
HP Unified Correlation Analyzer



Unified Correlation Analyzer for Event Based Correlation

Inference Machine

Version 3.3

Installation Guide

Edition: 1.0

Windows 7 & RHEL 5.9-5.11, 6.4-6.6, 7.0-7.1 Operating Systems

64 bits architecture only

September 2015

Legal notices

Warranty

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

License requirement and U.S. Government legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright notices

© Copyright 2015 Hewlett-Packard Development Company, L.P.

Trademark notices

Adobe®, Acrobat® and PostScript® are trademarks of Adobe Systems Incorporated.

HP-UX Release 10.20 and later and HP-UX Release 11.00 and later (in both 32 and 64-bit configurations) on all HP 9000 computers are Open Group UNIX 95 branded products.

Java™ is a trademark of Oracle and/or its affiliates.

Microsoft®, Internet Explorer, Windows®, Windows Server®, and Windows NT® are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Oracle® is a registered U.S. trademark of Oracle Corporation, Redwood City, California.

UNIX® is a registered trademark of The Open Group.

X/Open® is a registered trademark, and the X device is a trademark of X/Open Company Ltd. in the UK and other countries.

Red Hat® is a registered trademark of the Red Hat Company.

Linux® is a registered trademark of Linus Torvalds in the U.S. and other countries.

Contents

Preface	6
Chapter 1.....	9
Introduction	9
1.1 Licensing	9
1.2 Disk requirements.....	9
1.3 Software prerequisites	10
1.3.1 Setting the <i>JAVA_HOME</i> environment variable	10
Chapter 2.....	12
Installation	12
2.1 Installing on Windows.....	12
2.2 Installing on Linux.....	12
Post-installation	13
2.3 TeMIP as NMS	13
2.4 Code signing.....	16
Chapter 3.....	18
Uninstallation	18
3.1 Uninstalling on Windows	18
3.2 Uninstalling on Linux	18

Tables

Table 1 - Software versions	7
Table 2 - UCA for EBC Inference Machine product names and features	9
Table 3 – Disk Requirements for UCA for EBC Inference Machine Development Kit	10
Table 4 – Software Prerequisites for UCA for EBC Inference Machine Development Kit	10

Figures

Figure 1 - Setting the *JAVA_HOME* environment variable on Windows systems..... 11

Preface

The purpose of this document is to collect all the information related to HP UCA for EBC Inference Machine installation and uninstallation.

Product name: Unified Correlation Analyzer for Event Based Correlation Inference Machine

Product version: 3.3

Kit version: V3.3

Intended audience

The intended audience of this guide is primarily developers: customers, or HP consultants, who need to create an Inference Machine Value Pack with Problem Detection, or a Topology State Propagator, or both, in UCA for EBC.

Prerequisites

It is presumed that the readers of this document have some basic knowledge of UCA for EBC.

The readers are advised to read the following documents first:

- “*HP UCA for Event Based Correlation – Installation Guide*”
- “*HP Unified Correlation Analyzer for EBC Inference Machine, version 3.3, Release Notes*”

Software versions

The term UNIX is used as a generic reference to the operating system, unless otherwise specified.

The software versions referred to in this document are listed in the following table.

Table 1 - Software versions

Product version	Supported operating systems
UCA for Event Based Correlation Server, version 3.3	<ul style="list-style-type: none">• HP-UX 11.31 for Itanium• Red Hat Enterprise Linux Server, 64 bits, Release 5.9-5.11, 6.4-6.6, 7.0-7.1
UCA for Event Based Correlation Channel Adapter, version 3.3	<ul style="list-style-type: none">• HP-UX 11.31 for Itanium• Red Hat Enterprise Linux Server, 64 bits, Release 5.9-5.11, 6.4-6.6, 7.0-7.1
UCA for Event Based Correlation Software Development Kit, version 3.3	<ul style="list-style-type: none">• Windows 7 64 bits• Red Hat Enterprise Linux Server 64 bits, Release, 5.9-5.11, 6.4-6.6, 7.0-7.1
UCA for Event Based Correlation Inference Machine Kit, version 3.3	<ul style="list-style-type: none">• Windows 7 64 bits• Red Hat Enterprise Linux Server, 64 bits, Release 5.9-5.11, 6.4-6.6, 7.0-7.1

Typographical Conventions

Courier font:

- Source code and examples of file contents
- Commands that you enter on the screen
- Pathnames
- Keyboard key names

Italic text:

- Filenames, programs and parameters
- The names of other documents referenced in this manual

Bold text:

- To introduce new terms and to emphasize important words.

Associated documents

The following documents contain useful reference information:

References

[R1] *Unified Correlation Analyzer for Event Based Correlation Reference Guide*

[R2] *Unified Correlation Analyzer for Event Based Correlation Value Pack Development Guide*

[R3] *Unified Correlation Analyzer for Event Based Correlation Installation Guide*

[R4] *Unified Correlation Analyzer for Event Based Correlation User Interface Guide*

[R5] *Unified Correlation Analyzer – Clustering and HA Guide*

[R6] *UCA for EBC JavaDoc - Inference Machine*
(C:\%UCA_EBC_DEV_HOME%\apidoc\inference-machine\index.html)

[R7] *UCA for EBC Inference Machine User Guide*

Support

Visit the HP Software Support website at <https://softwaresupport.hp.com/> for contact information, and details about HP software products, services, and support.

The Software support area of the website includes the following:

- Downloadable documentation
- Troubleshooting information
- Patches and updates
- Problem reporting
- Training information
- Support program information

Introduction

The Inference Machine Development Kit contains the Eclipse environment, including plug-ins, to develop an Inference Machine Value Pack, with Problem Detection, or Topology State Propagator, or both.

The Inference Machine Development Kit is an extension of the UCA EBC Development Kit.

Note that the deployment of an Inference Machine Value Pack is not covered in this document.

☞ See [R2] *Unified Correlation Analyzer for Event Based Correlation Value Pack Development Guide* for information on how to deploy a UCA EBC Value Pack.

☞ See [R7] *UCA for EBC Inference Machine User Guide* for information on Inference Machine Value Pack.

1.1 Licensing

The UCA for EBC Inference Machine Development Kit contains two distinct licensed products:

- UCA for EBC Problem Detection
- UCA for EBC Topology State Propagator

The following table shows the related UCA for EBC features:

Table 2 - UCA for EBC Inference Machine product names and features

Product name (license key)	UCA for EBC feature
UCA for EBC Problem Detection Value Pack	UCA for EBC Problem Detection Framework
UCA for EBC Topology State Propagator Value Pack	UCA for EBC Topology State Propagator Framework

For the product numbers associated with the product names, or for any questions related to licensing, contact your local HP sales representative, or HP partner.

1.2 Disk requirements

Make sure that the available disk space corresponds to the minimum disk requirements of the UCA for EBC Inference Machine Development Kit.

Table 3 – Disk Requirements for UCA for EBC Inference Machine Development Kit

Type	Minimum requirements
Temporary disk space	12 MB minimum: 4.5 MB minimum for the uca-ebp-im-packaging-3.3 msi.zip/linux.tar file 7.5 MB minimum for the UCA-EBC-DEVIM-V3.3-00B msi/noarch.rpm file (extracted from the above file)
Permanent disk space	10 MB minimum for UCA for EBC Inference Machine Development Kit v3.3 installed on the system

1.3 Software prerequisites

The UCA for EBC Inference Machine Development Kit is installed on top of the UCA for EBC Development Kit. It brings the ability to use UCA for EBC Problem Detection, or Topology State Propagator scenarios, or both, in the same, or in separate value packs.

Table 4 – Software Prerequisites for UCA for EBC Inference Machine Development Kit

Product	Version	Operating System
UCA for EBC Inference Machine Development Kit	3.3	Windows or Linux
Java	Java 1.7 or 1.8	Windows or Linux

The following products support both Java 1.7 and 1.8:

- UCA for EBC v3.3 Server
- UCA for EBC v3.3 Development Toolkit
- UCA for EBC v3.3 Value Packs

Note

UCA for EBC v3.3 Topology Extension supports only Java 1.7 (due to the use of Neo4J 1.9.x)

1.3.1 Setting the *JAVA_HOME* environment variable

The *JAVA_HOME* environment variable must be set before using UCA for EBC Inference Machine Development Kit

1.3.1.1 Setting the variable on Windows

If Java is not yet installed on your system, the latest JRE/JDK package for the Microsoft Windows operating systems can be downloaded for free from <http://java.com/en/download/manual.jsp>.

1. In the *Control Panel*, open *System Properties*.

2. In the *Advanced* tab, click *Environment Variables*.
3. Set the *JAVA_HOME* environment variable according to the location of your JDK.

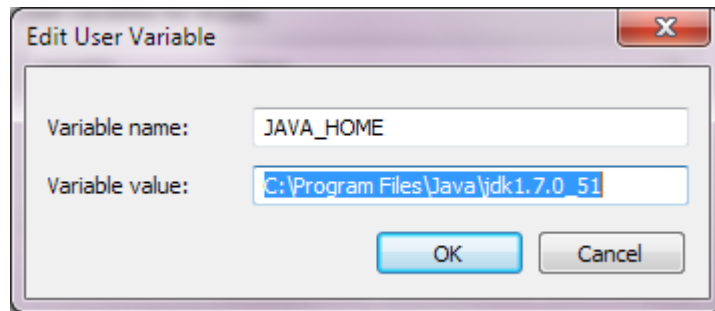


Figure 1 - Setting the *JAVA_HOME* environment variable on Windows systems

1.3.1.2 Setting the variable on Linux

1. Check if you already have Java installed:

```
$ rpm -qa | grep jdk
```

2. Depending on your shell, and on the location of the Java JRE/JDK software, use one of the following commands to set the *JAVA_HOME* environment variable:

Example for **cs**h-like shell:

```
$ setenv JAVA_HOME /usr/java/jdk1.7.0_51
```

Example for **sh**-like shell:

```
$ export JAVA_HOME=/usr/java/jdk1.7.0_51
```

Red Hat Enterprise Linux Server comes with OpenJDK Java VM. Output example:

```
java-1.7.0-openjdk-1.7.0.9-2.3.4.1.e16_3.x86_64  
java-1.7.0-openjdk-devel-1.7.0.9-2.3.4.1.e16_3.x86_64
```

You can also download, for free, the latest Java packages, HotSpot Java VM, from Oracle, from <http://java.com/en/download/manual.jsp>. This is usually installed under */usr/java*. Outlook example:

```
jdk-1.7.0_51-fcs.x86_64
```

Installation

This chapter explains how to install the Inference Machine Development Kit.

First, install the UCA for EBC Development Kit, then install the UCA for EBC Inference Machine Development Kit.

☞ See [R3] *Unified Correlation Analyzer for Event Based Correlation Installation Guide* for information on how to install the UCA for EBC Development Kit.

2.1 Installing on Windows

The UCA for EBC Inference Machine Development Kit is provided in the `uca-evp-im-packaging-3.3-msi.zip` file.

Before installing the kit, make sure that you have administrator rights.

1. Unzip `uca-evp-im-packaging-3.3-msi.zip` by using, for example, Winzip, or 7zip. Unzip the content into a temporary directory.
2. Double-click on `UCA-EBC-DEVIM-V3.3-00B.msi`.
3. Accept the terms of the license agreement.
4. For the destination folder, use the same location as the one defined by the `%UCA_EBC_DEV_HOME %` variable.
5. Click Install.
6. Click Finish.

Note

If installation fails because the same package has been installed already, you'll need to uninstall the package first.

2.2 Installing on Linux

The UCA for EBC Inference Machine Development Kit is provided in the `uca-evp-im-packaging-3.3-linux.tar` file.

1. Before installing the kit, make sure you are root and a `uca` user is created on your system. If `uca` user is not created, you can create it:
 - a. `# groupadd uca`
 - b. `# useradd -g uca -s /bin/bash -m uca`

2. (Optional) If you are a Linux user with no root access, you need to set up your RPM database in a directory with write access.

Example: if \$HOME/.rpmdb is the default directory used by UCA-EBC Server kits as well, enter:

```
$ echo %_dbpath $HOME/.rpmdb > $HOME/.rpmmacros
```

3. # cd 'a temporary directory of your choice'

4. # tar xvf uca-evp-im-packaging-3.3-linux.tar

5. # ./install-uca-ebc-im.sh [-r <root_directory>]

The destination folder can be specified with the `-r` option and has to be set to the one used by *UCA-EBC-DEVTOOLKIT* package, known as `$UCA_EBC_DEV_HOME`, where the default is `/opt/UCA-EBC-DEV`.

On Linux, there is no sub directory for the version.

Note

Note: The described installation steps automatically uninstall any previous installation of the same package if needed. However, any previous UCA-EBC-DEVPD package has to be uninstalled if the `$UCA_EBC_DEV_HOME` is the same as the one used by UCA-EBC-DEVPD.

Post-installation

2.3 TeMIP as NMS

Note: This post-installation step is optional. It only applies if the target for your Inference Machine Value Packs is TeMIP.

TeMIP comes in a package with a project called TPD, containing all registered Custom Alarm Object Fields. For the UCA for EBC Inference Machine Value Packs to function, the following new user-defined TeMIP Alarm Object attributes need to be added to the TeMIP Dictionary on the systems hosting your TeMIP directors:

- PB

Type: Latin1String

ID:10100

This attribute defines the category of the alarm:

- ProblemAlarm (parent)
- ServiceAlarm (parent)
- SubAlarm (child)
- SubProblemAlarm (parent and child)

- SubServiceAlarm (parent and child)
 - Candidate (not yet a child), only for problem detection
 - Alarm (no more a child or a parent)
- Grouping Keys

Type: Latin1String

ID:10101

This attribute is used by TPD to support real-time parent-children navigation in the TeMIP Client.
 - Number of Cleared Alarms

Type: Unsigned32

ID:10102
 - Number of Total Alarms

Type: Unsigned32

ID:10103
 - Number of Acknowledged Alarms

Type: Unsigned32

ID:10005
 - Number of Outstanding Alarms

Type: Unsigned32

ID:10006

These attributes are available on the following versions and above:

- Linux, TFR (TeMIP framework) V61L Maintenance Release
- HP-UX, PHSS_43236 E-Patch on HP-UX IA platform (TFR v6.1)

These user-defined fields are added through the dedicated tool, on the machine where the TeMIP server runs.

1. Run the following command to configure *temip_ah_user_defined_attr*, which is located in */usr/opt/temip/bin*, and the project TPD:

```
# temip_ah_user_defined_attr -project TPD
```

2. Check the data defined in the TPD, by checking your *temip_ah_user_defined_attr.data* file at the following location:

```
usr/opt/temip/mmtoolkit/msl/temip_ah_user_defined_attr.data
```

Output example:

```
...
PROJECT TPD
ATTRIBUTE
name="PB"
id=10100
```

```
type=Latin1String
settable=Y
archived=YES
summarized=Y
event=Y
createAO=Y
SA=Y
createSA=Y
END ATTRIBUTE
ATTRIBUTE
name="Grouping Keys"
id=10101
type=Latin1String
settable=Y
archived=YES
summarized=Y
event=Y
createAO=Y
SA=Y
createSA=Y
END ATTRIBUTE
ATTRIBUTE
name="Number of Cleared Alarms"
id=10102
type=Unsigned32
settable=Y
archived=YES
summarized=Y
event=Y
createAO=Y
SA=Y
createSA=Y
END ATTRIBUTE
ATTRIBUTE
name="Number of Total Alarms"
id=10103
type=Unsigned32
settable=Y
archived=YES
summarized=Y
event=Y
createAO=Y
SA=Y
createSA=Y
END ATTRIBUTE
ATTRIBUTE
name="Number of Acknowledged Alarms"
id=10005
type=Unsigned32
settable=Y
archived=YES
summarized=Y
event=Y
createAO=Y
SA=Y
createSA=Y
END ATTRIBUTE
ATTRIBUTE
name="Number of Outstanding Alarms"
id=10006
type=Unsigned32
settable=Y
archived=YES
```

```

summarized=Y
event=Y
createAO=Y
SA=Y
createSA=Y
END ATTRIBUTE
END PROJECT

```

3. To confirm that the attributes listed before are correctly added in the dictionary, enter:

```
# temp_ah_user_defined_attr
```

Output example:

```

----- User Defined Attributes -----
[##] Pres. Name = MSL ID : Data Type - Symbol Settable
[ 1] PB = 10100 : Latin1String -> AO_PB
[ 2] Grouping Keys = 10101 : Latin1String -> AO_GROUPING_KEYS
[ 3] Number of Cleared Alarms = 10102 : Unsigned32 -> AO_NUMBER_OF_CLEARED_ALARMS
[ 4] Number of Total Alarms = 10103 : Unsigned32 -> AO_NUMBER_OF_TOTAL_ALARMS
[ 5] Number of Acknowledged Alarms = 10005 : Unsigned32 -> AO_NUMBER_OF_ACKNOWLEDGED_ALARMS [
[ 6] Number of Outstanding Alarms = 10006 : Unsigned32 -> AO_NUMBER_OF_OUTSTANDING_ALARMS

```

Alternatively, you can check the dictionary:

```
# mcc_dap_browser&
```

Then

```
operation_context->alarm_object->partition->user_defined
```

2.4 Code signing

This post-installation step is optional; however, HP strongly recommends using signature verification on its products. Customers have the choice of running this verification according to their IT policies.

The below mentioned procedure allows you to assess the integrity of the delivered product before installing it, by verifying the signature of the software packages.

- 1) Install the *GnuPG* tool.
 - a. Get the GPG software for Windows from the GnuPG website. You can find it in the Binaries subsection.
 - b. Verify the downloaded software via its *SHA1* checksum if it is a first installation, or via its associated signature if a previous version is already installed.
 - c. Install the downloaded software.
 - d. Start a cmd.exe to have a windows shell.
- 2) Download *hpPublicKey*.
 - a. Open the command prompt.

- b. Browse to the bin directory in the GnuPG installed folder.
- c. Get the *hpPublicKey* from the following location:
<https://h20392.www2.hp.com/portal/swdepot/displayProductInfo.do?productNumber=HPLinuxCodeSigning>
- d. Follow the instruction found in the website.
- e. Save it as *hpPublicKey.pub*.

3) Import *gpg-hpPublicKey.pub*.

Enter:

```
gpg --import <location of HPSignClient installed directory>\gpg-hpPublicKey.pub
```

4) Verify the signed binary.

Enter:

```
gpg --verify <Problem Detection.sig > <Problem Detection .zip >*
```

Output example:

```
gpg: Signature made Wed Nov 17 12:32:46 2010 IST using DSA key ID 2689B887
gpg: Good signature from "Hewlett-Packard Company (HP Codesigning Service)"
gpg: WARNING: This key is not certified with a trusted signature!
gpg:   There is no indication that the signature belongs to the owner.
Primary key fingerprint: FB41 0E68 CEDF 95D0 6681 1E95 527B C53A 2689 B887
```

Note

Message "Good signature from "Hewlett-Packard Company (HP Codesigning Service)" " indicates that the code sign verification is successful.

Uninstallation

This chapter explains how to uninstall the Inference Machine Development Kit.

3.1 Uninstalling on Windows

Right-click on *UCA-EBC-DEVIM-V3.3-00B.msi* and select *Uninstall*.

3.2 Uninstalling on Linux

`$UCA_EBC_DEV_HOME` usually refers to `/opt/UCA-EBC-DEV`.

1. Enter:
`# $UCA_EBC_DEV_HOME/bin/uninstall-uca-ebc-im`
2. Enter the index number, which is usually 0
3. Press *Enter*.