every node regardless of whether you run a stand-alone application or a node application on this computer.

### **1.1.2 Licenses for encryption**

The basic licenses do not include licenses for encryption functions. Additional licenses are required for encryption.

## **1.2 Delivery**

### **1.2.1 Software on data media**

openUTM (Unix, Linux and Windows systems) V6.3A without encryption functions is supplied on one openUTM EED-DVD which contains additional software products from the openUTM product family and communication products.

The openUTM (Unix, Linux and Windows systems) V6.3A delivery includes the product FUJITSU Software openUTM-WinAdmin V6.3A with which you can administer UTM applications with a convenient graphical user interface.

The openUTM EED-DVDs must be ordered separately.

### **1.2.2 Software for downloading**

The product FUJITSU Software openUTM WebAdmin V6.3A is released together with openUTM V6.3A. openUTM WebAdmin provides an additional graphic administration interface which runs on a web server in contrast to WinAdmin.

OpenUTM WebAdmin is probably available for downloading end of March 2015 on the openUTM web site <http://de.ts.fujitsu.com/openUTM>

### **1.2.3 Software with encryption functions**

openUTM (Unix, Linux und Windows systems) V6.3A with encryption functions is delivered on a separate openUTM EED- DVD Crypt. This DVD contains all the components of the openUTM EED-DVD plus the CRYPT functions.

The openUTM EED-DVD Crypt must be ordered separately.

If openUTM(Unix, Linux and Windows systems) V6.3A with encryption functions is to be used outside Germany a special export permit must be obtained in advance.

### **1.2.4 XML for openUTM**

The software "XML for openUTM" is an add-on to openUTM which is free of charge. Fujitsu Technology Solutions does not accept obligation for bug-fixing. The software "XML for openUTM" is a free add-on to openUTM. It is supplied "as is" and all use is at your own risk.

The Software is available on the web under  
<http://de.ts.fujitsu.com/openutm>

This component contains the Open Source XML parser libxml2, which is provided in accordance with the conditions of the MIT license. This is included in the application package ('rt') and is also available on the web under  
<http://www.opensource.org/licenses/mit-license.html>

For further details, refer to the separate Release Notice.

### **1.2.5 Software for WS4UTM and BizXML2COBOL**

Consulting project packages, in the context of which the software „WebServices for openUTM" (WS4UTM) and BizXML2Cobol is made available, can be purchased from your local Fujitsu regional office. The manuals can be downloaded from the Internet:  
<http://manuals.ts.fujitsu.com>

The use of the software is only permitted within the framework of the license purchased in the project package.  
For further details, refer to the separate Release Notice.

## **1.3 Documentation**

### **1.3.1 Availability**

The documentation is also available in Internet:  
<http://manuals.ts.fujitsu.com>

Manuals displayed with an order number can also be ordered as print versions.

The manuals are also included on the product DVDs.

### **1.3.2 Documentation of the Open Group interfaces**

Title:    Order number:

Distributed Transaction Processing  
The XCPIC Specification, Version 2  
X/Open CAE Specification ISBN 1 85912 135 7

Distributed Transaction Processing  
The XATMI Specification  
X/Open CAE Specification ISBN 1 85912 130 6

Distributed Transaction Processing  
The TX (Transaction Demarcation) Specification  
X/Open CAE Specification ISBN 1 85912 094 6

The X/Open manuals are available on the web (see above). They can be obtained in printed form from bookstores quoting the ISBN number.

## 2 Software extensions

### 2.1 New functions in openUTM (Unix, Linux and Windows systems) V6.3A

#### 2.1.1 Additional system processes for internal tasks

In addition to the processes specified via the start parameters, UTM starts up to three more processes which are kept free for internal open UTM tasks or for privileged administrator jobs. UTM system processes once started are retained for the entire application run even if the number of tasks is later reduced.

Generation and administration has been extended:

- Generation, KDCDEF statement MAX:
  - New operand PRIVILEGED-LTERM in order to mark a LTERM as privileged. All user jobs become privileged jobs via the log-in of a user with administration permissions.
  - Operand tasks: The maximum value has been reduced to 240 due to the additional system process.
  
- Administration interface KDCADMI:
  - Data structure *kc\_max\_par\_str*:  
New field *privileged\_lterm* for the generated privileged LTERM.
  - Data structure *kc\_tasks\_par\_str*:  
New fields *gen\_system\_tasks* and *curr\_system\_tasks* for the system processes.
  - Data structure *kc\_curr\_par\_str*:  
New field *curr\_system\_tasks* for the system processes.

#### 2.1.2 Higher resolution of the consumed CPU time

The consumed CPU time is now output in microseconds for TACs and in milliseconds for USERS. The following interfaces have been modified:

- KDCADMI:
  - Data structure *kc\_tac\_str*:  
New field *taccpu\_micro\_sec* for the average consumed time in microseconds.
  - Data structures *kc\_user\_str* and *kc\_user\_dyn1\_str*:  
New field *cputime\_msec* for the consumed time in milliseconds.
  
- Command interface KDCADM:
  - KDCINF type=TAC:  
TACCPU outputs the average consumed time in microseconds.
  - KDCINF type=USER:  
CPUTIME outputs the consumed time in milliseconds.
  
- KDCEVAL lists:
  - Some times in the KDCEVAL lists are now output in microseconds.

#### 2.1.3 New trace functions

Additional traces can be activated and deactivated in run mode:

- ADMI trace, i.e. trace the program interface to the administration (KDCADMI)
- X/Open traces (CPI-C, TX, XATMI)

The following interfaces have been modified:

- Start parameters:  
New start parameters ADMI-TRACE, CPIC-TRACE, TX-TRACE and XATMI trace to activate the traces
- KDCADMI:  
Data structure *kc\_diag\_and\_account\_par\_str*  
New fields *admi\_trace*, *cpic\_trace*, *tx\_trace* and *xatmi\_trace* to activate and deactivate the traces.

#### 2.1.4 Performance improvement

- UTM cache  
The UTM cache has been optimized in order to improve performance when the UTM cache is used intensively (e.g. with very comprehensive transaction data).
- UTM lock algorithm  
On open platforms the functions offered by the operating system for atom memory accesses are used for concurrent accesses to UTM internal administration data.
- UTM network connection  
The network connection on open platforms has been improved so that there are no longer any delays when sending data to UTM partner applications.

#### 2.1.5 Other changes

- Messages
  - The message area for system messages has been increased and now covers K001 to K399 (previously as far as K249). The following message areas have been moved:
    - The message numbers for mere KDCUPD messages now use K800 to K899 instead of K250 to K322.  
  
The messages output by KDCUPD and by the online import are seen as system messages and remain unchanged.
    - The message numbers for KDCCSYSL and KDCPSYSL messages now cover K600 to K649 instead of K550 to K599.
  - New message K235 if the name resolution for a processor name lasts too long.
  - The standard message targets have been modified with messages K162 and K163.
- KDCADMI
  - The fields *auto\_connect* for *kc\_lpap\_str* and *auto\_connect\_number* for *kc\_osi\_lpap\_str* have the feature GPD instead of PD as before, i.e. changes for these fields now have a global application effect. An administrative change in the features "Automatic connection set-up" for LPAP and "Number of connections" for OSI-LPAP has an effect beyond the end-of-application.
  - New field *max\_btrace\_lth* for *kc\_diag\_and\_account\_par\_str* for the maximum length of the recording data with an activated BCAM trace function.
- For platforms where UTM in 64-bit mode can run, KDCUPD enables the conversion from a 32-bit application environment to a 64-bit application environment. UTM currently supports the 64-bit mode only on Unix platforms.



- The Oracle user ID can also be specified in lower case letters with the KDCDEF statements DATABASE and RMXA.
- The installation procedure InstallAware is used on Windows systems. openUTM on Windows systems is thus supplied as MSI files.
- New example program ADJTCLT

Using the C part program ADJTCLT (ADJust TacClass Table) the user can manage how the processes are divided into the TAC classes depending on the current number of all processes and the current number of the asynchronous processes. The user creates a table with the required settings. The settings must be aligned so that at least one process is always free in order to handle other tasks, e.g. end-of-transaction processing of distributed transactions.

### 2.1.6 Last simulation with "Workload capture & replay"

The new function "Workload capture & replay" is used to record the communication between UTM applications and UPIC clients and then played with settable load profiles. The behaviour of the UTM application can thus be tested using high workload under real conditions.

"Workload capture & replay" consists of the following components:

- *UPIC capture:*  
Records the communication with the UPIC client.

The trace function BTRACE (BCAM trace) is used to record a UPIC session (Capture); the function is available on all server platforms.  
The UTM-BCAM trace file must be transferred binarily to Linux.

The utility kdcsort is also supplied with openUTM-EED V6.3. kdcsort is used to sort by time the communication recorded by BTRACE if the UTM application runs with more than one process during recording and several process-specific files have thus been created.

- *UPIC Analyzer:*  
Is used to analyze the recorded communication.
- *UPIC Replay:*  
Is used to play the recorded UPIC session with various load parameters (speed, number of clients).

*UPIC Analyzer* and *UPIC Replay* are part of openUTM-Client (Unix, Linux and Windows systems) V6.3 and are only available on 64-bit Linux systems.

### 2.1.7 New and modified functions for openUTM WinAdmin

- WinAdmin supports all the new features of UTM V6.3 regarding the program interface to the administration. These include, for example, the new trace functions or the display of the user's consumed CPU time in milliseconds.
- Introduction of a lifespan for statistic values in order to restrict the number of statistic values stored in the configuration database.

### 2.1.8 New functions for openUTM WebAdmin

#### Additional functions

WebAdmin offers additional functions which go beyond KDCADMI functionality and which were so far only available in WinAdmin:

- Display message queues (DADM functionality)
- Manage statistic collectors and display their values in a table (including the new function "Lifespan for statistic values")
- Show statistics in graphic Form (run graphic)
- Execute threshold actions for statistic collectors

### Support of new features in openUTM V6.3A

WebAdmin supports all the new features of UTM V6.3A regarding the program interface to the administration. These include, for example, the new trace functions or the display of the user's consumed CPU time in milliseconds.

## 2.2 Implemented change requests (CR)

Some of the new functions listed in section 2.1 were implemented as a result of change requests. A list of all change requests is shown below since there are change requests that cannot be directly assigned to a function topic:

- Installation Windows – improved entry in system control
- Support of processor name with more than 8 characters

# 3 Technical information

## 3.1 Resource requirements

Hard disk storage requirements for installation:

### openUTM:

Solaris SPARC	32/64-bit: < 1 GB
Linux	32/64-bit: < 500 MB
HP-UX(Risc)	32/64-bit: < 500 MB
HP-UX (Itanium)	32/64-bit: < 500 MB
AIX	32/64-bit: < 500 MB
Windows	32-bit: < 100 MB

The main memory requirements for an executable UTM application depends on the one hand on the types of products used, such as language connections, databases, X/Open interface and distributed processing, and on the other hand on the number and extent of the application programs.

The dynamic memory requirements of the UTM resources for process communication (shared memory) depend on the extent and parameterization of the KDCDEF generation.

## 3.2 Software configurations

Solaris SPARC	32/64 Bit	as of V10
Linux(SuSE)	32/64 Bit	as of SLES 10 for clusters as of kernel version 2.6.32.27-02.2.2
Linux(RedHat)	32/64 Bit	as of RHEL 5 for clusters as of kernel version 2.6.32.27-02.2.2
HP-UX (PA-RISC)	32/64 Bit	as of V11.31
HP-UX (Itanium)	32/64 Bit	as of V11.31
IBM-AIX	32/64 Bit	as of V6.1
Windows 7 / Server 2008 R2	32 Bit	
Windows 8.1 / Server 2012	32 Bit	

This version supports for the last time HP-UX(PA-RISC).

There has not yet been a general release for openUTM-Cluster applications on Windows.

The node applications of a UTM-Cluster application must run on identical platforms under the same bit mode.

NFS V4 is needed for access to the global cluster files of openUTM-Cluster applications.

In order to use the openSM2 function the following are valid:

openSM2	V9
---------	----

The following are supplied with the product for communication over TCP/IP:

PCMX(Solaris)	6.0B00
PCMX(Linux x86)	6.0B00
PCMX(HP-UX)	6.0B00
PCMX(HP-UX Itanium)	6.0B00
PCMX(AIX)	6.0B00
PCMX(Windows)	5.0A80

The use of PCMX is only license-free in connection with openUTM.

When using openUTM V6.3A, the PCMX version supplied with the DVD or a higher one has to be used.

The following versions are supported for distributed transaction processing with UTM applications:

openUTM(BS2000)	as of V6.0
openUTM(Unix systems, Linux, Windows)	as of V5.3

If problems occur with UTM versions that Fujitsu Technology Solutions is no longer obliged to maintain, the user is not entitled to warranty or correction claims. In this case we recommend that you update to current versions of openUTM.

For client/server communication optionally:

openUTM-Client(Unix systems/Linux/Windows)	as of V6.0A
(contains carrier systems UPIC V6.0; OpenCPIC V4.0)	
openUTM-Client(BS2000)	as of V6.0A

BeanConnect	as of V3.0
(also contains the component JConnect)	

openUTM-WinAdmin	V6.3
openUTM-WebAdmin	V6.3 (as of end of March 2015)

A C/C++ compiler is required for all Unix, Linux and Windows systems, for Windows Visual Studio 2005/2008 or higher. For the use of the QuickStart Kit on Windows Visual Studio 2010 is required.

Note about using the compiler XL C/C++ for AIX, V11.1:

There is a hindering error in conjunction with shared objects

<<http://www-01.ibm.com/support/docview.wss?uid=swg21443810>>

There is a patch for this problem from IBM which can be installed via "smit".

<<http://www-01.ibm.com/support/docview.wss?uid=swg24032451>>

When running an openUTM application and PCMX a 32 bit C run-time environment is mandatory. This also applies when the openUTM application runs in 64 bit mode!

On purely Linux 64 bit systems the 32 bit variant of the C standard library e.g.

glibc-2.12-1.107.el6\_4.4.i686

and the thereof dependent rpm- packages must be installed.

COBOL-Compiler:

Development of Cobol UTM applications:

NetCOBOL as of V10.4 (Linux64bit)

NetCOBOL as of V10.1 (Windows)

Micro Focus Visual COBOL as of V2.0

For runtime:

Cobol runtime licenses from MicroFocus

Note: The introduction of UTM system processes can result in problems with Cobol runtime licenses if the calculation is too exact.

When calculating the required number of licenses a maximum of three additional licenses must be taken into account.

On Unix and Linux systems, the following database systems can be used:

ORACLE as of 10gR2

INFORMIX as of V7.3 32-bit variant

INFORMIX as of V9.4 64-bit variant

On Windows systems, the following database systems can be used:

ORACLE as of 10gR2

Alternatively, the following products can be used for communication with IBM-SNA systems via LU6.2:

openUTM-LU62 (Unix or Windows systems) V5.1

and, depending on the operating system, the following third-party products:

Solaris:

SNAP-IX as of version 7.0.2.4; from Metaswitch Networks

Linux:

IBM Communications Server for Linux as of version 6.3

Windows:

IBM Communications Server for Windows, as of version 6.4

The following versions are supported for distributed transaction processing with Java EE applications:

BeanConnect as of V2.1

Note:

The software configuration includes some products that are still technically usable but for which later versions are available. You should always try and use the latest versions that are available.

## 3.3 Product installation

### 3.3.1 Installation on Unix/Linux platforms

On **Solaris** mount the DVD according to your operating system manual. In the appropriate directory you will find the products which can be installed with „pkgadd“:

```
pkgadd -d <DVD>/<package>.pkg
```

On **Linux** use the standard procedure „rpm“:

```
rpm -i <DVD>/<package>.rpm --nodeps [--ignorearch] [--replacefiles]
[--prefix=<install-dir>]
```

For installing PCMX, please read the corresponding readme file.

For installing OUTM-LU62, please read the corresponding readme file.

On **HP-UX** mount the DVD (mount -F cdfs -o cdcase <device> <mountpoint>).

Change to the product directory and run the installation script with:

```
./xpininstall.sh
```

For installing PCMX use:

```
swinstall -s <prod-dir>/cmx.ds -x target_type=tape -x reinstall=true CMX
```

On **AIX** mount the DVD (mount -v cdrfs -r <device> <mountpoint>).  
Change to the product directory and run the installation script with:  
./xpinstall

For installing PCMX use:

```
installp -a -F -d <prod-dir>/cmx.ds CMX.all
```

For installing OUTM-LU62, please read the corresponding readme file.

The Korn shell (ksh) must be in use in order to use openUTM.

The pdksh package (Public Domain Ksh) must be installed on Linux.

### 3.3.2 Installation on Windows platforms

For the installation on Windows, please change to the product directory and run "utm.msi" respectively "utm-crypt.msi".

The recommended installation sequence is as follows:

1. Microsoft Visual C++
2. (optional) selected Cobol Compiler
3. PCMX (from openUTM-DVD)
4. openUTM

### 3.3.3 Installation together with C++ environment on Unix/Linux platforms

When using C++ subprograms make sure that the C++ development environment of openUTM can be activated with the installation.

The following applies for the C++ Compiler:

- Solaris: type CC must return as result the C++ Compiler
- Linux: type g++ must return as result the C++ Compiler
- AIX: the C++ Compiler must be under /usr/vacpp/bin/cc
- HP: the C++ Compiler must be under /opt/aCC/bin/aCC

mainutmCC.o is then created during installation of openUTM.

mainutmCC.o can also be created after installation of openUTM (if required) by making the following entries:

```
UTMPATH=<utmpfad>  
          <utmpfad>/shsc/CCmainutm  
export UTMPATH
```

For further details, see the manual "Using openUTM Applications".

### 3.3.4 Installation logs

During installation on Unix/Linux systems, a file with the name <utmpath>/utm.log is created. This file stores the information output to the terminal during installation. During installation on Windows, a file with the name <utmpath>\install.log is created. This is required for a possible installation diagnosis.

### 3.3.5 Installing openUTM-CRYPT

- openUTM-CRYPT on Solaris and Linux

On Solaris and Linux, both openUTM and the add-on for the encryption functions must be installed.

In order to use the encryption functions, the code for the add-on product must be installed on the computer using the relevant installation procedure.

To do this, the same directory must be specified for LOCATION as the one in which openUTM was installed or is to be installed.

There are no dependencies with respect to the installation sequence. When the add-on software has been installed, it is not overwritten by subsequently installing openUTM.

When openUTM-CRYPT is installed on Solaris, you must respond to the following prompt with 'y':

"The following files are already installed on the system and are being used by another package:

<utmpfad>/sys/libutmencrypt.a

<utmpfad>/sys/libutmencrypt.so

Do you want to install these conflicting files [y,n,?,q]"

The add-on package must be installed on Linux using the additional switch --replacefiles.

- openUTM-CRYPT on HP-UX, AIX and Windows

In order to use the encryption functions, it is not necessary for the customer to install an add-on product. On these systems, openUTM-CRYPT contains both the openUTM product and the encryption functions.

### 3.3.6 Notes: Kernel parameters for openUTM

It may be necessary to increase the value of some of the Unix kernel parameters in order to run a UTM application. This affects the following kernel parameters (if more than one kernel parameter is specified per line, all the specified parameters must be changed to the same value):

- MAXUP Maximum number of processes per user

It is always possible that a process bottleneck can arise for individual users (error message: fork failed) if they have to start a very large number of processes. In this event, MAXUP must be adjusted.

- SEMMNS, SEMMNU Maximum number of semaphores and semaphore structures

The maximum number of semaphores required can be calculated using the description of MAX SEMKEY in the "Generating Applications" manual.

You must take into account the fact that other products may be installed on the computer that also require semaphores. For instance, at least 20 semaphores are required for every INFORMIX database that has been started.

- SEMMAP, SEMMNI Maximum number of semaphore identifiers

The semaphores are grouped to form semaphore fields and a (unique) semaphore identifier is assigned to each semaphore field.

openUTM and Informix each group 20 semaphores to form one semaphore field.

- SFSZLIM, HFSZLIM Maximum size of a file (in bytes)  
Values as of 48 MB (0x3000000) are generally sufficient even for large UTM dumps.  
The entry ULIMIT=... in the file /etc/default/login must also be adjusted to match the new value.  
ULIMIT specifies the maximum file size (for normal users) in blocks (1 block = 512 bytes).
- SDATLIM, HDATLIM Maximum swap space per process (in bytes)
- SVMMLIM, HVMMLIM Maximum virtual memory per process (in bytes)  
In order to allow larger UTM applications to be linked with COBOL program units, these values must be increased to at least 48 MB(0x3000000). When doing this, you must make sure that the swap area of the system is >= 48 MB.

#### Changing the kernel parameters under Solaris

The values can be changed using the entries in /etc/system. The system must be rebooted after you have changed the values.

Examples of entries:

```
set shmsys:shminfo_shmmax=268435456
set shmsys:shminfo_shmseg=32
set semsys:seminfo_semmmap=64
set semsys:seminfo_semmni=64
set semsys:seminfo_semmns=1600
set semsys:seminfo_semmnu=64
set semsys:seminfo_semume=900
```

#### Changing the kernel parameters under Linux

It is not generally necessary to change the kernel parameters. Please refer to the documentation from your Linux distributor for details on the files in which the values can be changed and on how to generate a new kernel.

#### Changing the kernel parameters under HP-UX

Parameter	Define	Default	Recommendation
Max. no. of processes/user	maxuprc	256	sufficient
Max. no. of semaphore IDs	semmni	64	sufficient
Max. no. of semaphores	semmns	256	1024
Max. no. of semaphore structures	semmnu	30	200
	semume	10	66
Swap space	-	2*Mem	4*Mem

It is recommended that you use the HP tool "/usr/bin/sam" for tuning. For details, refer to the HP manual: System Administration Tasks - HP-9000 Series 800 Computers.

#### Changing the kernel parameters under AIX

Parameter	Define	Default	Recommendation
Max. no. of processes/user	maxuprc	256	sufficient
Max. no. of semaphore IDs	semmni	36	64
Max. no. of semaphores	semmns	128	256
Swap space	-	2*Mem	4*Mem

It is recommended that you use the AIX tool "/usr/bin/smit" for tuning. For details, refer to the AIX manual: System Management Guides - RS/6000.

## 3.4 Product use

### 3.4.1 Notes for upgrading to openUTM V6.3A

Carry out the following steps when upgrading from an earlier version

- Generate KDCFILE and KDCROOT with the KDCDEF of V6.3A
- Compile the ROOT source
- Compile the MSGTAC program unit, if present
- If a separate message module is used, the source for the message module must first be newly generated with KDCMMOD and then compiled again
- Link the application program
- Transfer the user data with the KDCUPD of openUTM V6.3A to the new KDCFILE if the data is relevant

KDCUPD supports the following version upgrades:

openUTM(Unix systems/Linux/Windows) V5.3A	to	openUTM(Unix systems/Linux/Windows) V6.3A
openUTM(Unix systems/Linux/Windows) V6.0A	to	openUTM(Unix systems/Linux/Windows) V6.3A
openUTM(Unix systems/Linux/Windows) V6.1A	to	openUTM(Unix systems/Linux/Windows) V6.3A
openUTM(Unix systems/Linux/Windows) V6.2A	to	openUTM(Unix systems/Linux/Windows) V6.3A
openUTM(Unix systems/Linux/Windows) V6.3A	to	openUTM(Unix systems/Linux/Windows) V6.3A

Note:

With this Version (V6.3A) a transfer of the user data with KDCUPD from 32-bit mode to 64-bit mode and vice versa is possible

### 3.4.2 Converting a stand-alone application to a UTM-Cluster application

The following conversions are possible with a UTM-Cluster application:

- Conversion of a stand-alone application V6.3 to a UTM-Cluster application V6.3
- Conversion of a UTM-Cluster application V6.3 to a stand-alone UTM application V6.3

For detailed description see the manual "Using under Unix and Windows Systems".

### 3.4.3 Client programs with a local connection to openUTM

Client programs with a local connection to openUTM must be recompiled and linked again.

### 3.4.4 Linking UTM OSI-TP applications on Unix and Linux platforms:

The following sequence must be adhered to when specifying the OSS libraries:

The library libossutm.so must be specified first followed by the library liboss.so.

If the sequence is not adhered to, the following startup error occurs:

```
P001 Error on OSS call (o_create() .): - 1, 300, 199, 0
K060 Application run aborted; reason = XINI06
```

### 3.4.5 Notes on improving performance

Measurements have shown that performance benefits if the KDCFILE is split when generated.  
(Parameter MAX PGPOOLFS=number or MAX RECBUFFS=number)



### 3.4.6 Notes on a sample application for UTM-Cluster applications

The following must be taken into account when installing a sample application for a UTM node:

- For a sample application the conversion of an installed UTM node application into a stand-alone application is not supported.
- A filebase directory for a node application and a filebase directory for a stand-alone application must not be created in the file tree under the same superordinate directory.

## 3.5 Functions to be discontinued

### XA start parameters:

This version supports for the last time plain text specification of the user ID in the open string in the start parameter file.

### Network coupling:

This version supports for the last time the specification of the generation option "MAX NET-ACCESS".

### SW configuration:

This version supports for the last time HP-UX (PA-RISC).

## 3.6 Incompatibilities

### Program interface administration:

The data structure *kc\_create\_statments\_str* was extended with object-incompatible; the source compatibility is retained. Subprograms which use the administration opcode KC\_CREATE\_STATEMENTS must be re-compiled for UTM V6.3A.

The feature *auto\_connect\_number* now has the feature GPD instead of GP as before. An administrative change of *auto\_connect\_number* for a OSI-LPAP in UTM-F now has an effect beyond the end of the application.

### Command interface administration

The output of the command KDCINF TAC,L=KDCALL has been modified.

- The output of the field TACCPU has been extended from 8 to 10 digits and the unit of the field TACCPU has been changed from milliseconds to microseconds.
- The output of the field USED has been reduced from 9 to 7 digits.
- The outputs of the fields USED, ERROR, DBCNT, TACELAP and DBELAP move to the rear as a result of the changes.

The output of the command KDCINF USER,L=KDCALL has been modified.

- The output of the field CPUTIME has been extended from 7 to 10 digits and the unit of the field CPUTIME has been changed from seconds to milliseconds.
- The outputs of the fields SECCNT and LTERM move to the rear as a result of the change.

### Messages:

- Utilities KDCCSYSL and KDCPSYSL:  
The numbers of the messages output by the utilities KDCCSYSL and KDCPSYSL have been modified. They are now between K600 and K649 (previously K550 to K599).
- Utility KDCUPD:  
Some of the numbers of the messages output by the utility KDCUPD have been modified. They are now between K800 and K899 (previously K250 to K349).
- Utility KDCMMOD:  
The following transaction monitor messages have been withdrawn and may have to be removed from the change description for KDCMMOD for creating your own message module:  
K014, K015, K063, K087, K131, K171

### Support of the UTM Socket Protocol (USP):

As of UTM V6.3 only Socket partners are supported which use UTM Socket Protocol (USP) as of version UTM V5.2. Communication with Socket partners which use the old USP released with UTM V5.1 is rejected.

### KDCEVAL:

Trace2 list:

The unit of the difference value (output in "WAIT end" records) for the CPU time consumed in the partial program run has been modified from milliseconds to microseconds.

Trace and Trace2 list:

The wait time in database calls has been modified from milliseconds to microseconds.

### UTM server:

- CPI-C interface:  
The CPI-C trace can no longer be activated via the environment variables CPICTRACE, CPICBUFFER and CPICDUMP but only via the UTM start parameter CPIC-TRACE and the program interface to the administration.  
The standard name of the trace file is KDC.TRC.CPIC.appliname.host name.pid.
- TX interface:  
The TX trace can no longer be activated via the environment variable TXTRACE but only via the UTM start parameter TX-TRACE and the program interface to the administration.  
The standard name of the trace file is KDC.TRC.TX.appliname.hostname.pid. Only one trace file is written. No prefix for the trace file can be set using the job variable TXPATH
- XATMI interface:  
The XATMI trace can no longer be activated via the environment variable XTSVRTR but only via the UTM start parameter XATMI-TRACE and the program interface to the administration.  
The standard name of the trace file is KDC.TRC.XATMI.appliname.hostname.pid. Only one trace file is written. No prefix for the trace file can be set using the job variable XTPATH
- Database coupling with XA:  
The standard name of the trace file is KDC.TRC.XA.appliname.hostname.pid. No prefix for the trace file can be set using the environment variable KDCXADBG

### UTM(Unix/Linux systems):

Libraries:

As of openUTM V6.3A the libraries are only supplied as dynamic libraries on the platforms Solaris, Linux, HP-UX-Risc and HP-UX-Itanium

If - as before - the static openUTM libraries have been used to bind the UTM application programs (utmwork), the bind procedures must be modified.

Only static libraries are supplied for the platform AIX.

Processor name=local:

In openUTM applications so-called "local loop" connections which are generated via PRONAM=local or N-SEL=C'local' are no longer generated. If a connection set-up is required for one's own application, the primary computer name must be used in future instead of the local name.

## **3.7 Restrictions**

### **3.7.1 Restrictions in UTM-Cluster applications**

None

### 3.7.2 General restrictions

- Notes on using the sample application:  
No warranty is offered for the sample application and no service commitment is entered into.
- On AIX, the UTM system modules can only be linked in statically. This is the same as for predecessor versions of UTM V6.3.

## 3.8 Procedure in the event of errors

The following documentation is required for diagnosis if an error occurs:

- Detailed description of the error situation
- Versions/correction statuses of the software involved
- Exact details on the computer type

All error documentation should be as complete as possible. The following error documentation is required:

- UTM dumps for all work processes and the associated gcores. These files should be available as binary files, i.e. not in print-edited form.
- SYSLOG file(s)
- stdout/stderr logs of the UTM processes
- KDCDEF generation, start procedure and start parameter
- core files with the associated phases (utmwork) and shared objects. The shared objects can be determined using "ldd utmwork".
- Linker and compiler lists

Also in the event of errors in a UTM-Cluster application:

- all Cluster global files, logs, and DUMPS of all node applications
- In case of problems that were caused by the interaction of the node applications the log files of all other node applications
- The start procedure and the procedures specified during generations as EMERGENCY-CMD and FAILURE-CMD.

An attempt should be made to reproduce the errors using static libraries.

Procedure in the event of errors:

- Analyze the PENDER dump
- Reproduce the error using a suitable debugger such as dbx, gdb.
- Determine the call hierarchy for the core using a debugger

If the application terminates abnormally, the UTM dumps and gcores of all processes must be saved. If possible, the termination should be reproduced with TESTMODE=ON (start parameter).

Termination with signals

If a PENDER dump occurred with 70Z/XT10 or XT11 or if the application terminated with SIG010/SIG011 (SIGBUS/SIGSEGV signal), UTM signal handling should be deactivated with the start parameter START STXIT=OFF so that a core dump will immediately be generated when the faulty command is issued.

Before the application is restarted again, it is absolutely necessary to call kdcrem in order to delete or reset any semaphores and shared memories which may still be present as well as any status information on the application in the *<utmpfad>/applfile*.

The following additional documentation can also be created in the event of errors relating to the openUTM network connection:

- Messages from the UTM network processes in stdout/stderr
- CMX traces
- OSS traces
- UTM-BCAM trace
- Conversion file for containing entries for mapped host names
- In the case of heterogeneous connections, generation information and trace from openUTM LU6.2 and its SNA components and from the IBM SNA system.

For details on creating the UTM-BCAM traces and the OSS traces, refer to the manual "Messages, Debugging and Diagnostics".

The following additional documentation can also be created in the event of errors relating to databases:

- XA debug trace
- For further documentation, see the relevant Release Notice for the database system.

Following startup errors (message K049), for instance with insert 40 / 32, the UTM tool kdcrem must be called before the program is restarted.

## 4 Hardware support

Support is provided for the hardware, on which the mentioned operating system versions can run. This includes all systems based on Intel x86 technology, such as laptops, PCs, PRIMERGY systems; Itanium-based systems; SPARC systems, e.g. PRIMEPOWER systems and other Unix systems: IBM pSeries; HP PA-RISC, e.g. series 9000; other platforms on request.

CPU at least 250 MHz, RAM at least 256 MB

UTM-Cluster applications:

- A network file system/service (NFS) is necessary for the operation of UTM-Cluster applications

- openUTM (Solaris):

Nodes of a UTM-Cluster application can be systems with different Solaris versions, but standardized addressing (either 64-bit or 32-bit). It is not possible to use nodes with other operating systems (Linux, BS2000/OSD).

- openUTM (Linux):

Nodes of a UTM-Cluster application can be systems with different Linux versions, but standardized addressing (either 64-bit or 32-bit). It is not possible to use nodes with other operating systems (Unix systems, other Linux distributions, BS2000/OSD).