HP UCA Automation

UCA Automation

Version 1.0

Administrator
And
User Interface Guide

Edition: 1.1

November 2013

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 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 2007®, Windows XP®, and Windows 7® 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.

EnterpriseDB® is a registered trademark of EnterpriseDB.

Postgres Plus® Advanced Server is a registered U.S. trademark of EnterpriseDB.

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

This guide describes how to use the web-based user interface of the UCA Automation. The administrators, operators and observers have different privileges provided through the user interfaces.

Product Name: UCA Automation

Product Version: 1.0

Kit Version: V1.0

Intended Audience

Here are some recommendations based on possible reader profiles:

- Solution Developers
- Software Development Engineers
- Solution Administrator
- Solution Operators

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 Automation 1.0	Linux Red Hat Enterprise Linux Server release 6.4

Table 1 - Software versions

As the provided user interface is web based, the rendering of some components may be slightly different depending on the browser used.

However the described functionalities should be identical on any supported browser.

For the UCA Automation 1.0, the supported browser is Microsoft Internet Explorer 9.0.

Typographical Conventions

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] *HP UCA Automation V1.0 - Installation Guide V1.1.pdf*

[R2] *HP UCA Automation V1.0 – Integrator's Guide V1.1.pdf*

[R3] *DeploymentManager.pdf (HPSA)*

[R4] NOM Installation and Configuration Guide

[R5] NOM HPSA Channel Adapter Installation guide

[R6] NOM UCA Automation Console Channel Adapter Installation guide

[R7] NOM UCA EBC Channel Adapter Installation guide

[R8] NOM TEMIP Channel Adapter Installation guide

Support

Please visit our HP Software Support Online Web site at www.hp.com/go/hpsoftwaresupport 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

Introduction

This guide describes the web-based user interface for the UCA Automation solution, which covers the functionality of the following roles:

- Administrator
- Operator
- Observer

This guide also describes the administrative tasks that must be carried out in the UCA Automation solution, and covers the following parts:

- Populate the HP Service Activator inventory data
- Populate the Neo4J graph DB data
- Launching test on UCA Automation from TeMIP Client
- Internationalization (I18N)

1.1 Software pre-requisites

The following browsers are supported in UCA Automation V1.0

Browser	Version
Microsoft Internet Explorer	9.0

Table 2 – Supported Web Browsers

1.2 Launching the UCA Automation User Interface

Once the UCA Automation application is started, the user interface can be launched, by accessing the following URL:

http://<hostname or IP address>:<port #>/UCAAutomation

where

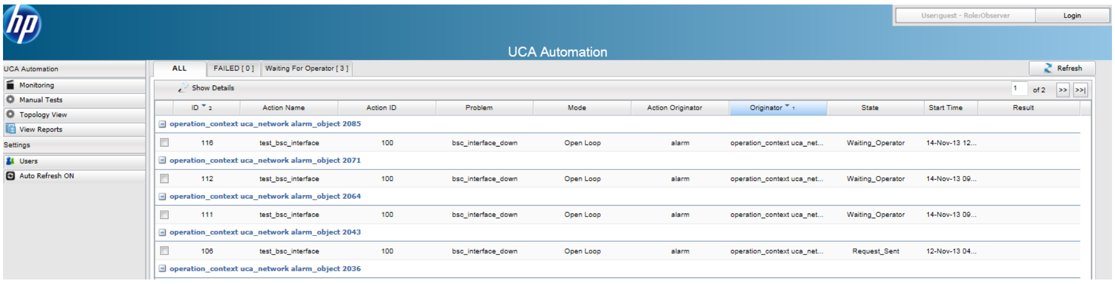
<hostname or IP address> should be replaced by the actual hostname (full DNS name) or IP address of the UCA Automation Server system. If UCA Automation server is running on your local host, you can use ‚localhost' as the name of the host to connect to using your web browser.

<port #> is the port number for UCA Automation User Interface, 9080 by default.

1.3 UCA Automation User Interface Layout

The following picture shows the UCA Automation User Interface main screen, representing the Monitoring page. This is the default screen for the Observer role.

The various Actions, and their State's are displayed in this page. The Actions are grouped by Originator and sorted by (1. Originator (Descending) and 2. Id(Descending))



	ID 2	Action Name	Action ID	Problem	Mode	Action Originator	Originator 1	State	Start Time	Result
operation_context uca_network alarm_object 2085										
□	116	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_net...	Waiting_Operator	14-Nov-13 12...	
operation_context uca_network alarm_object 2071										
□	112	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_net...	Waiting_Operator	14-Nov-13 09...	
operation_context uca_network alarm_object 2064										
□	111	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_net...	Waiting_Operator	14-Nov-13 09...	
operation_context uca_network alarm_object 2043										
□	106	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_net...	Request_Sent	12-Nov-13 04...	
operation_context uca_network alarm_object 2036										

Figure 1 - UCA Automation User Interface main view

The various roles available in the UCA Automation User Interface are Observer, Operator and Administrator.

In order to get access to the UCA Automation User Interface, get in touch with the UCA Automation administrator, who will create a user and associate a role with this user.

On the left panel, the various operations possible are User Management, Monitoring, Manual Tests, Topology View and View Reports. Following chapters discuss each operation in details.

Chapter 2

UCA Automation Settings

2.1 User logging

Different operations in the UCA Automation User Interface are accessible to users with different roles.

Following are the various operations allowed for the given role

Operation	Observer	Operator	Administrator
User Management			√
Monitoring	√	√	√
Manual Tests	√	√	√
Topology Views	√	√	√
View Reports	√	√	√

Table 3 – roles and privileges

NOTE: Observer only has the view privileges.

When the UCA Automation web interface is started, by default, ‘**guest**’ user with role ‘**Observer**’ is logged in.

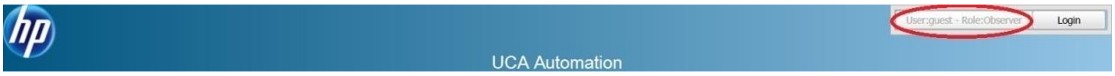


Figure 2 - Default login role

Log-in to the interface as another user is performed by click on “Login” link on upper right corner of the page.

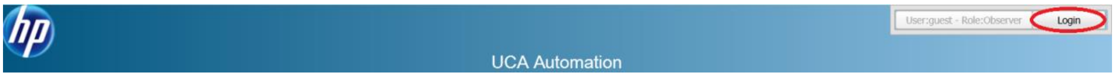


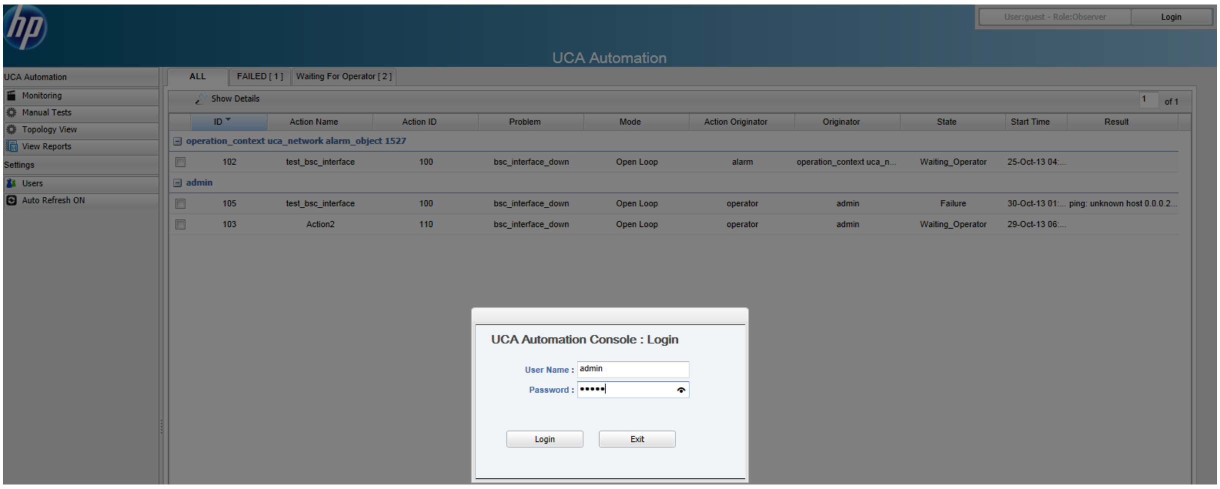
Figure 3 – Login panel access

At installation time, the ‘admin’ user is created with Administrator privileges. The credentials are

User Name: **admin**

Password: **admin**

Login to the application as admin user to create additional users.



ID	Action Name	Action ID	Problem	Mode	Action Originator	Originator	State	Start Time	Result
operation_context uca_network alarm_object 1527									
102	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_n...	Waiting_Operator	25-Oct-13 04:...	
admin									
105	test_bsc_interface	100	bsc_interface_down	Open Loop	operator	admin	Failure	30-Oct-13 01:...	ping: unknown host 0.0.0.2...
103	Action2	110	bsc_interface_down	Open Loop	operator	admin	Waiting_Operator	29-Oct-13 06:...	

UCA Automation Console : Login

User Name : admin

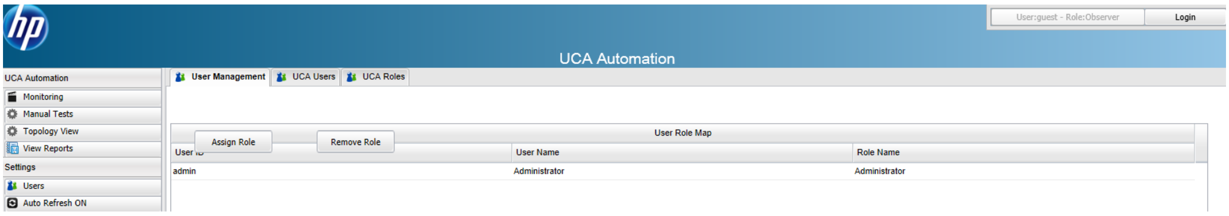
Password : •••••

Figure 4 – Login panel

2.2 User Management

A user can be added, modified or deleted only by a user with Administrator role.

The user management panel can be opened by selecting Settings → Users menu options.



User Role Map

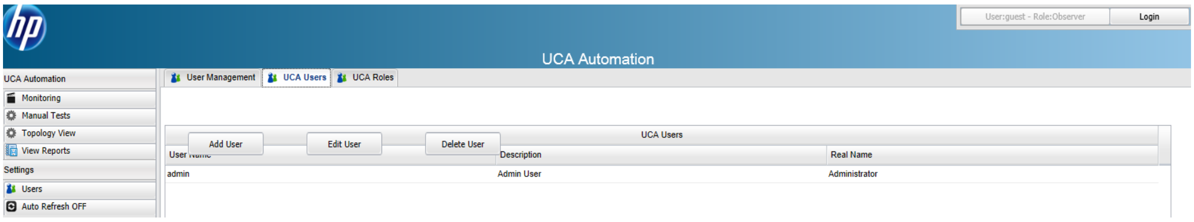
User ID	User Name	Role Name
admin	Administrator	Administrator

Figure 5 – User management panel

The User Management tab displays the different users and the roles that they are associated with.

Roles are prefined (as part of post-install script to be restricted into Administator, Operator and Observer, which cannot be edited, deleted or new can be inserted)

Select the Users → UCA Users menu option and Add User, Edit User , and Delete User to add, change or remove a user.

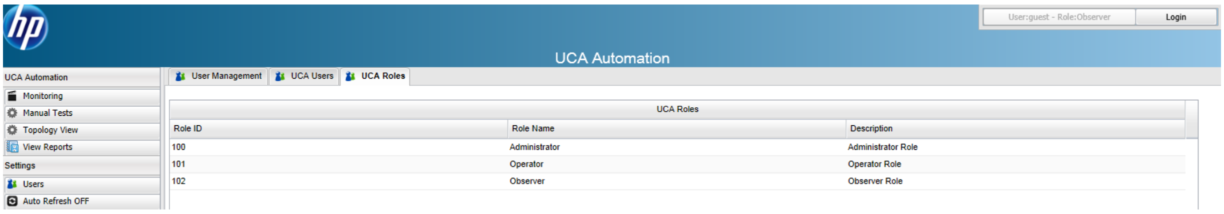


UCA Users

User Name	Description	Real Name
admin	Admin User	Administrator

Figure 6 – UCA Automation users

The different roles available in UCA Automation are as shown in the figure below.



UCA Roles

Role ID	Role Name	Description
100	Administrator	Administrator Role
101	Operator	Operator Role
102	Observer	Observer Role

Figure 7 – UCA roles

2.3 Auto Refresh

Records in the monitoring page are displayed almost at runtime, when the new event occurs.

The page is automatically refreshed to fetch the new records, at an interval defined by the */opt/UCA-ATM/webapp/UCAAutomation.war/WEB-INF/classes/UCAAutomation.properties*, UI_AUTO_REFRESH_INTERVAL attribute. By default the auto refresh timer is set to 10 seconds (10000ms). User is also provided with a Refresh button on the right of the monitoring page for usability.

In order to improve user experience, once the record of interest is available in the UI, Auto Refresh can be turned OFF to avoid losing the record selections made.

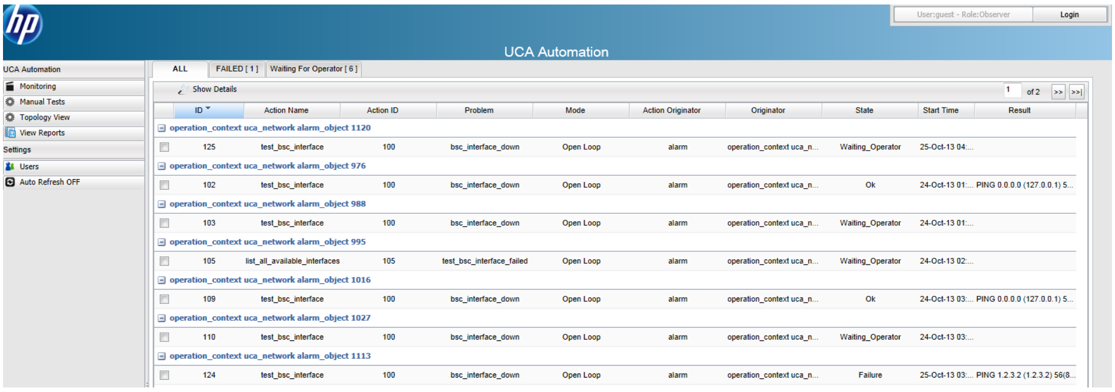
The Auto Refresh value can be toggled by clicking on Settings → Auto Refresh option.

Chapter 3

Monitoring

Status of each test performed can be viewed in this panel.

Select the The user management panel can be opened by selecting UCA Automation → Monitoring menu options.



	ID	Action Name	Action ID	Problem	Mode	Action Originator	Originator	State	Start Time	Result
operation_context uca_network alarm_object 1120										
□	125	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_n...	Waiting_Operator	25-Oct-13 04:...	
operation_context uca_network alarm_object 976										
□	102	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_n...	Ok	24-Oct-13 01:...	PING 0.0.0.0 (127.0.0.1) 5...
operation_context uca_network alarm_object 988										
□	103	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_n...	Waiting_Operator	24-Oct-13 01:...	
operation_context uca_network alarm_object 995										
□	105	list_all_available_interfaces	105	test_bsc_interface_failed	Open Loop	alarm	operation_context uca_n...	Waiting_Operator	24-Oct-13 02:...	
operation_context uca_network alarm_object 1016										
□	109	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_n...	Ok	24-Oct-13 03:...	PING 0.0.0.0 (127.0.0.1) 5...
operation_context uca_network alarm_object 1027										
□	110	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_n...	Waiting_Operator	24-Oct-13 03:...	
operation_context uca_network alarm_object 1113										
□	124	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_n...	Failure	25-Oct-13 03:...	PING 1.2.3.2 (1.2.3.2) 56(8...

Figure 8 – Monitor view

The tab option "All" shows the State of all the tests that are currently being run, and also the Failed tests.

If the number of tests increases, it is easier to separately view the Failed tests in "FAILED[x]" tab, and tests where operator's inputs are required, they can be viewed in "Waiting For Operator[x]" tab. The value 'x' represents the number of Actions in these tabs.

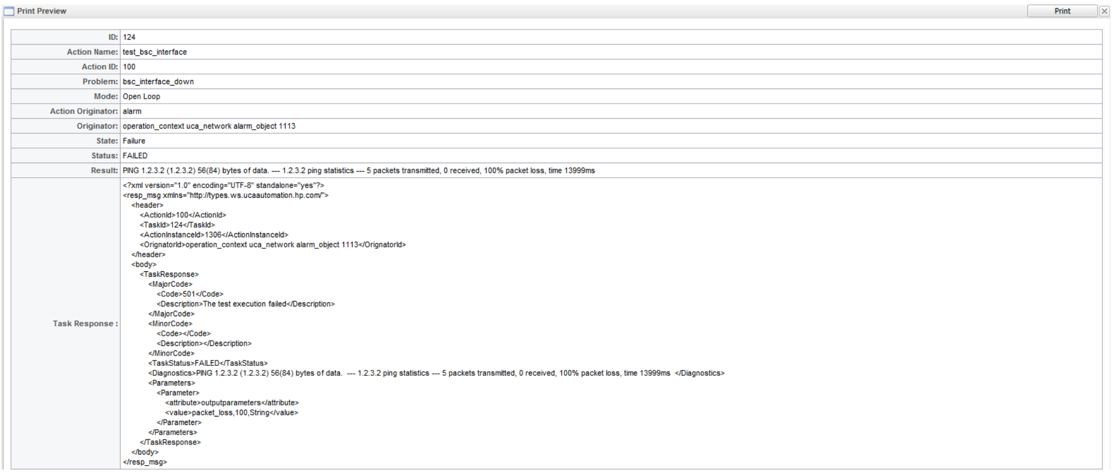
The default view shows the State of the Action taken on a given Problem. Note that if the Status is Successful it is not listed here – they can be viewed in Reports menu option.

Field	Description
ID	Task ID
Action Name	Diagnostic Action taken for a given problem
Action ID	Diagnostic Action ID
Problem	The problem symptom
Mode	Diagnostic Action mode Open/Closed loop
Action Originator	alarm/operator
Originator	Alarm Id /Operator Id
State	Waiting_Operator/Request_Sent/In_Progress, Failure, Ok
Start Time	When was the task started
Result	Diagnostic Action results

Table 4 – Monitoring

Any record can be selected, by checking the check box, and its details can be viewed by clicking on the Show Details button.

The result of a failed test can be viewed by selecting the Failed record, as shown in the following figure.



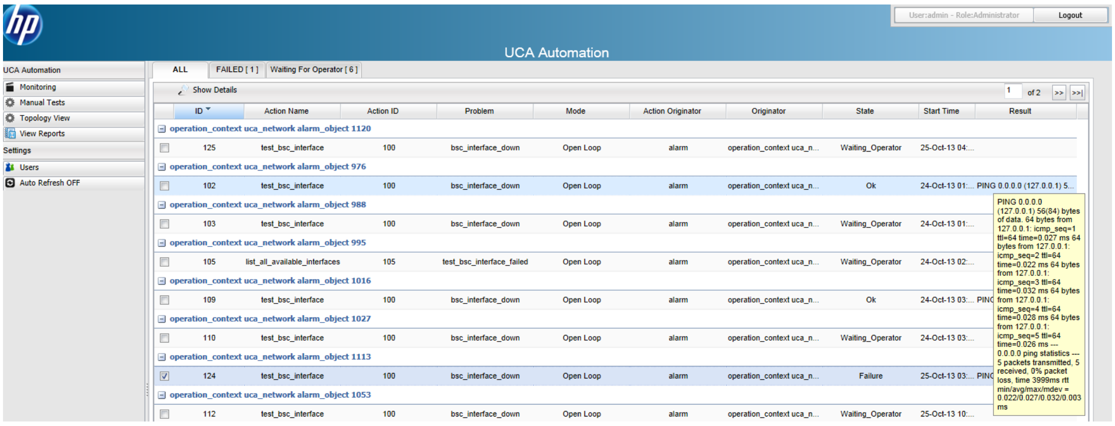
ID:	124
Action Name:	test_bsc_interface
Action ID:	100
Problem:	bsc_interface_down
Mode:	Open Loop
Action Originator:	alarm
Originator:	operation_context uca_network alarm_object 1113
State:	Failure
Status:	FAILED
Result:	PING 1.2.3.2 (1.2.3.2) 56(84) bytes of data. --- 1.2.3.2 ping statistics --- 5 packets transmitted, 0 received, 100% packet loss, time 13999ms

Task Response :

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<resp_msg xmlns="http://types.ws.ucaautomation.hp.com/">
  <header>
    <ActionId>100</ActionId>
    <TaskId>124</TaskId>
    <ActionInstanceId>1306</ActionInstanceId>
    <OrignatorId>operation_context uca_network alarm_object 1113</OrignatorId>
  </header>
  <body>
    <TaskResponse>
      <MajorCode>
        <Code>501</Code>
        <Description>The test execution failed</Description>
      </MajorCode>
      <MinorCode>
        <Code></Code>
        <Description></Description>
      </MinorCode>
      <TaskStatus>FAILED</TaskStatus>
      <Diagnostics>PING 1.2.3.2 (1.2.3.2) 56(84) bytes of data.  --- 1.2.3.2 ping statistics --- 5 packets transmitted, 0 received, 100% packet loss, time 13999ms  </Diagnostics>
      <Parameters>
        <Parameter>
          <attribute>outputparameters</attribute>
          <value>packet_loss,100,String</value>
        </Parameter>
      </Parameters>
    </TaskResponse>
  </body>
</resp_msg>
```

Figure 9 – Monitor show details

The Result can also be viewed by highlighting the mouse over the record’s Result cell.



	ID	Action Name	Action ID	Problem	Mode	Action Originator	Originator	State	Start Time	Result
⊟ operation_context uca_network alarm_object 1120										
☐	125	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_n...	Waiting_Operator	25-Oct-13 04:...	
⊟ operation_context uca_network alarm_object 976										
☐	102	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_n...	Ok	24-Oct-13 01:...	PING 0.0.0.0 (127.0.0.1) 5...
⊟ operation_context uca_network alarm_object 988										
☐	103	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_n...	Waiting_Operator	24-Oct-13 01:...	
⊟ operation_context uca_network alarm_object 995										
☐	105	list_all_available_interfaces	105	test_bsc_interface_failed	Open Loop	alarm	operation_context uca_n...	Waiting_Operator	24-Oct-13 02:...	
⊟ operation_context uca_network alarm_object 1016										
☐	109	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_n...	Ok	24-Oct-13 03:...	PING
⊟ operation_context uca_network alarm_object 1027										
☐	110	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_n...	Waiting_Operator	24-Oct-13 03:...	
⊟ operation_context uca_network alarm_object 1113										
☑	124	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_n...	Failure	25-Oct-13 03:...	PING
⊟ operation_context uca_network alarm_object 1053										
☐	112	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca_n...	Waiting_Operator	25-Oct-13 10:...	

PING 0.0.0.0 (127.0.0.1) 56(84) bytes of data. 64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.027 ms 64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.022 ms 64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.032 ms 64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.028 ms 64 bytes from 127.0.0.1: icmp_seq=5 ttl=64 time=0.026 ms --- 0.0.0.0 ping statistics --- 5 packets transmitted, 5 received, 0% packet loss, time 3999ms rtt min/avg/max/mdev = 0.022/0.027/0.032/0.003 ms

Figure 10 – Monitor view result

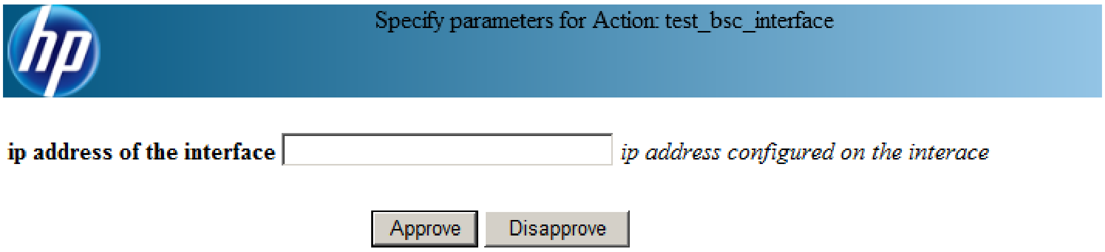
If a test has the state “Waiting for Operator”, it indicates that one or more inputs need to be provided by the operator.

Go to the “Waiting For Operator” tab, select the record of your choice by clicking the check box, and provide inputs to the task by clicking on the “Interact” button.

It opens up a new UI, where appropriate values are keyed in, and can be either Approved or Disapproved.

In case the field had been set with default value in the HPSA inventory, the default value for the field would appear in the text box. If the field is editable, this value can be edited before submitting.

In case the field has been set to be non-editable, value will be displayed, but cannot be edited before submitting.



Specify parameters for Action: test_bsc_interface

ip address of the interface *ip address configured on the interace*

Figure 11 – Monitor interact page

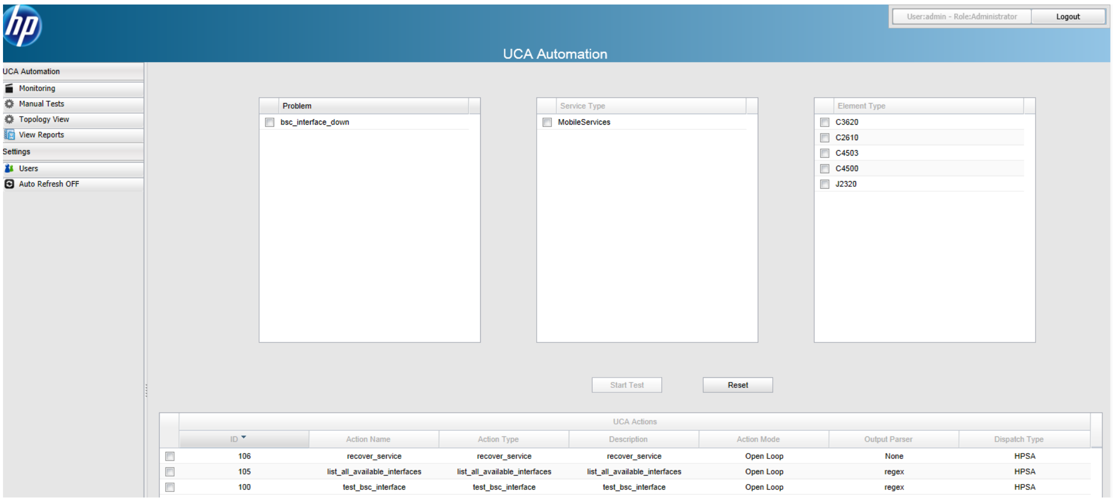
Chapter 4

Manual Tests

An Action (test) can be triggered manually using this interface.

Clicking on UCA Automation → Manual Tests displays a list of Problems, Service Types, Resource Types, and Actions.

Each of these values can be populated using the HP Service Activator inventory system.



	Problem
□	bsc_interface_down

	Service Type
□	MobileServices

	Element Type
□	C3620
□	C2610
□	C4503
□	C4500
□	J2320

UCA Actions

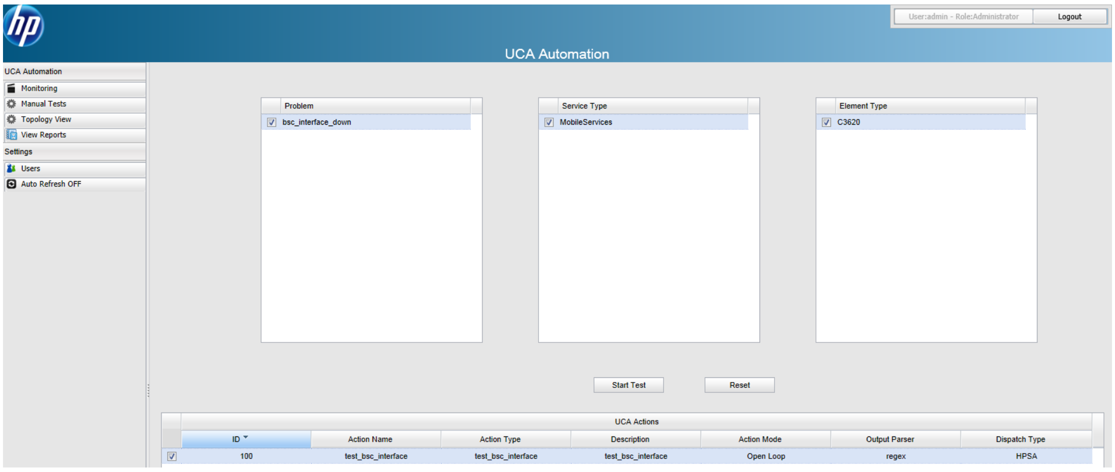
	ID	Action Name	Action Type	Description	Action Mode	Output Parser	Dispatch Type
□	106	recover_service	recover_service	recover_service	Open Loop	None	HPSA
□	105	list_all_available_interfaces	list_all_available_interfaces	list_all_available_interfaces	Open Loop	regex	HPSA
□	100	test_bsc_interface	test_bsc_interface	test_bsc_interface	Open Loop	regex	HPSA

Figure 12 – Manual Tests main page

An action can be performed on selected resource, for a chosen service type and/or problem.

Select a Problem in the “Problem” table by clicking on the check box. All the associated Service Types for Problem will be listed in the “Service Type” table. Select a Service Type from the available list by clicking on the check box. All the associated Resource Types (Element Types) will be listed in the “Resource Type” table. Select a Resource Type by clicking on the check box.

For the above combination, all the available actions possible will be listed in the “UCA Actions” table. One of the Actions can be selected by clicking on the check box, and a test can be performed by clicking on the “Start Test” button.



Problem
☑ bsc_interface_down

Service Type
☑ MobileServices

Element Type
☑ C3620

UCA Actions

	ID	Action Name	Action Type	Description	Action Mode	Output Parser	Dispatch Type
☑	100	test_bsc_interface	test_bsc_interface	test_bsc_interface	Open Loop	regex	HPSA

Figure 13 – Manual Tests – Selected action for a problem

Once a Test has been started, the progress can be viewed in the Monitor page. Upon clicking on “Start Test” button, the task goes into Request_Sent state, and then to In_Progress state, and then changes to other states.

Details of the various fields in the UCA Actions are as follows:

Field	Description
ID	Action ID
Action Name	Name of the diagnostic action to be performed
Action Type	Recover/resolve, test, audit, read-only-test, internal, escalate
Description	Brief description of the action
Action Mode	Open/Closed loop
Element Type	Resource Type, eg C3620 etc
Output Parser	None/Regular Expression/XPath. The output from the Diagnostic actions can be parsed using either regular expressions or XPath
Dispatch Type	HPSA. Only actions with Dispatch Type HPSA are listed.

Table 5 – Action description

Chapter 5

Topology View

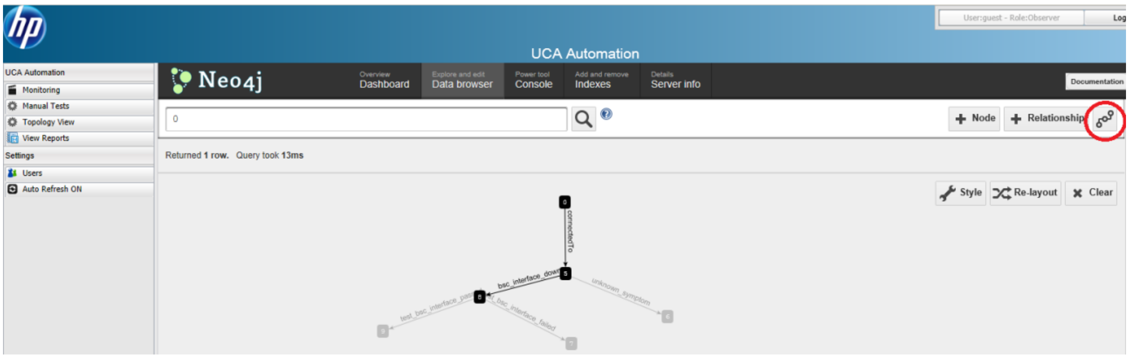
Graph depicting the action path taken for a given test can be viewed using this panel.

In order to launch the Neo4J view, edit the following attributes in */opt/UCA-ATM/webapp/UCAAutomation.war/WEB-INF/classes/UCAAutomation.properties* file to reflect the appropriate values.

neo4j_host=<Neo4J server hostname>

neo4j_port=<Neo4J port>

Click on the UCA Automation → Topology View launches the Neo4J data browser. Click on the icon as marked in red in the figure below to view the graph.



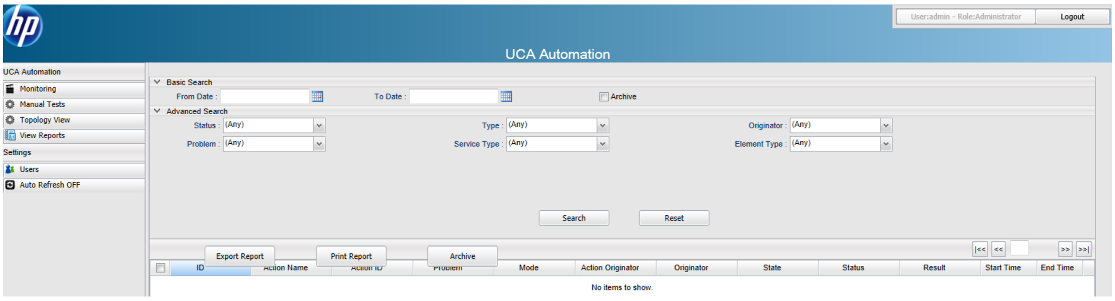
Returned 1 row. Query took 13ms

Figure 14 – Topology View – launch Neo4j data browser

Chapter 6

View Reports

Once a test has been completed, and it reflects the status as Successful or Failure, this test is moved from the Monitor panel to the reports panel.



From Date :

To Date :

□ Archive

Status : (Any)

Type : (Any)

Originator : (Any)

Problem : (Any)

Service Type : (Any)

Element Type : (Any)

No items to show.

Figure 15 – Report search criteria

Reports can be searched based on various criteria, as listed in the following table

Search Field	Description
From Date	If set, records are searched from this date onwards
To Date	If set, records are searched up to this date
Status	PASSED, FAILED, DISAPPROVED, INTERNALERROR
Type	Open Loop/Closed Loop
Originator	Alarm/operator/test
Problem	List of all problems defined in the HPSA inventory
Service Type	List of all service types (domain) defined in the HPSA inventory
Element Type	List of all Element Types defined in the HPSA inventory
Archive	Searches for archived records, if selected

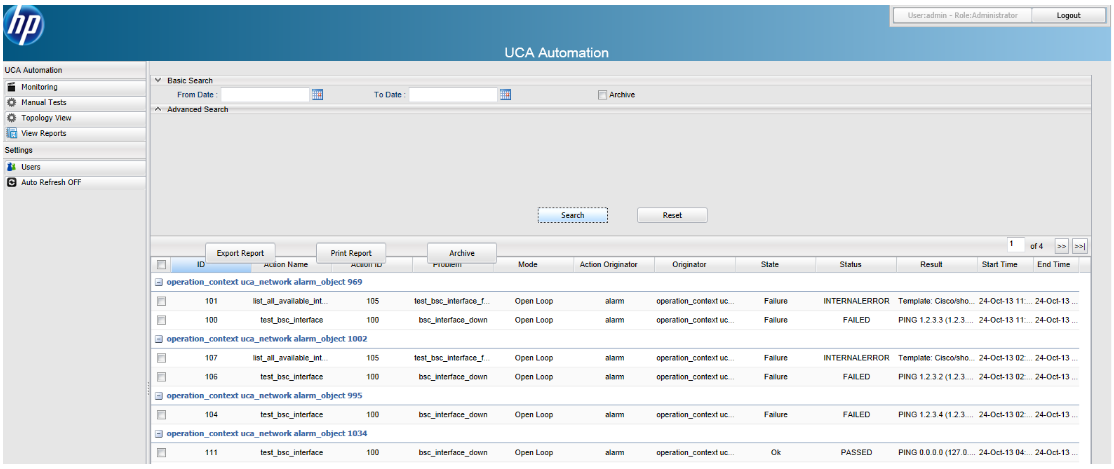
Table 6 – Report search criteria

Report Actions	Description
Search	Search for records with the filter criteria applied. By default, search for all records that are not archived
Export Report	Allows to save the records in CSV format
Print Report	Displays a print preview of the records from

	which the user can print the records
Archive	Archives the selected records. Only enabled when Archive search option is unchecked and logged in with sufficient privileges.
Delete	Selected archived records can be permanently deleted. Enabled when Archive search option is checked.
Reset	Resets the search criteria to default

Table 7 – Report actions

Following figure depicts the search results when performed with default options.



☐ Archive

☐	ID	Action Name	Action ID	Problem	Mode	Action Originator	Originator	State	Status	Result	Start Time	End Time
⊟ operation_context uca_network alarm_object 969												
☐	101	list_all_available_int...	105	test_bsc_interface_f...	Open Loop	alarm	operation_context uc...	Failure	INTERNALERROR	Template: Cisco/sho...	24-Oct-13 11:...	24-Oct-13 ...
☐	100	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uc...	Failure	FAILED	PING 1.2.3.3 (1.2.3....	24-Oct-13 11:...	24-Oct-13 ...
⊟ operation_context uca_network alarm_object 1002												
☐	107	list_all_available_int...	105	test_bsc_interface_f...	Open Loop	alarm	operation_context uc...	Failure	INTERNALERROR	Template: Cisco/sho...	24-Oct-13 02:...	24-Oct-13 ...
☐	106	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uc...	Failure	FAILED	PING 1.2.3.2 (1.2.3....	24-Oct-13 02:...	24-Oct-13 ...
⊟ operation_context uca_network alarm_object 995												
☐	104	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uc...	Failure	FAILED	PING 1.2.3.4 (1.2.3....	24-Oct-13 02:...	24-Oct-13 ...
⊟ operation_context uca_network alarm_object 1034												
☐	111	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uc...	Ok	PASSED	PING 0.0.0.0 (127.0....	24-Oct-13 04:...	24-Oct-13 ...

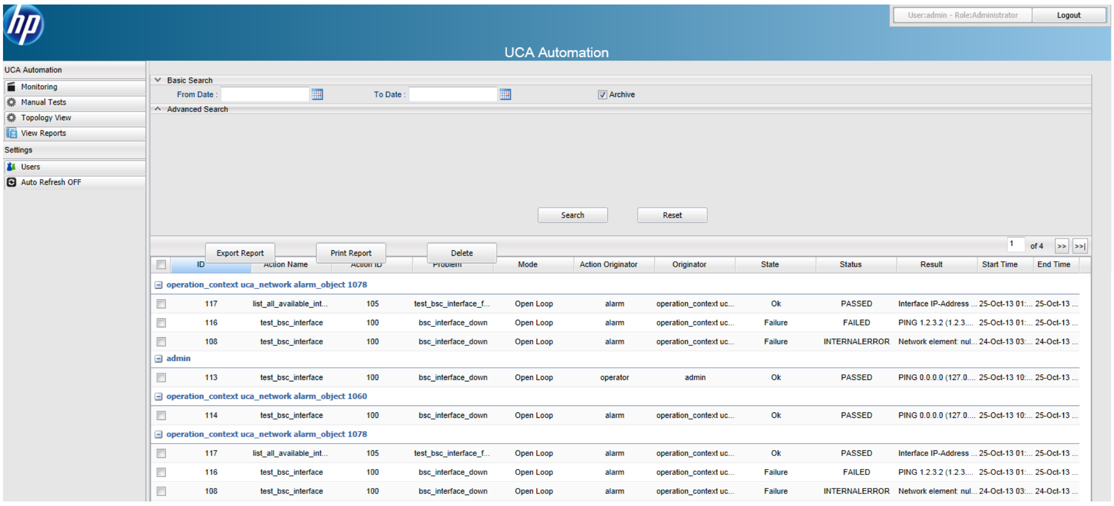
Figure 16 – Report default search

Individual records in the search result can be selected, and can be moved to Archive by clicking on the “Archive” button.

Results can be printed by clicking on the “Print Report” button.

Results can be exported into CSV format by clicking on the “Export Report” button.

Following figure depicts the search results when performed with Archive option enabled. The archived records can be selected and permanently deleted by clicking on the “Delete” button.



☑ Archive

☐	ID	Action Name	Action ID	Problem	Mode	Action Originator	Originator	State	Status	Result	Start Time	End Time
⊟ operation_context uca_network alarm_object 1078												
☐	117	list_all_available_int...	105	test_bsc_interface_f...	Open Loop	alarm	operation_context uc...	Ok	PASSED	Interface IP-Address ...	25-Oct-13 01:...	25-Oct-13 ...
☐	116	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uc...	Failure	FAILED	PING 1.2.3.2 (1.2.3....	25-Oct-13 01:...	25-Oct-13 ...
☐	108	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uc...	Failure	INTERNALERROR	Network element: nul...	24-Oct-13 03:...	24-Oct-13 ...
⊟ admin												
☐	113	test_bsc_interface	100	bsc_interface_down	Open Loop	operator	admin	Ok	PASSED	PING 0.0.0.0 (127.0....	25-Oct-13 10:...	25-Oct-13 ...
⊟ operation_context uca_network alarm_object 1060												
☐	114	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uc...	Ok	PASSED	PING 0.0.0.0 (127.0....	25-Oct-13 10:...	25-Oct-13 ...
⊟ operation_context uca_network alarm_object 1078												
☐	117	list_all_available_int...	105	test_bsc_interface_f...	Open Loop	alarm	operation_context uc...	Ok	PASSED	Interface IP-Address ...	25-Oct-13 01:...	25-Oct-13 ...
☐	116	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uc...	Failure	FAILED	PING 1.2.3.2 (1.2.3....	25-Oct-13 01:...	25-Oct-13 ...
☐	108	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uc...	Failure	INTERNALERROR	Network element: nul...	24-Oct-13 03:...	24-Oct-13 ...

Figure 17 – Report search for archived records

Chapter 7

Populate HP Service Activator Inventory data

HP Service Activator inventory must be populated with the following data:

7.1 Pre-requisites

HP Service Activator V62-1A has been installed, and CRModel and the UCA Automation foundation SP have been deployed.

7.2 Login

Once the HP Service Activator service is started, login to HP Service Activator portal by accessing the following URL:

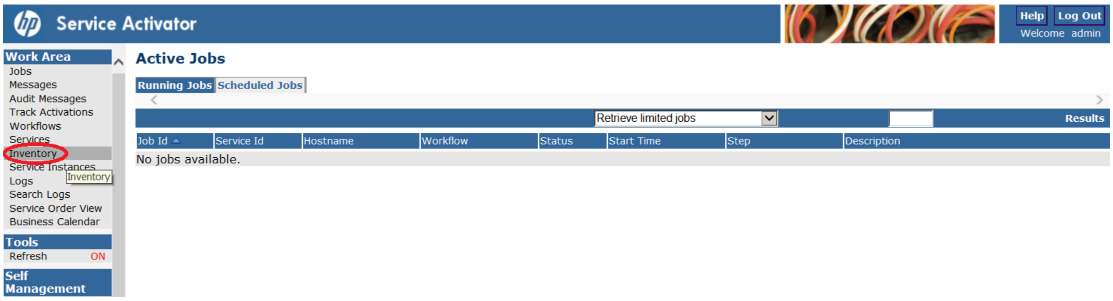
http://<*hostname or IP address*>:<*port #*>/activator/jsp/login.jsp

Here

<hostname or IP address> should be replaced by the actual hostname (full DNS name) or IP address of the HP Service Activator system. If HP Service Activator is running on your local host, you can use ‚localhost' as the name of the host to connect to using your web browser.

<port #> is the port number for HP Service Activator portal, 8080 by default.

Login to the portal using the administrator privileges.



Active Jobs

Job Id	Service Id	Hostname	Workflow	Status	Start Time	Step	Description
No jobs available.							

Figure 18 – HP Service Activator (HPSA) portal

Click on the Inventory Menu option to launch the HPSA Inventory

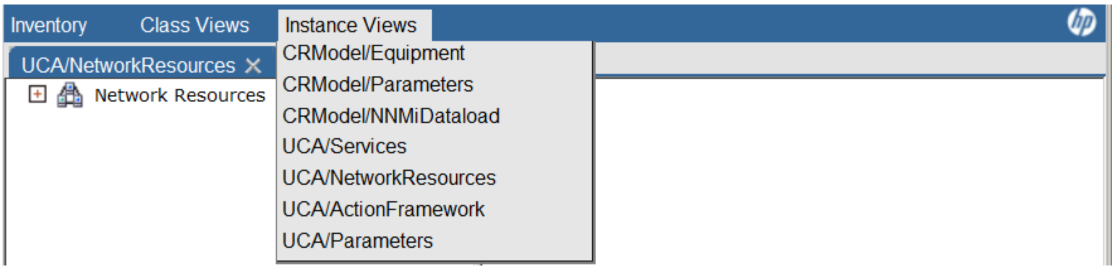


Figure 19 – HPSA inventory

The instances CRModel/Equipment, CRModel/Parameters and CRModel/NNMiDataload represents the inventory model provided by CRModel when HPSA product is installed and CRModel service pack is deployed.

When UCA Automation service pack is deployed, we get the UCA/Services, UCA/NetworkResources, UCA/Actionframework and UCA/Parameters inventory model.

7.3 HP Service Activator Inventory

HPSA Inventory provides a skeleton to create the Resource Types, Resource instances, Problems and Actions as inventory objects. Once they are available, a relationship tree representing the Action to be taken for a given Problem can be producted, that is used by the UCA Automation solution to identify the steps to be followed in the Problem resolution process.

7.3.1 UCA/NetworkResources

UCA/NetworkResources provides options to define Resource Vendor’s, Resource Types, and Resource Instances.

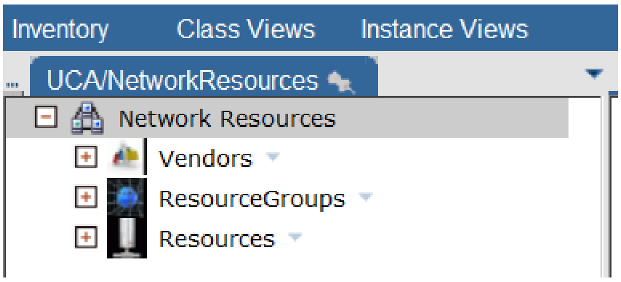
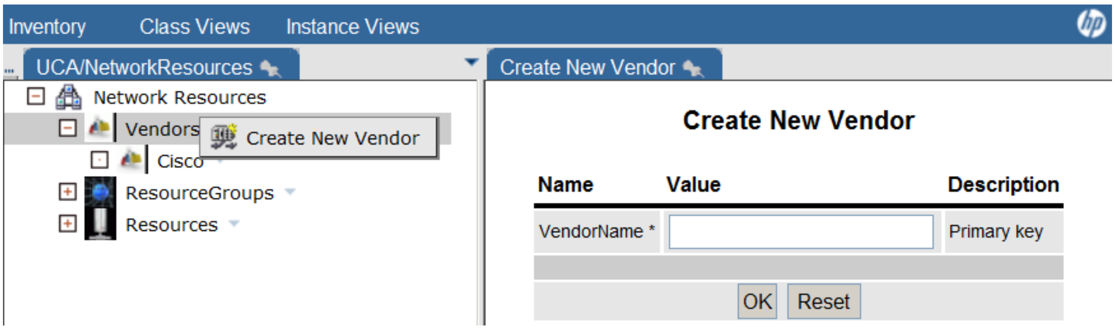


Figure 20 – HPSA Inventory – UCA/NetworkResources

Create a new Vendor by mouse right click on UCA/NetworkResources → Network Resources → Vendors → Create New Vendor option.

Enter the value for VendorName and click on OK to create a new vendor instance.

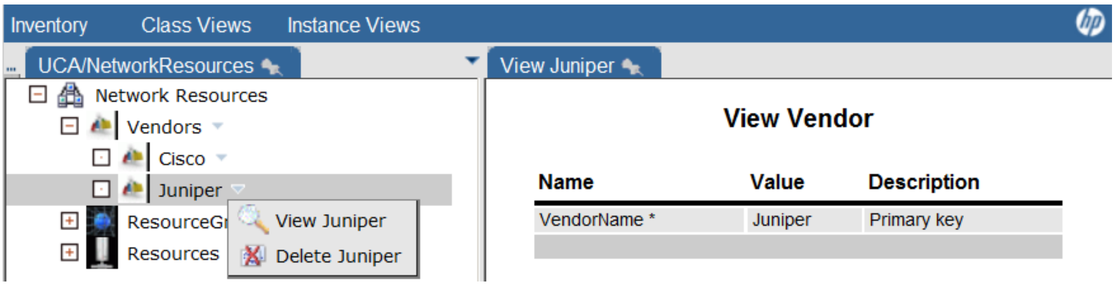


Create New Vendor

Name	Value	Description
VendorName *		Primary key

Figure 21 – HPSA Inventory – UCA/NetworkResources – Create New Vendor

Vendor instance can be viewed or deleted by mouse right click on the vendor, and choosing appropriate action.



View Vendor

Name	Value	Description
VendorName *	Juniper	Primary key

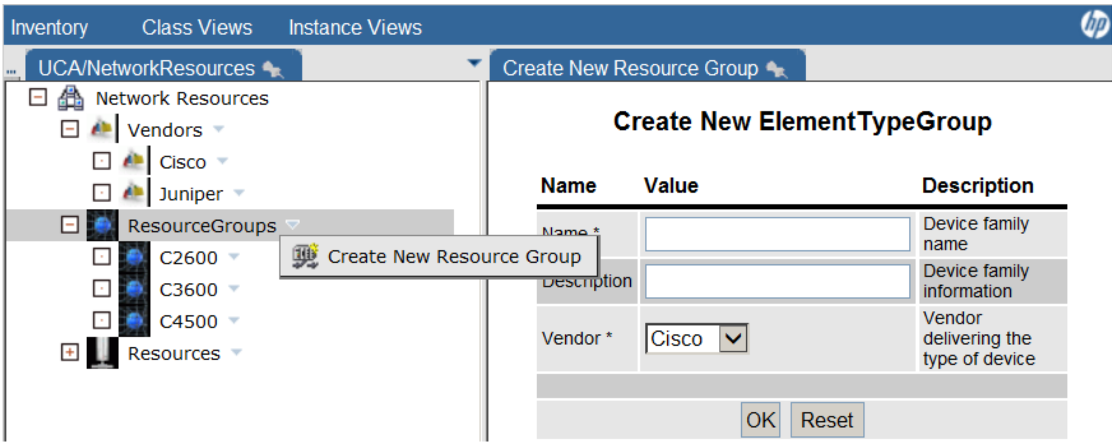
Figure 22 – HPSA Inventory – UCA/NetworkResources – View Vendor

Before creating any resource instance, we must create ResourceGroups, that identifies the resource type.

Create a new resource group instance by mouse right click on

UCA/NetworkResources → Network Resources → ResourceGroups → Create New Resource Group option.

Enter the value for resource group name, choose among the device vendor instances available, and click on OK to create a new resource group.

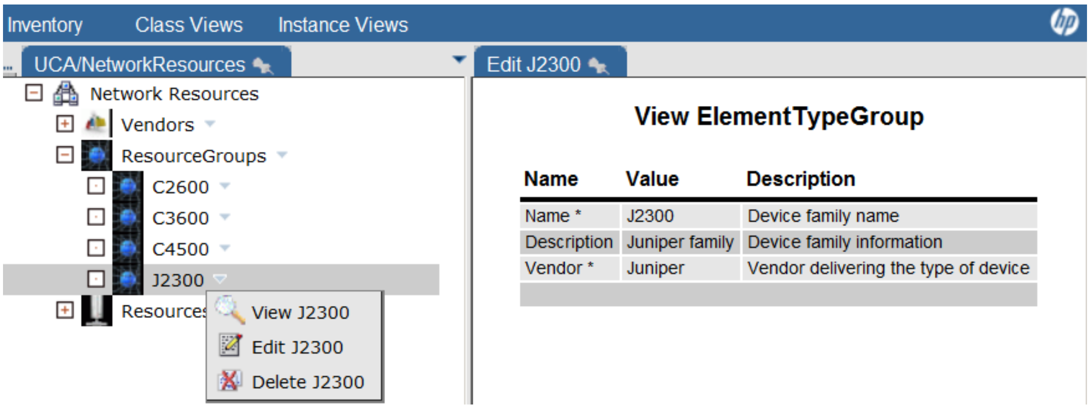


Create New ElementTypeGroup

Name	Value	Description
Name *		Device family name
Description		Device family information
Vendor *	Cisco	Vendor delivering the type of device

Figure 23 – HPSA Inventory – UCA/NetworkResources – Create New Resource Group

Resource group instance can be viewed,edited or deleted by mouse right click on the ResourceGroups instance, and choosing appropriate action.



View ElementTypeGroup

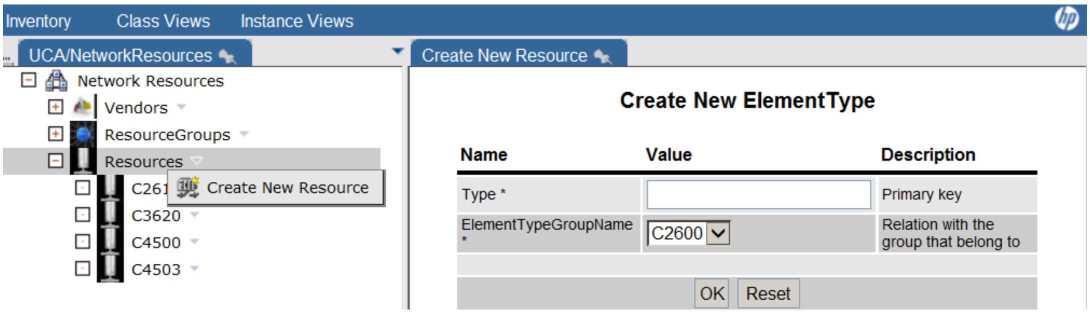
Name	Value	Description
Name *	J2300	Device family name
Description	Juniper family	Device family information
Vendor *	Juniper	Vendor delivering the type of device

Figure 24 – HPSA Inventory – UCA/NetworkResources – View Resource Group

Create a new resource instance by mouse right click on

UCA/NetworkResources → Network Resources → Resources → Create New Resource option.

Enter the value for resource instance, choose among the available resource groups device instances, and click on OK to create a new resource instance.

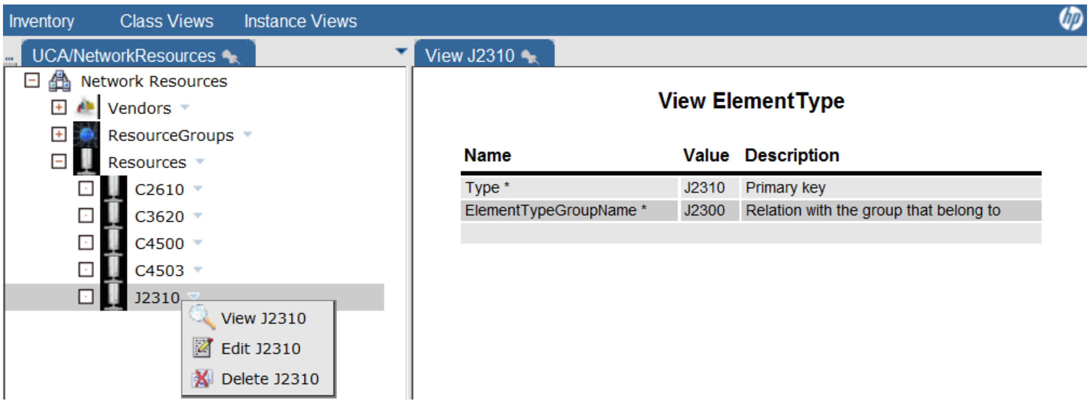


Create New ElementType

Name	Value	Description
Type *		Primary key
ElementTypeGroupName *	C2600	Relation with the group that belong to

Figure 25 – HPSA Inventory – UCA/NetworkResources – Create New Resource Type

Resource instance can be viewed,edited or deleted by mouse right click on the Resource instance, and choosing appropriate action.



View ElementType

Name	Value	Description
Type *	J2310	Primary key
ElementTypeGroupName *	J2300	Relation with the group that belong to

Figure 26 – HPSA Inventory – UCA/NetworkResources – View Resource Type

7.3.2 UCA/Services

UCA/Services provides option to capture the domain name (service name) to be supported.

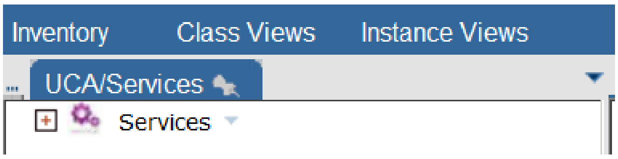
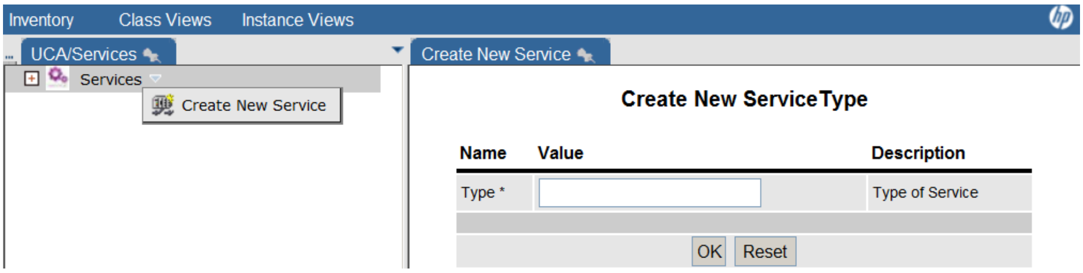


Figure 27 – HPSA Inventory – UCA/Services

Create a new service by mouse right click on UCA/Services → Services → Create New Service.

Enter the value for service and click on OK to create a new service type.

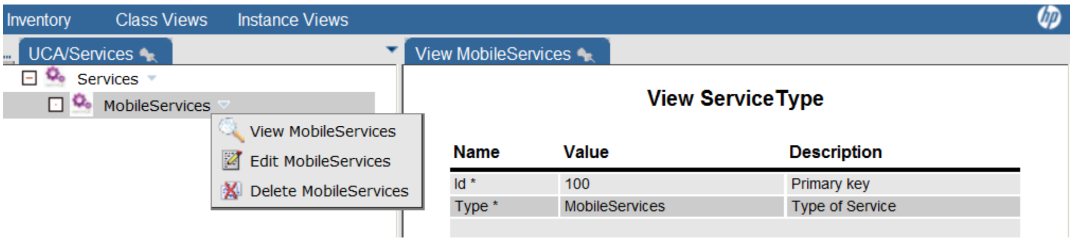


Create New ServiceType

Name	Value	Description
Type *		Type of Service

Figure 28 – HPSA Inventory – UCA/Services – Create New Service Type

Service instance can be viewed,edited or deleted by mouse right click on the Service instance, and choosing appropriate action.



View ServiceType

Name	Value	Description
Id *	100	Primary key
Type *	MobileServices	Type of Service

Figure 29 – HPSA Inventory – UCA/Services – View Service Type

7.3.3 UCA/ActionFramework

UCA/ActionFramework provides options to define the various Problems, Actions, and Actions to Resource mappings. Once created, the ActionFramework can be exported into a file. This file now can be edited to be used as an input for Neo4J graph.

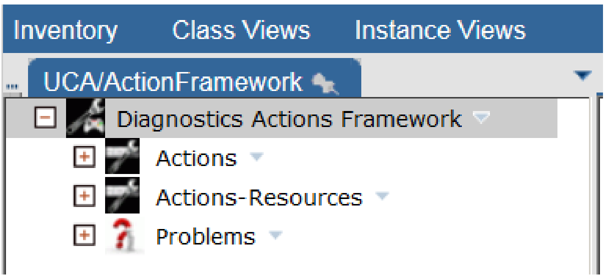
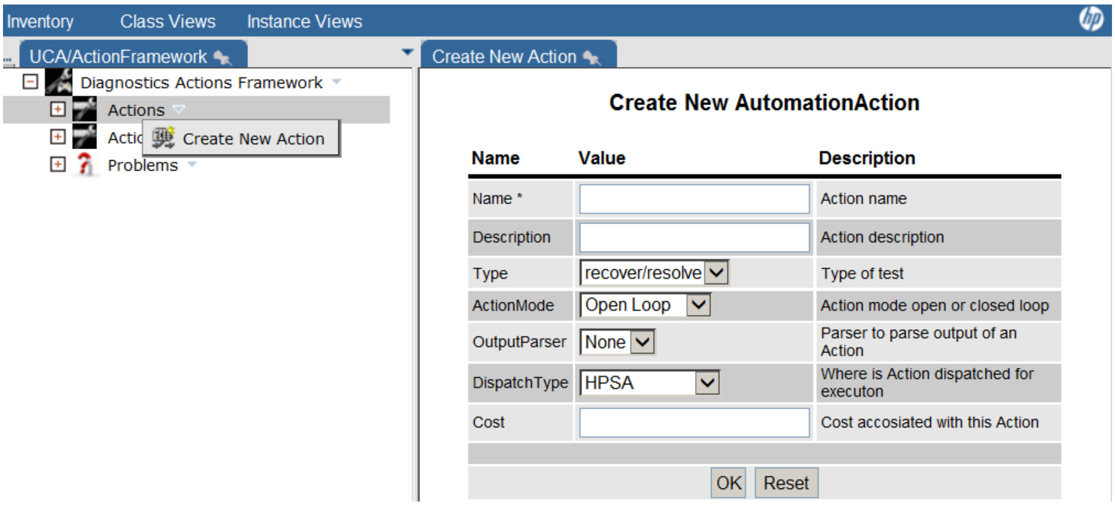


Figure 30 – HPSA Inventory – UCA/ActionFramework

Create a new action by mouse right click on UCA/ActionFramework → Diagnostics Actions Framework → Actions → Create New Action.

Enter the name for Action, action Type (recover/resolve, test, audit, read-only-test, internal, escalate), ActionMode (Open Loop, Closed Loop, None), OutputParser (None, regex, xpath), DispatchType (HPSA, OO, ShellScript, Alarm, Trouble Ticket) and click on OK to create a new action.



Create New AutomationAction

Name	Value	Description
Name *		Action name
Description		Action description
Type	recover/resolve	Type of test
ActionMode	Open Loop	Action mode open or closed loop
OutputParser	None	Parser to parse output of an Action
DispatchType	HPSA	Where is Action dispatched for executon
Cost		Cost accosiated with this Action

Figure 31 – HPSA Inventory – UCA/ActionFramework – Create New Action

Action Type	Description
Recover/Resolve	Action that can recover or resolve a Problem symptom
Test	Actions performed to test the network
Audit	Action to audit a pre-defined configuration
Read-only-test	Read only actions performed on the network
Internal	Trouble Ticket and Alarm handling actions
Escalate	Actions to escalate when a problem resolution is not performed

Table 8 – Action Types

Action Mode	Description
Open Loop	Actions that require operator intervention. The operator can optionally disapprove a test
Closed Loop	Actions that do not require operator intervention.
None	If set to None, the value set in the Parameters->GlobalParameter->ActionMode will be used

Table 9 – Action Modes

Output Parser	Description

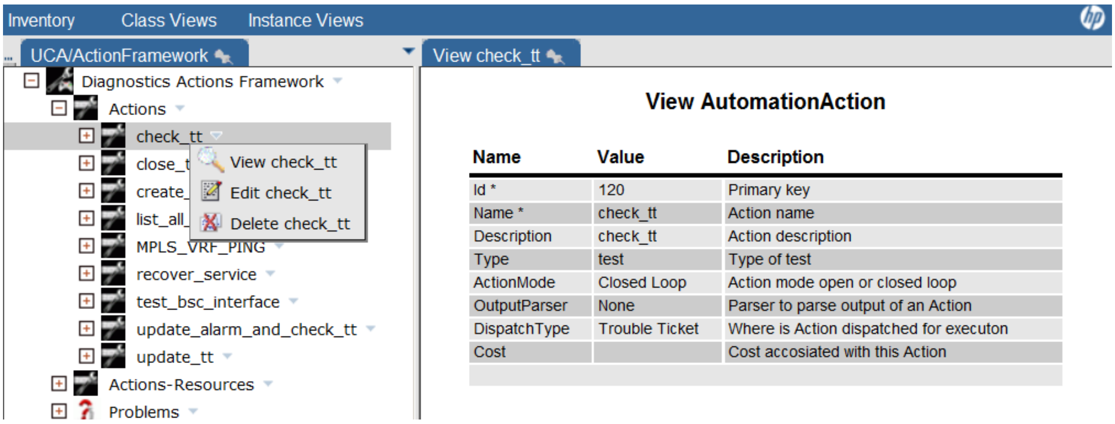
Regex	The action output is parsed using regular expressions.
Xpath	The action output is parsed using XPaths.
None	Action outputs are not parsed if this options is selected

Table 10 – Output Parsers

Dispatch Type	Description
HPSA	Action is dispatched to HP Service Activator for execution.
OO	HP Object Orchestration. For future possible integration
ShellScript	Action that can be executed using a shell script. For future possible integration
Alarm	An internal Alarm handling action. Possible operations could be Alarm creation/updation
Trouble Ticket	An internal Trouble Ticket action. Possible operations could be Trouble Ticket creation/updation/close

Table 11 – Dispatch Types

Action instance can be viewed,edited or deleted by mouse right click on the Actions instance, and choosing appropriate action.



View AutomationAction

Name	Value	Description
Id *	120	Primary key
Name *	check_tt	Action name
Description	check_tt	Action description
Type	test	Type of test
ActionMode	Closed Loop	Action mode open or closed loop
OutputParser	None	Parser to parse output of an Action
DispatchType	Trouble Ticket	Where is Action dispatched for executon
Cost		Cost accosiated with this Action

Figure 32 – HPSA Inventory – UCA/ActionFramework – View Action

Each Action instance can be designed to take one or more parameters as input.

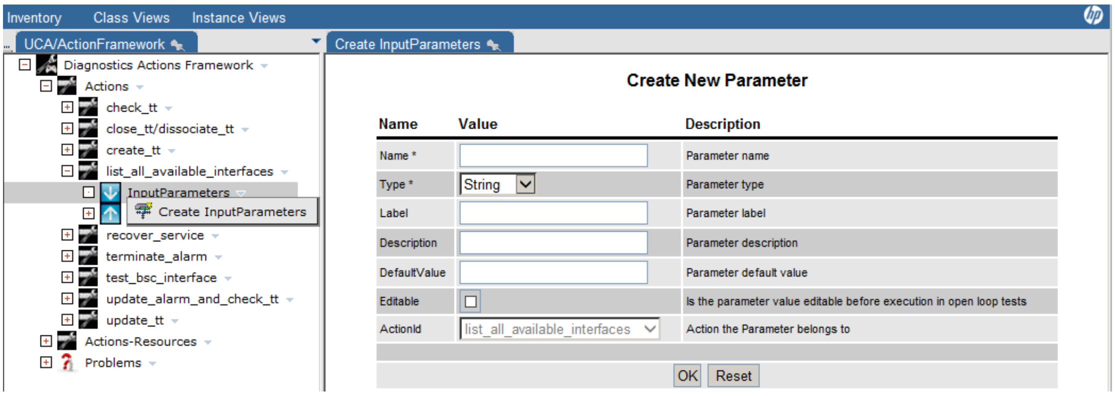
If the action mode is Open Loop, the operator can key in the input parameter at run time to supplement the alarm request.

If the action mode is Closed Loop, the input parameter value must be mapped from one of the fields in the Problem Alarm.

Create a new input parameter for an action by mouse right click on

UCA/ActionFramework → Diagnostics Actions Framework → Actions → <*Action* Instance> → Create InputParameters.

Enter the Name of the input parameter, data Type of the input parameter (String, int, Boolean), give a human readable Label to this input parameter, DefaultValue to the input parameter in case no value could be set to the input parameter, set the Editable flag to indicate whether the value is editable. Click on the OK button to create the input parameter.

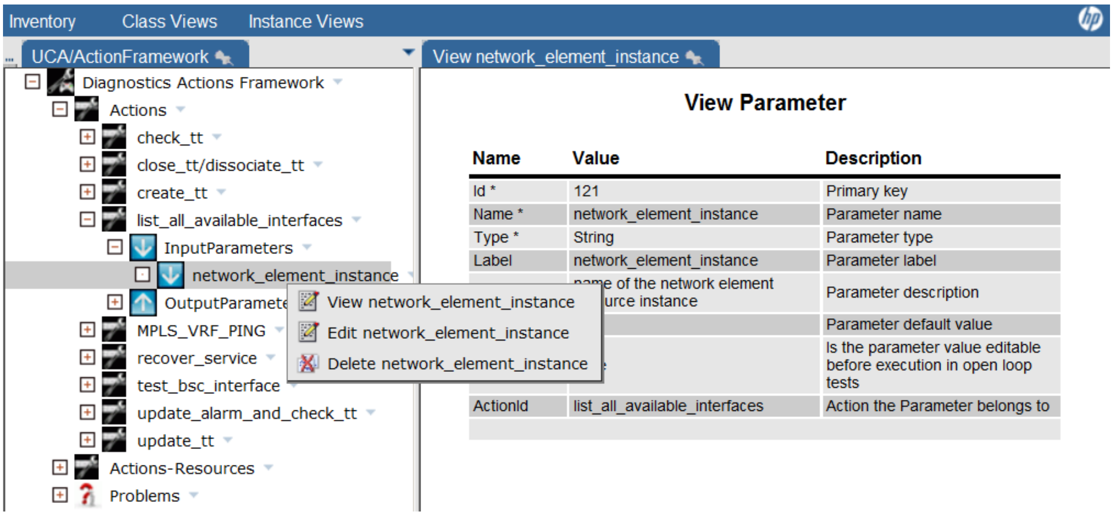


Create New Parameter

Name	Value	Description
Name *		Parameter name
Type *	String	Parameter type
Label		Parameter label
Description		Parameter description
DefaultValue		Parameter default value
Editable	□	Is the parameter value editable before execution in open loop tests
ActionId	list_all_available_interfaces	Action the Parameter belongs to

Figure 33 – HPSA Inventory – UCA/ActionFramework – Create New Input Parameter

Input parameter can be viewed,edited or deleted by mouse right click on the Input Parameter instance, and choosing appropriate action.



View Parameter

Name	Value	Description
Id *	121	Primary key
Name *	network_element_instance	Parameter name
Type *	String	Parameter type
Label	network_element_instance	Parameter label
	name of the network element urce instance	Parameter description
		Parameter default value
		Is the parameter value editable before execution in open loop tests
ActionId	list_all_available_interfaces	Action the Parameter belongs to

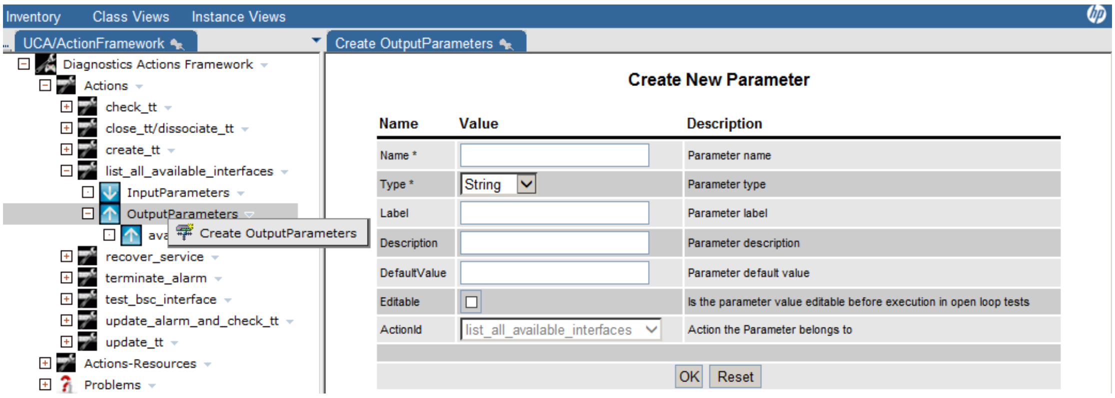
Figure 34 – HPSA Inventory – UCA/ActionFramework – View Input Parameter

Action involves invoking HP Service Activator workflows to perform specific task on the resource instance, and that may result in return of one or more values as output parameter, that can be used to take more granular decision in the problem resolution process.

Create a new output parameter for an action by mouse right click on

UCA/ActionFramework → Diagnostics Actions Framework → Actions → <*Action* Instance> → Create OutputParameters.

Enter the Name of the output parameter. Click on the OK button to create the output parameter.

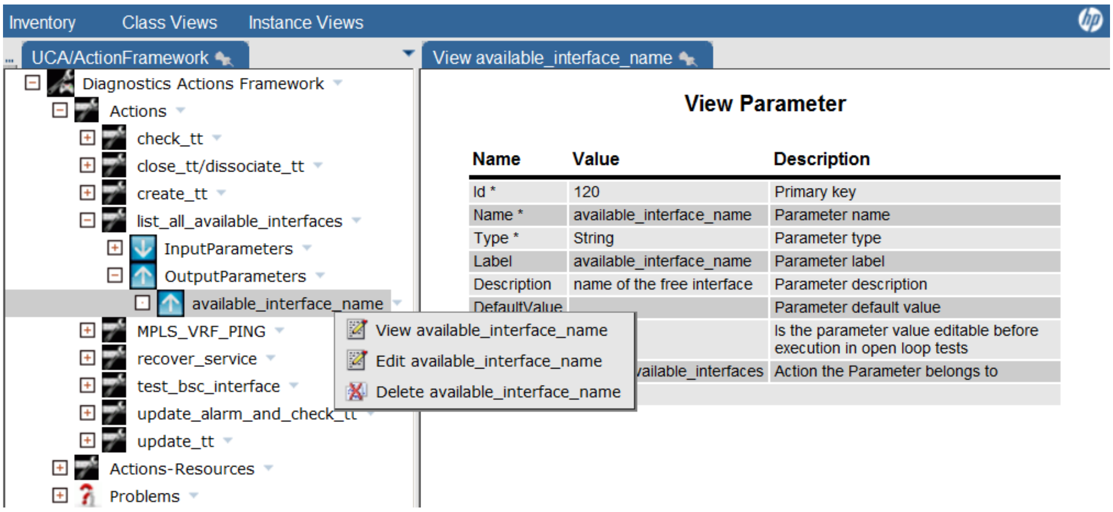


Create New Parameter

Name	Value	Description
Name *		Parameter name
Type *	String	Parameter type
Label		Parameter label
Description		Parameter description
DefaultValue		Parameter default value
Editable	□	Is the parameter value editable before execution in open loop tests
ActionId	list_all_available_interfaces	Action the Parameter belongs to

Figure 35 – HPSA Inventory – UCA/ActionFramework – Create New Output Parameter

Output parameter can be viewed, edited or deleted by mouse right click on the Output Parameter instance, and choosing appropriate action.



View Parameter

Name	Value	Description
Id *	120	Primary key
Name *	available_interface_name	Parameter name
Type *	String	Parameter type
Label	available_interface_name	Parameter label
Description	name of the free interface	Parameter description
DefaultValue		Parameter default value
		Is the parameter value editable before execution in open loop tests
	vailable_interfaces	Action the Parameter belongs to

Figure 36 – HPSA Inventory – UCA/ActionFramework – View Output Parameter

An alarm from a Resource Instance (device) can be identified as a Problem in the UCA Automation solution. Each of these problems can be associated with an action to be taken as a part of resolution. Various Problems that can occur in the device can be captured in this HPSA Inventory object.

Create a new Problem instance by mouse right click on

UCA/ActionFramework → Diagnostics Actions Framework → Problems → Create New Problem.

Enter the Name of the Problem, associate this Problem with a Service (domain), associate this Problem with a resolution Action, and an indication if this is the Root Level Problem.

Field	Description
Primary Problem	Indicates if the problem can be tested as an independent test.
Root Level Problem	For use by Neo4J graph DB to attach to the root node of the service name (domain)

ActionTraversalPath	This attribute is significant only for secondary problems which originate from internal actions like trouble ticketing or alarm handling. The value can be either 'true' or 'false' depending upon result of originating action. For all non-internal actions, this value should be set to 'None'

Table 12 – Problem Types

NOTE: '***Unknown Symptom***' represents a problem that does not fit in the given domain. Any coorelated alarm that is not represented as a problem in the domain takes the 'Unknown Symptom' resolution path. The problem must be named as "unknown_symptom" for such cases to be handled as expected in the graph. Else the alarm is updated and a trouble ticket is updated or created.

A sample path taken by the Unknown Symptom is as shown below.

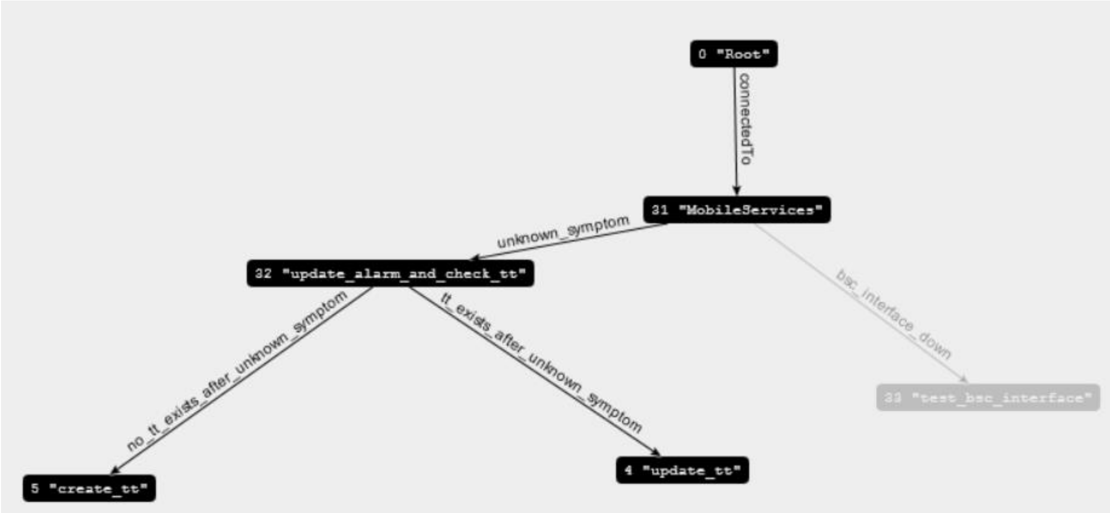
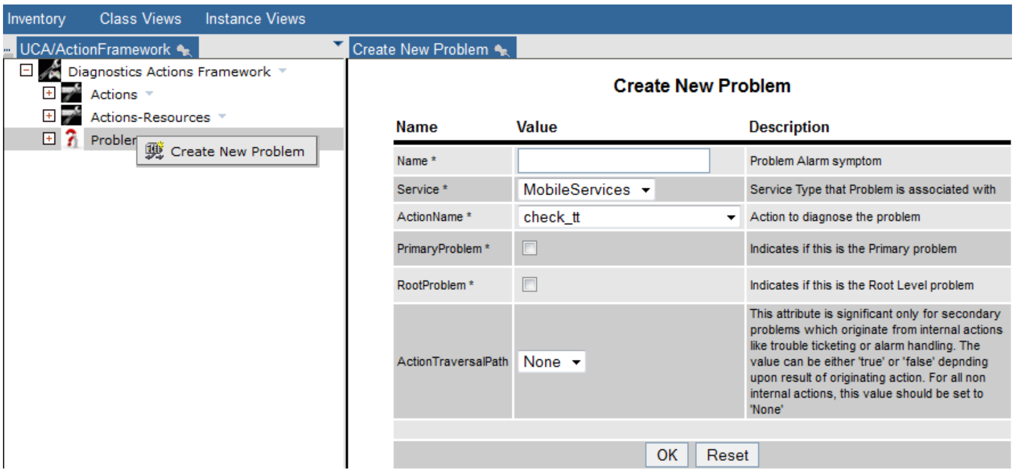


Figure 37 – Neo4J Graph sample Unknown Symptom path

Click on the OK button to create the Problem instance.

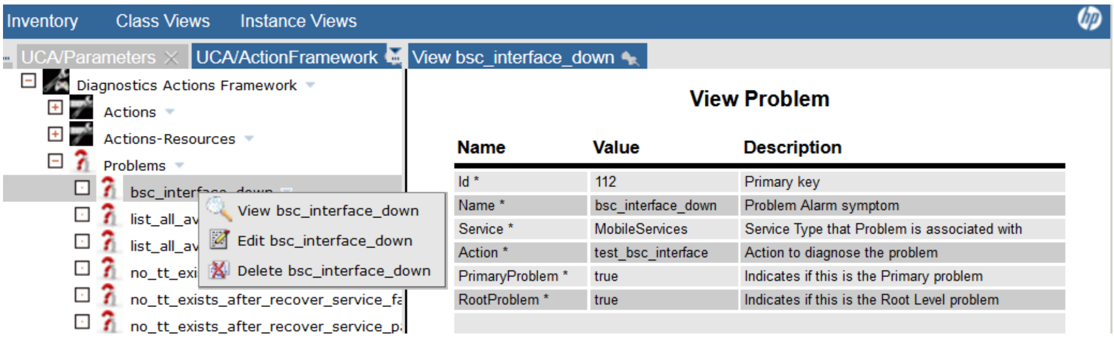


Create New Problem

Name	Value	Description
Name *		Problem Alarm symptom
Service *	MobileServices	Service Type that Problem is associated with
ActionName *	check_tt	Action to diagnose the problem
PrimaryProblem *	□	Indicates if this is the Primary problem
RootProblem *	□	Indicates if this is the Root Level problem
ActionTraversalPath	None	This attribute is significant only for secondary problems which originate from internal actions like trouble ticketing or alarm handling. The value can be either 'true' or 'false' depnding upon result of originating action. For all non internal actions, this value should be set to 'None'

Figure 38 – HPSA Inventory – UCA/ActionFramework – Create New Problem

Problem instance can be viewed, edited or deleted by mouse right click on the Problem instance, and choosing appropriate action.



View Problem

Name	Value	Description
Id *	112	Primary key
Name *	bsc_interface_down	Problem Alarm symptom
Service *	MobileServices	Service Type that Problem is associated with
Action *	test_bsc_interface	Action to diagnose the problem
PrimaryProblem *	true	Indicates if this is the Primary problem
RootProblem *	true	Indicates if this is the Root Level problem

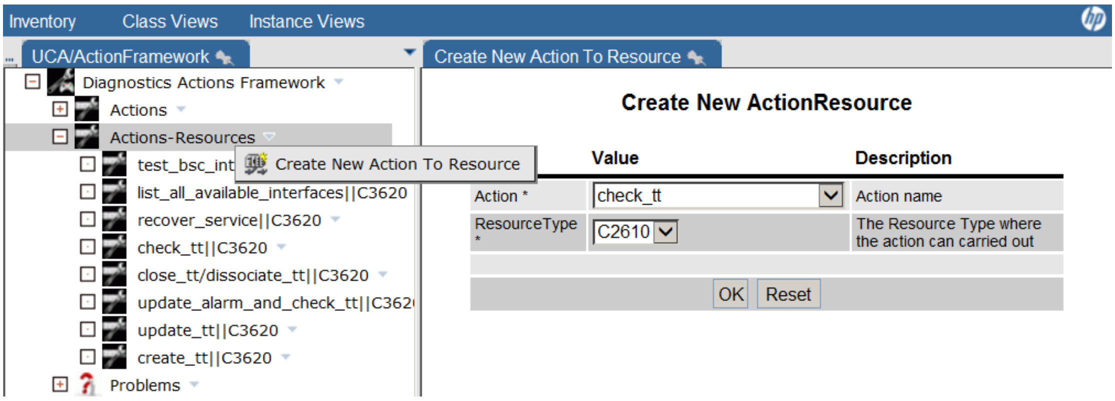
Figure 39 – HPSA Inventory – UCA/ActionFramework – View Problem

HPSA Inventory object Actions-Resources associated an action to one or more Resource Types. This indicates that the resolution action can be carried out on the selected ResourceType.

Create a new Actions-Resources instance by mouse right click on

UCA/ActionFramework → Diagnostics Actions Framework → Actions-Resources → Create New Action to Resource.

Select the available actions from the drop down, and associate it with one or more ResourceTypes. Click on OK to create the association.


Inventory
Class Views
Instance Views
UCA/ActionFramework
Create New Action To Resource
Diagnostics Actions Framework
Actions
Actions-Resources
test_bsc_int
list_all_available_interfaces||C3620
recover_service||C3620
check_tt||C3620
close_tt/dissociate_tt||C3620
update_alarm_and_check_tt||C362
update_tt||C3620
create_tt||C3620
Problems
Create New Action To Resource
OK
Reset

Create New ActionResource

	Value	Description
Action *	check_tt	Action name
ResourceType *	C2610	The Resource Type where the action can carried out

Figure 40 – HPSA Inventory – UCA/ActionFramework – Create New Action to Resource Mapping

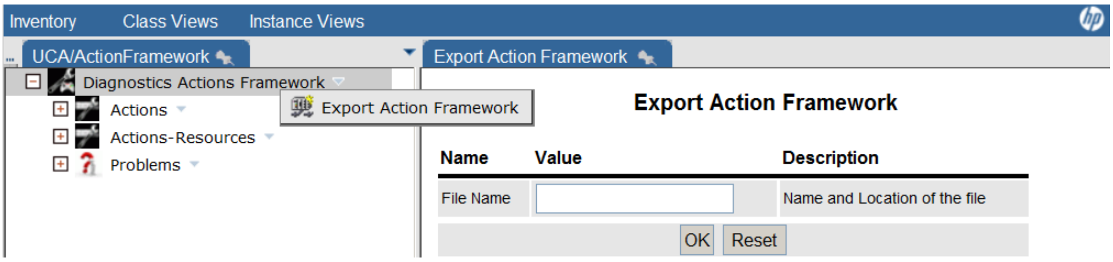
HPSA Inventory provides a mechanism to create an XML file that contains the list of all Actions, Problems, and Actions-Resources associations.

This XML file can be manually updated to associate the Problems to Actions and that can be uploaded into the Neo4J graph database so that it can be used by UCA Automation solution for decision making.

Export the Action Framework by mouse right click on

UCA/ActionFramework → Diagnostics Actions Framework → Export Action Framework.

Enter the absolute path and XML file name. Click on OK to export the file.



Export Action Framework

Name	Value	Description
File Name		Name and Location of the file

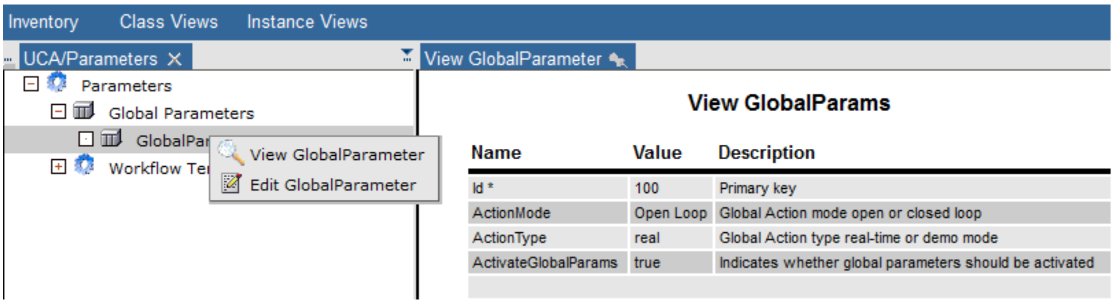
Figure 41 – HPSA Inventory – UCA/ActionFramework – Export Action Framework

NOTE: The parameters in the file /opt/OV/ServiceActivator/solutions/UCA/etc/template_files/GraphDBUtilities.properties must be set before using this feature.

NOTE: The absolute path entered must be the path in the HPSA server.

7.3.4 UCA/Parameters

Global Parameters that needs to be used in the application can be viewed or edited from Parameters → Global Parameters → Global Parameter.



View GlobalParams

Name	Value	Description
Id *	100	Primary key
ActionMode	Open Loop	Global Action mode open or closed loop
ActionType	real	Global Action type real-time or demo mode
ActivateGlobalParams	true	Indicates whether global parameters should be activated

Figure 42 – HPSA Inventory – UCA/Parameters – Global Parameters

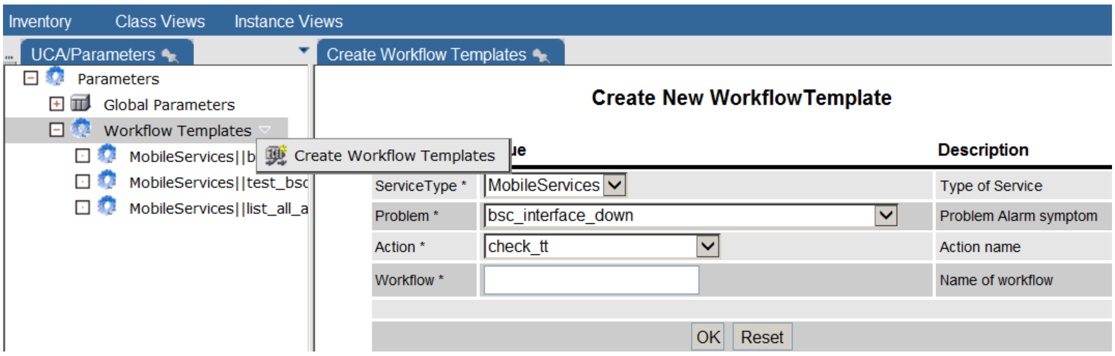
Parameter	Description
ActionMode	Open loop / Closed loop. This value is considered if Action instance ActionMode value is set to None.
Action Type	Real-time / demo mode. It indicates the HPSA Activation Engine to skip connecting to the resources (devices) if set as demo mode.
ActivateGlobalParams	The Global parameters can be switched off by setting this to false. The Action Mode and Type that are set in the individual Actions will be used.

Table 13 – Global Parameters

For a given ServiceType (domain), Problem to Action association must be handled by HP Service Activator workflow.
This association is captured in this HPSA inventory object.

Create a new HPSA Workflow association by mouse right click on

UCA/Parameters → Workflow Templates → Create Workflow Templates.



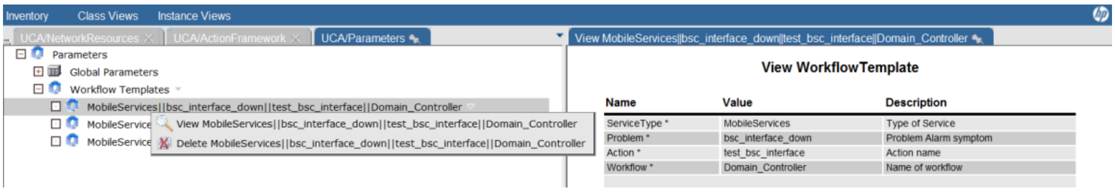
Create New WorkflowTemplate

ue		Description
ServiceType *	MobileServices	Type of Service
Problem *	bsc_interface_down	Problem Alarm symptom
Action *	check_tt	Action name
Workflow *		Name of workflow

OK Reset

Figure 43 – HPSA Inventory – UCA/Parameters – Create New Workflow Template

Workflow associations can be viewed or deleted by mouse right click on the Workflow Template instance, and choosing appropriate action.


Inventory
Class Views
Instance Views
UCA/NetworkResources
UCA/ActionFramework
UCA/Parameters
View MobileServices||bsc_interface_down||test_bsc_interface||Domain_Controller
Parameters
Global Parameters
Workflow Templates
MobileServices||bsc_interface_down||test_bsc_interface||Domain_Controller
MobileService
MobileService
View MobileServices||bsc_interface_down||test_bsc_interface||Domain_Controller
Delete MobileServices||bsc_interface_down||test_bsc_interface||Domain_Controller

View WorkflowTemplate

Name	Value	Description
ServiceType *	MobileServices	Type of Service
Problem *	bsc_interface_down	Problem Alarm symptom
Action *	test_bsc_interface	Action name
Workflow *	Domain_Controller	Name of workflow

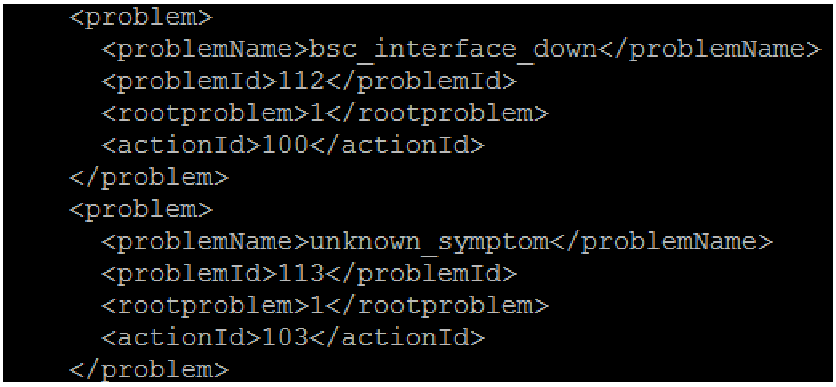
Figure 44 – HPSA Inventory – UCA/Parameters – View Workflow Template

Chapter 8

Populate Neo4J Graph DB data

The data in the Neo4J graph DB can be populated using the XML as input.
The XML is produced in two steps:

1. In the HPSA inventory, create all the Actions, Problems, and Action-Resource associations.
2. Once completed, export the Action framework, in the file system.
3. This XML file contains the <problem> blocks with the following structure.

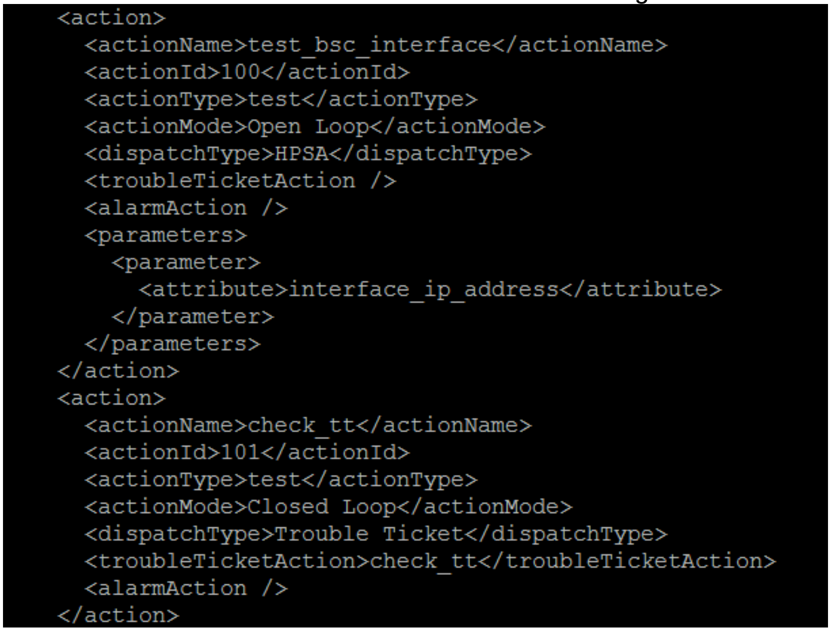


```
<problem>
  <problemName>bsc_interface_down</problemName>
  <problemId>112</problemId>
  <rootproblem>1</rootproblem>
  <actionId>100</actionId>
</problem>
<problem>
  <problemName>unknown_symptom</problemName>
  <problemId>113</problemId>
  <rootproblem>1</rootproblem>
  <actionId>103</actionId>
</problem>
```

Figure 45 – Neo4J XML - <problem> block

The tag <rootproblem> indicates that the specific problem is the parent problem, while generating the Neo0j graph.

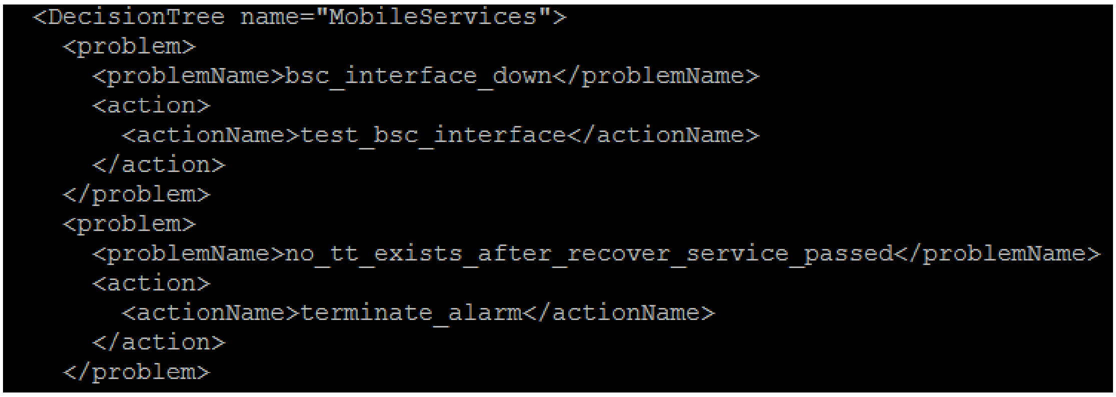
4. This XML file also contains the <action> blocks with the following structure.



```
<action>
  <actionName>test_bsc_interface</actionName>
  <actionId>100</actionId>
  <actionType>test</actionType>
  <actionMode>Open Loop</actionMode>
  <dispatchType>HPSA</dispatchType>
  <troubleTicketAction />
  <alarmAction />
  <parameters>
    <parameter>
      <attribute>interface_ip_address</attribute>
    </parameter>
  </parameters>
</action>
<action>
  <actionName>check_tt</actionName>
  <actionId>101</actionId>
  <actionType>test</actionType>
  <actionMode>Closed Loop</actionMode>
  <dispatchType>Trouble Ticket</dispatchType>
  <troubleTicketAction>check_tt</troubleTicketAction>
  <alarmAction />
</action>
```

Figure 46 – Neo4J XML - <action> block

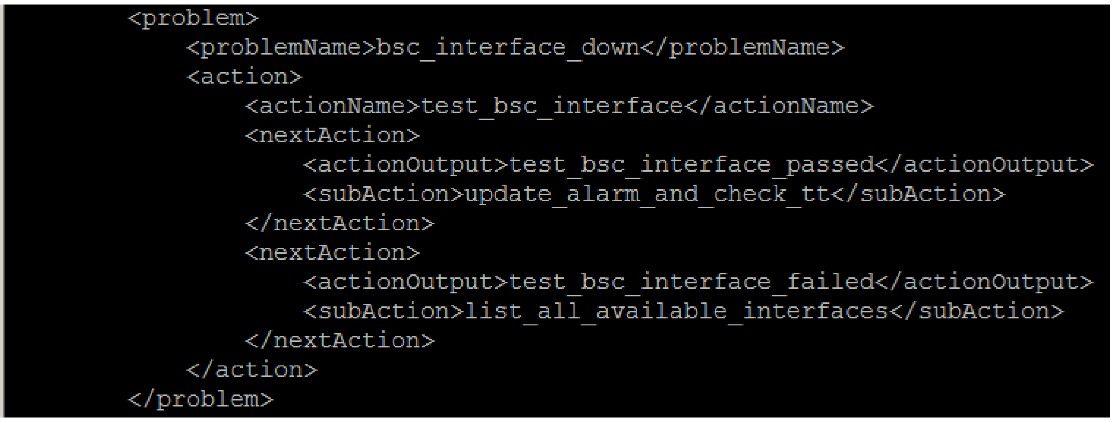
5. In the XML, there will be a block with <DecisionTree> tag, representing the service/domain.



```
<DecisionTree name="MobileServices">
  <problem>
    <problemName>bsc_interface_down</problemName>
    <action>
      <actionName>test_bsc_interface</actionName>
    </action>
  </problem>
  <problem>
    <problemName>no_tt_exists_after_recover_service_passed</problemName>
    <action>
      <actionName>terminate_alarm</actionName>
    </action>
  </problem>
```

Figure 47 – Neo4J XML - <DecisionTree> block

   a. The above block contains the Problem to Action relationship. Based on the outcome of the above Action, next set of Actions need to be defined. This relationship has to be manually built for each Problem in the <DecisionTree> block. Within the <nextAction> block, the outcome from the Action is represented as <actionOutput> tag and new Action to be taken for this outcome is represented as <subAction> tag.
   b. Each of the next Action is already represented as an Action object in the HPSA inventory.
   c. Each of the possible outcome from an Action is already represented as a Problem object in the HPSA inventory.



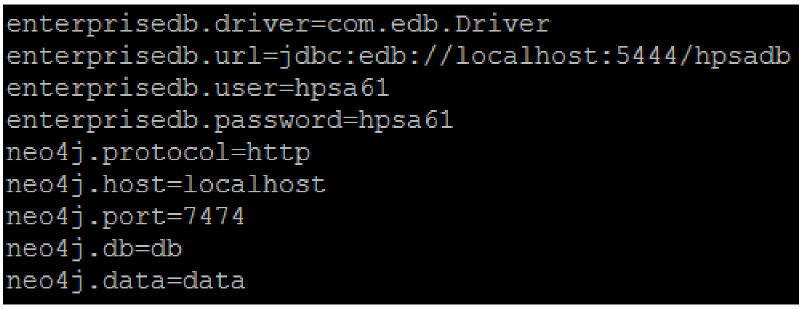
```
<problem>
    <problemName>bsc_interface_down</problemName>
    <action>
        <actionName>test_bsc_interface</actionName>
        <nextAction>
            <actionOutput>test_bsc_interface_passed</actionOutput>
            <subAction>update_alarm_and_check_tt</subAction>
        </nextAction>
        <nextAction>
            <actionOutput>test_bsc_interface_failed</actionOutput>
            <subAction>list_all_available_interfaces</subAction>
        </nextAction>
    </action>
</problem>
```

Figure 48 – Neo4J XML - <problem> to <action> association

   d. In the above snippet, the possible outcomes for the Action test_bsc_interface are test_bsc_interface_passed and test_bsc_interface_failed. If the outcome is test_bsc_interface_passed, the next Action to be taken is update_alarm_and_check_tt. If the outcome is test_bsc_interface_failed, the next Action to be taken is list_all_available_interfaces.

6. The above exercise of linking has to be carried out for each Problem.
7. Once the XML is modified appropriately, to view if it is as per expectation, it can be uploaded to Neo4J DB, and viewed. If corrections are needed, the same can be deleted and re uploaded.
8. Following process can be followed to upload this XML into Neo4J DB
   a. Ensure that the details are updated in the properties file GraphDBUtilities.properties

   Sample GraphDBUtilities.properties is as shown below

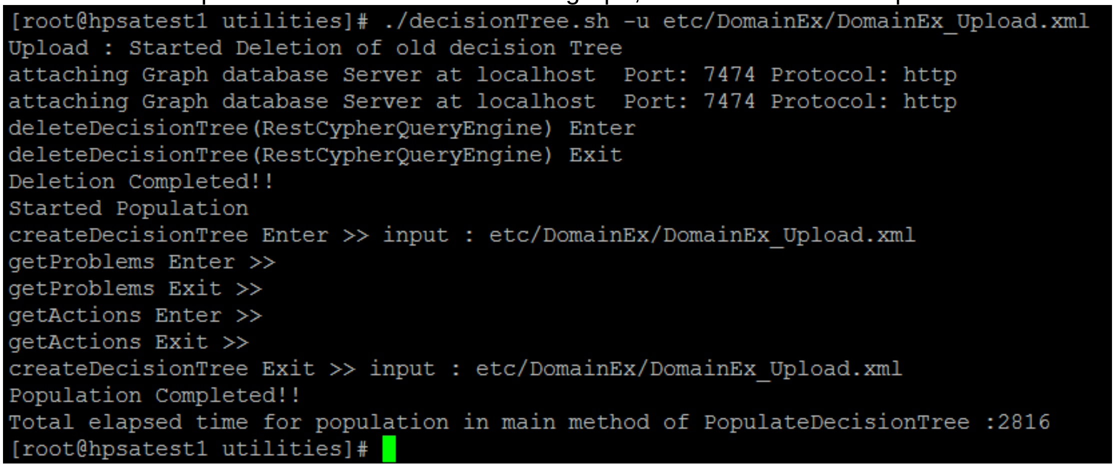


```
enterprisedb.driver=com.edb.Driver
enterprisedb.url=jdbc:edb://localhost:5444/hpsadb
enterprisedb.user=hpsa61
enterprisedb.password=hpsa61
neo4j.protocol=http
neo4j.host=localhost
neo4j.port=7474
neo4j.db=db
neo4j.data=data
```

Figure 49 – Neo4J XML – GraphDBUtilities.properties

b. Run the script /opt/UCA-ATM/utilities/decisionTree.sh

   i. To upload XML content into Neo4J graph, use –u <XML File> option



```
[root@hpsatest1 utilities]# ./decisionTree.sh -u etc/DomainEx/DomainEx_Upload.xml
Upload : Started Deletion of old decision Tree
attaching Graph database Server at localhost  Port: 7474 Protocol: http
attaching Graph database Server at localhost  Port: 7474 Protocol: http
deleteDecisionTree(RestCypherQueryEngine) Enter
deleteDecisionTree(RestCypherQueryEngine) Exit
Deletion Completed!!
Started Population
createDecisionTree Enter >> input : etc/DomainEx/DomainEx_Upload.xml
getProblems Enter >>
getProblems Exit >>
getActions Enter >>
getActions Exit >>
createDecisionTree Exit >> input : etc/DomainEx/DomainEx_Upload.xml
Population Completed!!
Total elapsed time for population in main method of PopulateDecisionTree :2816
[root@hpsatest1 utilities]# 
```

Figure 50 – Neo4J XML – decisionTree.sh – upload XML

   ii. To extract data from the HPSA inventory (Action, Problem, Problem Action association), use the –e option

   iii. To delete an already loaded graph from Neo4J DB, use –d option

9.

The snippet representing the above scenario is as shown below

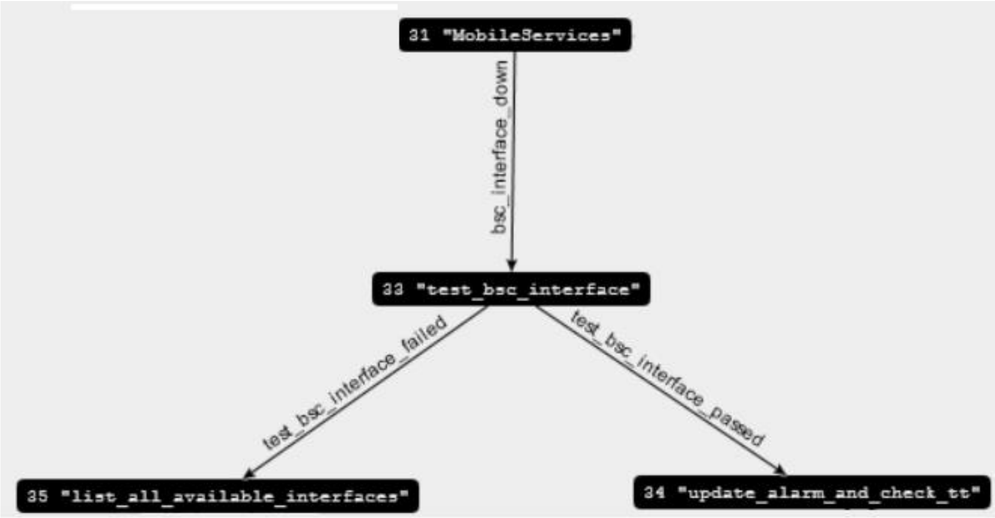

31 "MobileServices"
bsc_interface_down
33 "test_bsc_interface"
test_bsc_interface_failed
test_bsc_interface_passed
35 "list_all_available_interfaces"
34 "update_alarm_and_check_tt"

Figure 51 – Neo4J Graph snippet

10. A sample graph with various problems and actions as represented in Neo4J graph DB is shown below

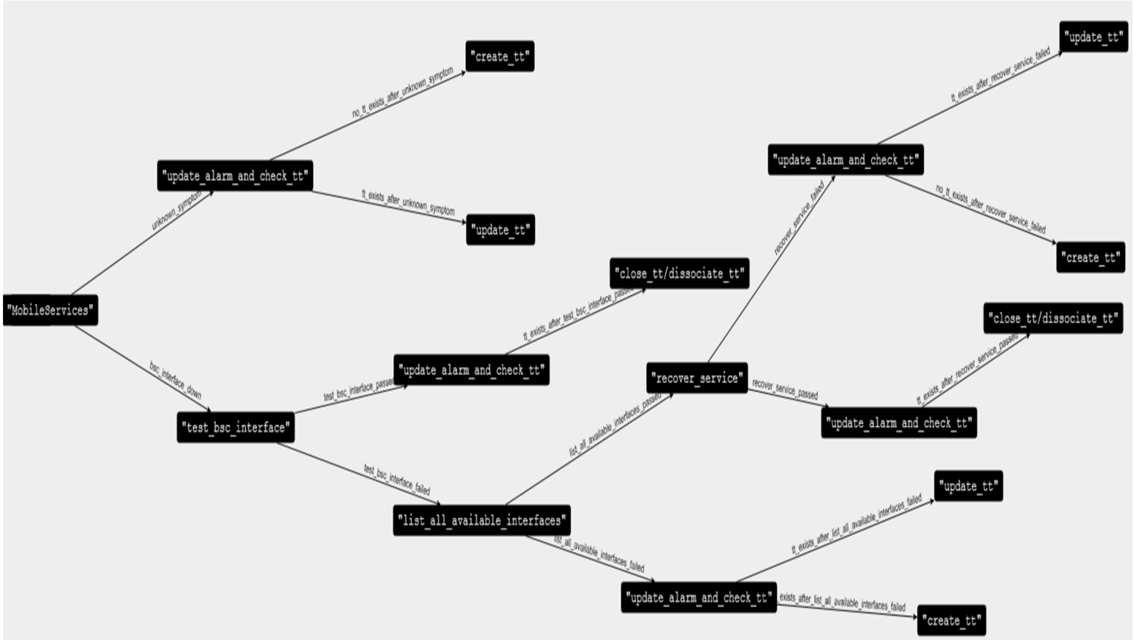

"create_tt"
no_tt_exists_after_unknown_symptom
"update_alarm_and_check_tt"
tt_exists_after_unknown_symptom
"update_tt"
unknown_symptom
"MobileServices"
bsc_interface_down
"test_bsc_interface"
test_bsc_interface_passed
"update_alarm_and_check_tt"
tt_exists_after_test_bsc_interface_passed
"close_tt/dissociate_tt"
test_bsc_interface_failed
"list_all_available_interfaces"
list_all_available_interfaces_passed
"recover_service"
recover_service_failed
"update_alarm_and_check_tt"
tt_exists_after_recover_service_failed
"update_tt"
no_tt_exists_after_recover_service_failed
"create_tt"
recover_service_passed
"update_alarm_and_check_tt"
tt_exists_after_recover_service_passed
"close_tt/dissociate_tt"
list_all_available_interfaces_failed
"update_alarm_and_check_tt"
tt_exists_after_list_all_available_interfaces_failed
"update_tt"
exists_after_list_all_available_interfaces_failed
"create_tt"

Figure 52 – Neo4J Graph sample

Chapter 9

Launching Test on UCA Automation from TeMIP Client

This chapter focuses on the steps that need be performed manually in order to integrate TeMIP Client with UCA Automation solution.

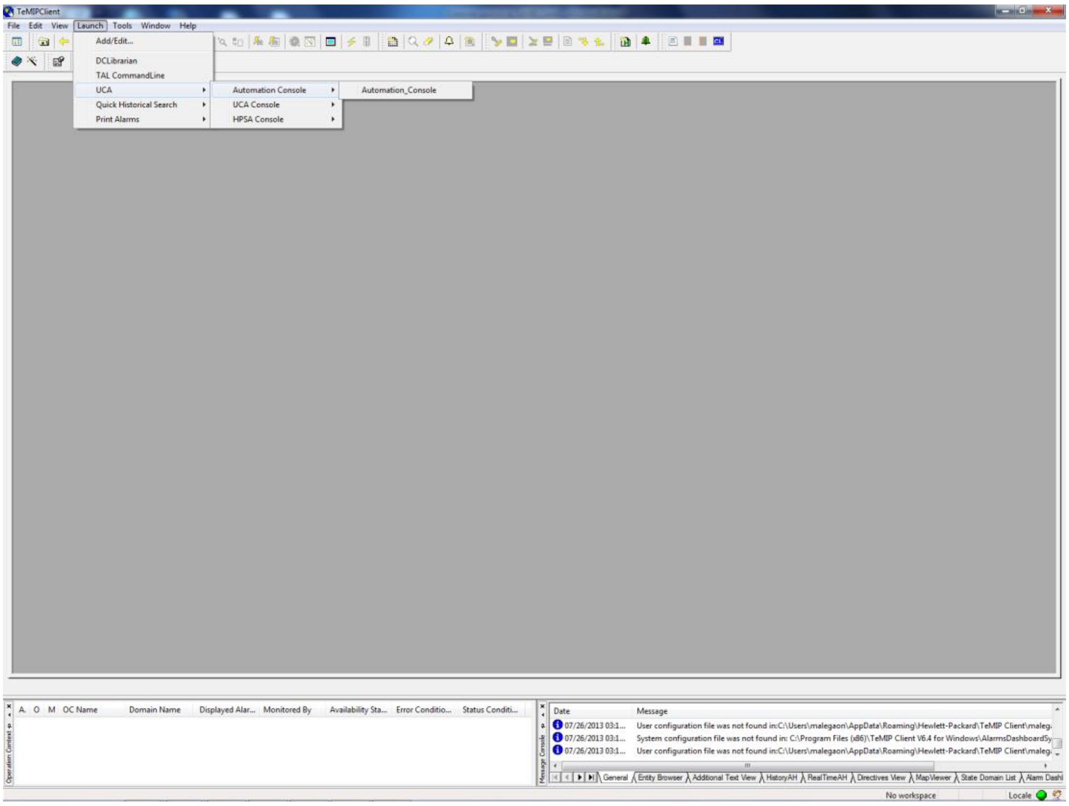
9.1 Integrate UCA Automation from TeMIP Client

In order to perform the cross launch of UCA Automation application from TeMIP Client, follow the instructions as given below:

- Go to UCA Automation installation directory : **/opt/UCA_Automation**
- Go to TeMIP_**Integration** directory.
- Copy the file **UCA_Automation_SetupLaunch.conf** to %TEMIP_CLIENT_HOME%/TeMIPClient_SystemLaunch directory on the machine on which TeMIP Client is installed.

  Eg – In Windows, it would be
  C:\Program Files (x86)\TeMIP Client V6.4 for Windows\TeMIPClient_SystemLaunch
- Customize the file for your needs and save the file. See Section 9.2 for details.
- Restart the TeMIP Client to validate options in Launch

Once the instructions are followed, the TeMIP Client will show the following menu option:



A.	O	M	OC Name	Domain Name	Displayed Alar...	Monitored By	Availability Sta...	Error Conditio...	Status Conditi...

Date	Message
07/26/2013 03:1...	User configuration file was not found in:C:\Users\malegaon\AppData\Roaming\Hewlett-Packard\TeMIP Client\maleg
07/26/2013 03:1...	System configuration file was not found in: C:\Program Files (x86)\TeMIP Client V6.4 for Windows\AlarmsDashboardSy
07/26/2013 03:1...	User configuration file was not found in:C:\Users\malegaon\AppData\Roaming\Hewlett-Packard\TeMIP Client\maleg

Figure 53 – TeMIP Client – UCA Automation cross launch

9.2 Customize the UCA Automation UI Integration file

9.2.1 Configure the Automation Console submenu

- Open the file **UCA_Automation_SetupLaunch.conf** in a suitable editor
- Locate the automation console sub-menu [*SUBMENU = UCA Automation \nAutomation Console*]
- Locate the open-url tag for this submenu [ARGUMENTS = @OpenURL /e http://<*hostname*>:<*port*>/UCAAutomation/]
- Customize ***hostname*** and ***port*** to point to the configured IP address / hostname and port.

9.2.2 Configure the UCA EBC Console submenu

- Open the file **UCA_Automation_SetupLaunch.conf** in a suitable editor
- Locate the uca-ebc console sub-menu [*SUBMENU = UCA Automation \nUCA EBC Console*]
- Locate the open-url tag for this submenu [ARGUMENTS = @OpenURL /e http://<*hostname*>:<*port*>/]
- Customize ***hostname*** and ***port*** to point to the configured IP address / hostname and port.

9.2.3 Configure the HPSA Console submenu

- Open the file **UCA_Automation_SetupLaunch.conf** in a suitable editor
- Locate the hpsa console sub-menu [*SUBMENU = UCA Automation \nHPSA Console*]
- Locate the open-url tag for this submenu [ARGUMENTS = @OpenURL /e http://<*hostname*>:<*port*>/activator/]
- Customize ***hostname*** and ***port*** to point to the configured IP address / hostname and port.