hp Unified Correlation Analyzer

Unified Correlation Analyzer
for
Event Based Correlation

Version 3.3

Release Notes

Edition: 1.0

For the Operating Systems:
Linux (RHEL 5, 6 & 7)
HP-UX (11.31)
Windows© (for development toolkit)

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

Trademark Notices

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

HP-UX Release 11.31 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.

Firefox® is a registered trademark of the Mozilla Foundation.

Google Chrome® is a trademark of Google Inc.

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.

Neo4j is a trademark of Neo Technology.

Contents

Figures

Tables

Preface

These Release Notes describe critical information related to the HP UCA for Event Based Correlation product.

Product Name: Unified Correlation Analyzer for Event Based Correlation

Product Version: 3.3

Kit Version: V3.3

Please read this document before installing or using this Software.

Intended Audience

Here are some recommendations based on possible reader profiles:

- Solution Developers
- Software Development Engineers

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 as follows:

Product Version	Supported Operating systems
UCA for Event Based Correlation Server Version 3.3	• HP-UX 11.31 for Itanium • Red Hat Enterprise Linux Server, 64 bits, Release RHEL 5, 6 & 7
UCA for Event Based Correlation Channel Adapter Version 3.3	• HP-UX 11.31 for Itanium • Red Hat Enterprise Linux Server, 64 bits, Release RHEL 5, 6 & 7
UCA for Event Based Correlation Topology Extension Version 3.3	• HP-UX 11.31 for Itanium • Red Hat Enterprise Linux Server, 64 bits, Release RHEL 5, 6 & 7
UCA for Event Based Correlation Software Development Kit Version 3.3	• Windows 7 64 bits • Red Hat Enterprise Linux Server release RHEL 5, 6 & 7

Table 1 - Software versions

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

HP UCA for Event Based Correlation - Installation Guide

HP UCA for Event Based Correlation - Administration, Configuration, and Troubleshooting Guide

HP UCA for Event Based Correlation - Reference Guide

HP UCA for Event Based Correlation - Topology Extension Guide

HP UCA for Event Based Correlation - Value Pack Development Guide

HP UCA for Event Based Correlation - User Interface Guide

HP UCA for Event Based Correlation – Clustering and HA Guide

Support

Please visit our HP Software Support Online Web site at https://softwaresupport.hp.com/ for contact information, and details about HP Software products, services, and support.

The Software support area of the Software Web site includes the following:

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

Chapter 1

New features provided by this release

The previous version of UCA for Event Based Correlation V3.3 was UCA for Event Based Correlation V3.2.

The following features and fixes have been implemented since.

1.1 Support of UMB (Unified Mediation Bus) mediation

The support of the UMB mediation is the major feature of this minor release.

UCA-EBC can now consume events from UMB flows and also provide static and dynamic flow services, as well as execute actions on remote UMB Adapters.

UMB support in UCA EBC is defined by the `use.new.generation.adapter` property in the `uca-ebc.properties` configuration file. By default this property is set to `true`, i.e. UMB support in UCA EBC is enabled.

1.1.1 Consuming UMB event flows

Each UCA-EBC value pack can be configured to collect events from UMB event flows provided by some distant UMB adapter Flow Providers.

Such configuration is made in the `ValuePackConfiguration.xml` file of the value pack in the 'mediationFlows' section.

An UMB consumer flow is defined with the Tag "UMBmediationFlow" has in the example below:

```
   <mediationFlows>
        <UMBmediationFlow name="temipFlow"
targetAdapterName="TeMIP" targetFlowName="PDExampleFlow"
automaticStart="true">
            <flowParameters>
                <flowParameter key="configurationFile"
value="TeMIP configuration.xml"/>
                <flowParameter key="operationContext"
value="myOC1"/>
                <flowParameter key="operationContext"
value="myOC2"/>
            </flowParameters>
        </UMBmediationFlow>
        <UMBmediationFlow name="SmartFlow"
                          targetAdapterName="SMART"
                          targetFlowName="smartFlow1"
                          automaticStart="true"/>
    </mediationFlows>
```

In this section, each UMBmediationFlow is defined specifying the following attributes:

- **name**: this is the local flow name.
- **targetAdapterName**: is the identifier of the adapter providing the production flow service
- **targetFlowName**: is the name of the flow definition on the target Adapter
- **automaticStart**: can be 'true' or 'false'. Indicates if the flow must be started along with the value pack. If omitted, the default value is 'true'. When set to false, the flow is not started at VP startup; it will have to be started manually from the GUI to become active.

Some flow creations require parameters to be provided (expected by the producer side). Flow parameters are defined in the `<flowParameters>` section of the `UMBmediationFlow`. Each parameter is a key/value pair defined with the `<flowParameter>` Tag with the following attributes:

- **key**: the parameter name
- **value** the parameter value

1.1.2 Defining static UMB flows

1.1.2.1 Configuration

For static Flows the collectorClass must be set to:
`com.hp.uca.expert.mediation.adapter.UcaStaticCollector`

No flow parameters need to be defined.

Here is an example of Static Flow Service definitions for UCA-EBC:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<adapter name="UCA-EBC" version="1.0"
xmlns="http://hp.com/umb/config">
    <flowServices>
        <flow name="UcaStaticForwarderFlow" type="Static"
collectorClass="com.hp.uca.expert.mediation.adapter.UcaStaticCo
llector">
        </flow>
        <flow name="UcaStaticEventForwarderFlow" type="Static"
collectorClass="com.hp.uca.expert.mediation.adapter.UcaStaticCo
llector">
        </flow>
    </flowServices>
</adapter>
```

1.1.2.2 UMB forwarder

One of the roles of the value packs is to forward correlation results (whatever their types: Alarms, Trouble tickets, events…) to some other applications.

From a scenario this is done by using an UMBForwarder object that makes the link between the scenario and the UCA-EBC flow service as defined in the `AdapterConfiguration.xml` file.

An UMBEventForwarder object can be easily created by requesting its creation from the value pack's Spring context (`context.xml` in the valuepack configuration directory).

Here is an example of UMBEventForwarder creation:

```
<bean name="mediationEventForwarder"
class="com.hp.uca.expert.event.UMBEventForwarder">
    <constructor-arg index="0">
        <value>UcaStaticEventForwarderFlow</value>
    </constructor-arg>
</bean>
```

The UMBEventForwarder object is created with an argument which is the name of the static flow as it is define in the UCA-EBC `AdapterConfiguration.xml` file.

Then from a rule file, this UMBEventForwarder object can be used as follow:

1. Define the object in the rule file 'global section'
2. Use the UMBEventForwarder push() method to forward an event to the bus.

Example of rule forwarding an event to the bus:

```
package com.hp.uca.expert.vp.alarmforwarder;

#list any import classes here.
import com.hp.uca.expert.event.EventForwarder;
import com.hp.uca.expert.event.Event;
import com.hp.uca.expert.x733alarm.PerceivedSeverity;
import com.hp.uca.expert.util.MessageFileHandler;
import java.util.ArrayList;
import com.hp.uca.expert.scenario.Scenario;
import com.hp.uca.common.trace.LogHelper;
import com.hp.uca.expert.flag.Flag;
import
com.hp.uca.expert.testmaterial.AbstractJunitIntegrationTest;

#declare any global variables here
global Scenario theScenario;
global EventForwarder mediationEventForwarder;

# Forward any event received
rule "Forward any event received"
no-loop
       when
          $event : Event()
       then
       LogHelper.enter(theScenario.getLogger(),
drools.getRule().getName());

       // Forward the event to ne new Mediation
       mediationEventForwarder.push($event);

       // Retract the event
       theScenario.getLogger().info("Retracting: \n"+
$event.toFormattedString());
       theScenario.getSession().retract($event);

       LogHelper.exit(theScenario.getLogger(),
drools.getRule().getName());
end
```

1.1.3 Defining Dynamic UMB Flows

For dynamic Flows the collectorClass must be set to:
`com.hp.uca.expert.mediation.adapter.UcaDynamicCollector`

No flow parameters need to be defined. When a dynamic is initialized an UMBAlarmForwarder is automatically created.

Here is an example of Dynamic Flow Service definitions for UCA-EBC:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<adapter name="UCA-EBC" version="1.0"
xmlns="http://hp.com/umb/config">
    <flowServices>
        <flow name="DynFlow" type="Dynamic"
collectorClass="com.hp.uca.expert.mediation.adapter.UcaDynamicC
ollector">
        </flow>
    </flowServices>
</adapter>
```

1.1.4 Defining DB Collector UMB Flows

To be able to retrieve alarms from UCA DB through UMB Flows, the collectorClass must be set to: `com.hp.uca.expert.mediation.adapter.UcaDbCollector`

The UcaDbCollector flow is an UcaDynamicCollector flow.

The parameters are "vp", "notifier", "summarize", "eligibilityScope", where:

- vp and notifier are used to retrieve the dbNotifier bean
- summarize is a Boolean used to retrieve all eligible alarms at flow startup
- eligibilityScope is used to select which alarms should pass through the flow

The UCA DB Flow is already configured in default adapter configuration file delivered with UCA-EBC. It is named “DB-Flow”.

1.1.5 Executing DB Actions through UMB

To be able to perform some actions on alarms stored in DB through UMB, the actionClass to use is: `com.hp.uca.expert.mediation.adapter.UcaDbActions`

The UCA DB Actions are already configured in default adapter configuration file delivered with UCA-EBC. You can do 3 actions:

- Terminate an alarm
- Clear an alarm
- Acknowledge an alarm

1.1.6 Executing actions on remote UMB Adapters

The UCA EBC action framework hasn’t changed with the introduction of UMB and UMB actions, i.e. the way to execute actions from UCA EBC value packs is the same.

The changes are only apparent in the following UCA EBC configuration files:

- In the `uca-ebc.properties` file, the `use.new.generation.adapter` property has to be set to `true` if you want to use UMB actions

- In the `ActionRegistry.xml` configuration file, UMB action references can now be defined alongside NOM action references. Both UMB and NOM actions can be used within the same instance of UCA EBC:
    - actions defined from NOM action references will be executed on the NOM framework
    - actions defined from UMB action references will be executed on the UMB framework

The way to define NOM action references in the `ActionRegistry.xml` configuration file hasn't changed. You can however now also define UMB action references in the same file.

UMB Action references are defined inside a `<UMBActions>... </UMBActions>` XML element. For example:

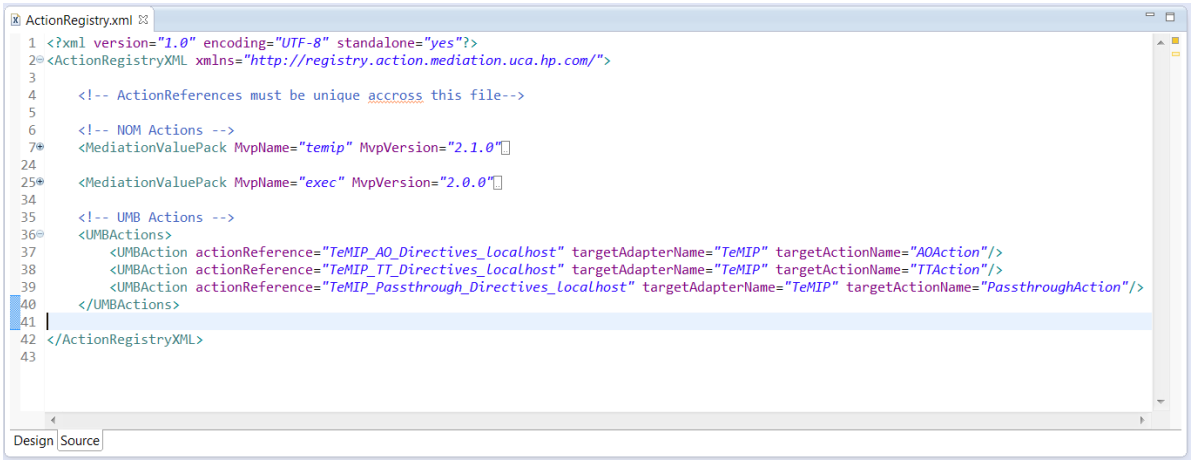

Figure 1 Defining UMB action references in the ActionRegistry.xml file

Please make sure that action references are unique in the `ActionRegistry.xml` configuration file, regardless of whether they are NOM or UMB action references. If some action references are not unique, UCA EBC won't start.

1.2 Upgraded NOM Channel Adapter

UCA EBC CA compatible with NOM V7.2 is now using NOM patched ActiveMQ library (version 5.9.0-nom-6, the same version that UCA EBC Server uses too).

The Channel Adapter has be extended to support actions coming from NOM. This will allow for the creation, deletion and resynchronization of a UCA EBC mediation flow (this will be used by UOC integration).

1.3 Enhanced uca-ebc-injector to support events

The uca-ebc-injector Command Line Interface (CLI) used to only be able to inject alarms.

The uca-ebc-injector CLI has now been enhanced to read files containing any kind of event and to inject them into UCA-EBC.

Basic example

```
uca-ebc-injector –f Eventsfile.xml
```

Below is a sample of an EventsFile.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Events xmlns="http://hp.com/uca/expert/event">

    <EventBoxBase xmlns="http://hp.com/uca/expert/event"
    eventClassName="myEvent Class Name">
      <eventString><![CDATA[
          <myEvent>
           <identifier>1</identifier>
           <value>23</value>
          </myEvent>
                                                    ]]>
      </eventString>
    </EventBoxBase>

    <EventBoxBase xmlns="http://hp.com/uca/expert/event"
    eventClassName="myOtherEvent Class Name">
      <eventString><![CDATA[
          <myOtherEvent>
           <identifier>B</identifier>
           <eventTime>13:00:00</eventTime>
           <value>start</value>
          </myEvent>
                                                    ]]>
      </eventString>
    </EventBoxBase>

</Events>
```

1.4 New API in the JUnit test framework to send events

The UCA-EBC JUnit test framework has been enriched: Value Pack developers can now inject events (other than alarms) to test their ValuePack.

Below is a snippet of a sample JUnit test file showing the still valid sendAlarm() API as well as the new sendEvents() API.

```
public class myJUnitTest extends AbstractJunitIntegrationTest {

[...]
@Test
  @DirtiesContext()
  public void test() throws Exception {
        LogHelper.enter(log, "test()");
[...]


 /*
  * Send alarms
  */
        getProducer().sendAlarms(ALARM_FILE);
```

```
 /*
```

```
 * Send events listed in file EVENTS FILE
 */
      getEventsProducer().sendEvents(EVENTS_FILE);
```

An example EVENTS_FILE can be found in section 1.3 of this document

1.5 Deprecated API removed

The following deprecated API (since 3.2) was removed:

Class com.hp.uca.expert.event.BasicEvent:

Method : getMappings()

Replaced by:
com.hp.uca.expert.instancemapper.MapperUtils.doMapping()

Class com.hp.uca.expert.event.BasicEvent:

Method : addOrchestraDataInScenario()

Replaced by:
com.hp.uca.expert.orchestra.OrchestraUtils.addOrchestraDataInScenario()

Class com.hp.uca.expert.alarm.AlarmCommon:

Method : getMappings()

Replaced by:
com.hp.uca.expert.instancemapper.MapperUtils.doMapping()

1.6 Support of LDAP authentication

Optional LDAP authentication mechanism has been added current login mechanism.

This covers

- simple authentication using LDAP standard protocol
- simple authentication using LDAP secured protocol + trusted certificate
- remote and local servers (Windows, Linux)

1.7 Miscellaneous security related enhancements

Additionally, the following security related enhancements have been implemented to solve

- cross site scripting, malicious script hijacking,
- authentication bypass,
- weak session management,
- (partly) no client/server side validation,
- user enumeration,

- directory listing enabled,
- anonymous login enabled,
- username and password policy is not properly implemented

UCA also now provides an HTTP POST API (based on GWT library) to retrieve authenticated user and role.

1.8 Bug fixing

Please refer to Chapter 2 “Fixed Problems”.

Chapter 2

Fixed Problems

The following problems/enhancements were fixed/implemented in this release

Reference Severity	Component	Description	Comment
CR#12218 Medium	GUI	It should be possible to add a text description at pattern level	Enhancement Implemented
CR#12220 Medium	GUI	Cannot have a tag value looking like an xml tag	Fixed
CR#12229 Medium	GUI	In Filter Builder view, Filter statement edition should propose configurable list of values (like for tags)	Enhancement Implemented
CR#12232 Medium	GUI	When copy&paste a pattern in a topFilter tab 'Problem_XXX', it switches back into the first top Filter tab 'Problem_AAA'	Fixed
CR#12233 Medium	GUI	In Filter Builder view and pattern edition, several actions loose the edition context	Fixed
CR#12235 Medium	GUI	In the Filter Builder view and pattern edition, new condition may modify previous ones	Fixed
CR#12236 Medium	GUI	In the Filter Builder view and pattern edition, new tag edition may modify previous ones	Fixed
CR#12237 Medium	GUI	In the Filter Builder view and pattern edition, the display mode should always be visible	Fixed
CR#12239 Medium	GUI	In the Filter Builder view and pattern edition, some menus should better be renamed for a better understanding	Enhancement Implemented
CR#12823 Medium	Channel Adapter	Simplify and unify Camel route tracing enablement for all channel adapters and provide up-to-date instructions	Enhancement Implemented
CR#12886 Medium	UMB Adapter	UCA NGMediationFlow (actionRegistry) schema enhancement request	Enhancement Implemented
CR#12890 Medium	Packaging	Two zookeeper libraries delivered in UCA Server kit (one from Ne04J and another one from UMB)	Fixed
CR#12893 Medium	Server	NPE at Actionregistry reload	Fixed
CR#12896 Medium	UMB Adapter	default values for UCA UMB adapter properties are not correct	Fixed
CR#12965 Low	Server	Bad warnings logged due to instant-on license	Fixed
CR#12968 Low	Toolkit	ant should be in $PATH	Fixed

Reference Severity	Component	Description	Comment
CR#12982 Medium	Toolkit	Topology aware scenarios (non Inference Machine) generated with eclipse plugin are missing an enum inside the ExtendedTopoAccess class.	Fixed
CR#12985 Low	Toolkit	persistence-example VP is delivered with version 3.2-SP6-SNAPSHOT in MR dev kit.	Fixed
CR#13000 Medium	Server	Support multiple replies when iterator is absent	Enhancement Implemented
CR#13040 Medium	Server	UCA-EBC takes a long time to stop while some flow connections are retried	Fixed
CR#13057 High	Server	UMB <-> NOM flow exception when resynchronizing the whole VP from Admin GUI	Fixed
CR#13063 Medium	GUI	Button Resynchronize is displayed even if UMB Flow is not started	Fixed
CR#13145 Medium	Server	FilterTags are randomly processed when overlapping filters are defined	Enhancement Implemented
CR#13161 Medium	Toolkit	ActionRegistry.xml file contains unfiltered properties	Fixed
CR#13165 Low	Server	ProblemDetection_filtersTags.xml is not loaded when running 'uca-ebc-admin -rc'	Fixed
CR#13180 Medium	Server	Trigger dataload from uca-ebc-admin command line	Enhancement Implemented
CR#13184 High	Server	UMB for IM: Need to introduce more parameters as ActionReference in the configuration	Enhancement Implemented
CR#13193 High	Server	Java 8 support	Enhancement Implemented
CR#13194 High	GUI	Custom Specific config scrambled	Fixed
CR#13214 Medium	Packaging	Installing UCA-EBCDEVKIT with no user uca created gives too many errors	Fixed
CR#13233 Medium	GUI	Filter Builder editor: not possible to add a value not part of an enum (when enum type is used)	Enhancement Implemented
CR#13243 Medium	Documentation	Document that ONLY 64 bit windows systems are supported	Fixed
CR#13251 High		NPE when filtering on dateStatement	Fixed
CR#13259 Urgent	Server	Can no more display configuration due to ClassCastException: com.hp.uca.expert.config.*JaxbAccessorF* cannot be cast to com.sun.xml.internal.bind.v2.runtime.reflect.Accessor	Fixed
CR#13316	GUI	Deadlock in GraphDisplay	Fixed
CR#13378 High	Toolkit	UCA-DEVTOOKIT Impossible to build an example due to missing umb-fwk jar file	Fixed
CR#13389 Urgent	Server	Allow Object/Serializable as a value for a key in UCA actions	Fixed
CR#13415 High	Packaging	not able to install UCA-EBCTOPO-V3.2-00B kit because of a conflict on file GraphDisplayProfiles.xml	Fixed
CR#13416 High	GUI	out of memory sometimes while trying to visualize a graph in the UCA-EBC GUI	Fixed

Reference Severity	Component	Description	Comment
CR#13426 High	GUI	Bad org.xml.sax.SAXParseException logged as ERROR when displaying VP configuration through GUI	Fixed
CR#13430 Medium	GUI	GUI security issue Case1- Script is entered in Username field	Fixed
CR#13432 Medium	GUI	GUI Security issue : Directory Listing Enabled	Fixed
CR#13433 Medium	GUI	GUI Security Issue : Username and Password Policy is not properly implemented	Fixed
CR#13434 Medium	GUI	GUI Security issue : User enumeration	Fixed
CR#13436 Medium	GUI	GUI Security issue : Authentication bypass using parameter manipulation	Fixed
CR#13439 Medium	GUI	GUI Security issue : Case 2- Request submited	Fixed
CR#13454 Medium	Server	Bad ERROR logged when stopping UMB flow that has never started	Fixed
CR#13484 Medium	Server	UCA EBC server crashes if user deploys and starts a value pack immediately after server startup	Fixed
CR#13515 Low	Server	Bad WARNing logged at UCA startup: Instance already exist	Fixed

Table 2 - Fixed Problems in UCA for EBC V3.3

Chapter 3

Known Problems

This section lists problems discovered during the product test campaign and that still have to be fixed.

Reference / Severity	Component	Description	Solution/Suggested workaround
CR#9929 Low	Toolkit	Skeleton: Error in rules when creating a new VP from the skeleton using the UCA plugin	Deleting and re-importing the project fixes the problem. No Problem if the plugin is not installed
CR#11444 Medium	GUI	When using the Topology graph visualisation tool from The GUI a session is created on the server side. When the browser is disconnected, the session is not fully destroyed leading to memory consumption	There is no real workaround to this problem unless stopping/re-starting the UCA-EBC server. This has a minimal memory impact though.
CR#11955 Medium	GUI	Multiple filter files not fully supported at the GUI	This can be used without UI.
CR#12335 Medium	Server	Null Pointer Exception in Problem Detection Value Pack under heavy stop/start combinations	Under rare Stop/start/stop/start conditions (but not for Production Use Case)
CR#12983 Medium	Toolkit	Eclipse plugin: Topology aware scenario with templated rules not working.	If you chose Topology Aware scenario, the templated rules option should be unselected
CR#13555 Medium	GUI	GUI may display Failure during Notification reception -- error is:	Workaround: - Press F5 to reload the GUI - log in
CR#13580 Medium	GUI	Japanese localization of UCA EBC GUI improperly shows _JP### suffixes to translated strings	Will be fixed during extension of Japanese translation for 3.3 (will be part of a patch)
CR#13581 Medium	Server	NullPointerException while stopping UCA EBC (at com.hp.uca.expert.vp.internal.ActiveValuePack.getScenarioRunningNumber(ActiveValuePack.java:626))	No functional impact, nothing to do
CR#13553 Low	GUI	It is not possible to stop a mediation flow that is in Failed state	Will be investigated for a future version
CR#13579 Medium	GUI	UCA EBC GUI: Flow status explanation says "Inactive" when status is "Active"	Status is the valid information (explanation is secondary)

Table 3 - Known Problems

Note that even though this is not an UCA EBC issue, we have noticed the following user error experienced by some projects:

When 'cleaning' the `/var/opt/UCA-EBC/instances/default/logs/` make sure you use

`rm /var/opt/UCA-EBC/instances/default/logs/*.log` command and **NOT** `rm /var/opt/UCA-EBC/instances/default/logs/*`

Chapter 4
Existing Value Packs migration steps

4.1 Non-Problem Detection and Non-Inference Machine Value Packs

- from a command line session, go to your project directory ("my_VP_project"), and run 'ant eclipse'
- from eclipse, refresh and clean your project.

4.2 Problem Detection or Inference Machine value packs

Same steps as above (4.1), and extra steps described in the Release Notes document of Inference Machine product.