HP UCA Automation



Version 1.1

Administrator and User Interface Guide

Edition: 1.0

October 2014

© Copyright 2014 Hewlett-Packard Development Company, L.P.

Legal notices

Warranty

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

License Requirement and U.S. Government Legend

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

Copyright notices

© Copyright 2014 Hewlett-Packard Development Company, L.P.

Trademark notices

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

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

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

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

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[®] is a registered U.S. trademark of EnterpriseDB, Bedford, MA.

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

Legal not	ices		2
Contents	•••••		3
Figures			5
Tables	•••••		6
Preface			7
	Intende	d audience	7
	Softwar	re versions	7
	Typogra	aphical conventions	7
	Referen	nce documents	8
	Support	t	8
Chapter 1	•••••		9
Overview			9
	1.1	Design theory 1	1
Chapter 2			2
Accessing	j UCA Ai	utomation 1	2
	2.1	Launch UCA Automation UI 1	2
	2.2	User interface layout 1	2
	2.3	Logging into UI 1	3
Chapter 3	•••••		4
Configuri	ng UCA	Automation	4
	-	Global parameters	
		Activating Global Parameters1	
Chapter 4	•••••		6
Monitorin	a		6
	-	View status of tests	
		Updating the Waiting for Operator records	
		Managing Auto refresh	
.			
Chapter 5	•••••		9
Starting t	ests ma	anually 1	9
Chapter 6			1

Launching	g Topo	logy view	21
Chapter 7	•••••		22
Generatin	ıg repo	rts	22
	7.1	Deleting a report	.23
Chapter 8	•••••		25
UCA Auto	mation	Orchestrator	25
	8.1	Prerequisites	. 25
	8.2	Create new decision tree project	. 25
	8.3	UCA Automation Orchestrator UI layout	. 27
	8.4	Define Model Objects	. 28
	8.4.1	Create a new service type	. 29
	8.4.2	Create a new problem	. 30
	8.4.3	Create a new action	. 30
	8.5	Update or delete model objects	. 33
	8.5.1	Modify or delete a Service type	. 33
	8.5.2	Modify or delete an Action	. 33
	8.5.3	Modify or delete a configured Problem	.34
	8.6	Define parameters	35
	8.6.1	Create a parameter	.35
	8.6.2	View and update existing parameters	.36
	8.6.3	Delete a parameter	. 37
	8.7	Create an Action Outcome	. 38
	8.7.1	View and update Action Outcomes	. 39
	8.7.2	Delete an action outcome	. 40
	8.8	Orchestrating Model Objects	. 41
	8.8.1	Launch Orchestration	.41
	8.8.2	Add a Service to decision tree	42
	8.8.3	Delete a Service from the decision tree using CLI	.43
	8.8.4	Add an Action to decision tree	.44
	8.8.5	Link Service node to Action node	.45
	8.8.6	Link Action nodes	.46
	8.8.7	Save decision tree	.47
	8.9	Deploy decision tree	.48
	8.10	Generate skeleton decision tree using CLI	.49
	8.10.1	Deploy using command Line utility	.49

Figures

Figure 1 UCA Automation	10
Figure 2 UCA Automation workflow	11
Figure 3 UCA Automation User Interface main view	13
Figure 4 Login panel	13
Figure 5 Global Parameters page	15
Figure 6 Monitor view	16
Figure 7 Monitor show details	17
Figure 8 Monitor view result	18
Figure 9 Monitor interact page	
Figure 10 Manual Tests main page	19
Figure 11 Manual Tests – Selected action for a problem	20
Figure 12 Topology View – launch Neo4j data browser	21
Figure 13 Report search criteria	
Figure 14 Report default search	
Figure 15 Report search for archived records	
Figure 16 Create New Decision Tree Project Wizard	
Figure 17 UCA Automation Orchestrator main page in Eclipse	
Figure 18 Automation Definition View	
Figure 19 Create a new service	
Figure 20 Create a new Problem	
Figure 21 Create a new Action	
Figure 22 Update a Service	
Figure 23 Update an Action	
Figure 24 Update a Problem	
Figure 25 Create a Parameter	
Figure 26 List of parameters	
Figure 27 List of parameters	
Figure 28 Create an Action Outcome	
Figure 29 Update an Action Outcome	
Figure 30 Update an Action Outcome	
Figure 31 UCA Automation Orchestration	
Figure 32 Add Service node to orchestrator editor	
Figure 33 Add an Action node to orchestrator editor	
Figure 34 Action Outcome link	
Figure 35 Final Decision Tree	
Figure 36 Confirmation message	
Figure 37 Decision tree after uploading into Neo4j graph database	49

Tables

Table 1 Monitoring	
Table 2 UCA Actions field descriptions	20
Table 3 Report search criteria	22
Table 4 Report actions	23
Table 5 Action Types	
Table 6 Action Modes	
Table 7 Output Parsers	
Table 8 Dispatch Types	

Preface

This guide provides an overview of the UCA Automation product and describes how to use the web-based user interface of the UCA Automation.

Product Name: UCA Automation

Product Version: 1.1

Read this document before installing or using this software.

Intended audience

This guide is intended for solution developers, software development engineers, solution administrator, and solution operators. The administrators, operators, and observers have different privileges provided through the user interfaces.

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 1.1	Linux Red Hat Enterprise Linux Server release RHEL 6.4

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

However, the described features should be identical on any supported browser. For UCA Automation 1.1, the supported browsers are Mozilla Firefox 32 and Microsoft Internet Explorer 9.0.

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.

Reference documents

- UCA Automation Installation Guide
- UCA Automation Integrator's Guide
- Deployment Manager Guide (HPSA)
- NOM Installation and Configuration Guide
- NOM HPSA Channel Adapter Installation Guide
- NOM UCA Automation Console Channel Adapter Installation Guide
- NOM UCA EBC Channel Adapter Installation Guide
- NOM TEMIP Channel Adapter Installation Guide

Support

You can visit the HP software support online web site at <u>www.hp.com/go/hpsoftwaresupport</u> 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

Overview

In any typical service provider environment, a need for isolating network related issues and automating resolutions for the same is always in demand. UCA Automation software is positioned primarily to address this need. It is implemented as a combination of business rules engine and workflows engine.

The system involves the integration of HP Unified Correlation Analyzer for Event Based Correlation (UCA EBC) system, which provides business rules capability and HP Service Activator (HPSA), which provides activation capability] glued via the enterprise service bus called NOM (NGOSS Open Mediation).

Generally, in the resolution solutions available today, there's no separate layering between what resolution steps need to be carried out on the incidence of a specific issue and how these resolution steps are carried out. This mixed up implementation of processes which represent both what and how part of the logic on the top of workflow engines OR in some cases on top of business rules engines, makes the workflows or business rules very complex to develop, comprehend, debug and maintain, read as 'modify' when a business decision changes - say support a new device type, support a new resolution command on the same device or support a new format of the output for an existing resolution command with an upgrade in device firmware. The problem scales to unimaginable magnitudes considering the different technologies such as DWDM, SDH, DSL, MPLS, LTE and legacies such as ATM, FR and X.25] and different layers / types of networks such as the transport, the access, the core, the radio access and so on.

UCA Automation software which is a combination of both business rule engine and the workflow engine will enable a clear separation of what to automate and how to automate. All the complexities of actual automation such as how to access a network resource [could be a network element, an element component or an EMS or NMS], what it's credentials could be, which specific transport mechanism to use to connect to the resource, what specific OS version of the device are to be supported, what specific commands need to be sent, would be abstracted from the business rules. This would enable the administrators to create-update-read the business rules with utmost clarity and maintain them efficiently. This would empower the administrators to store the knowledge gained regarding the automation in the form of business rules focusing on what part without bothering about the how part. One another advantage of UCA Automation software is, for most of the resolution automations - it would require the operator only to know business rules and he need not have knowledge of the business rules technologies to implement day to day operational changes to the decisions.

Thus UCA Automation System is a platform for building value added resolution automations based on a judicious combination of business rules and workflows.

The following diagram shows the architecture of the UCA Automation system.

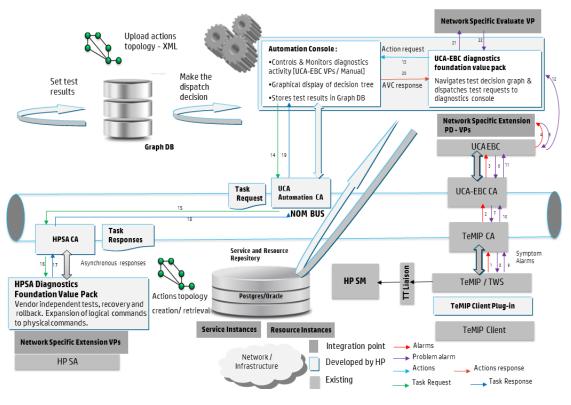


Figure 1 UCA Automation

The UCA Automation system process is as follows:

- 1. The network-specific PD value packs process a storm of alarms from an NMS such as TeMIP and create one Problem Alarm.
- 2. The UCA-EBC Diagnostics Foundation value pack acts on the Problem Alarm, which it receives either from the NOM bus or from the PD value pack.
- 3. The Decide-and-Act subsystem in the Foundation VP refers to a decision tree and determines the remedial actions for the problem.
- 4. The value pack sends these actions to the Automation console, enabling the user to view them, and later to an Activation System like HPSA.
- 5. The HPSA Foundation VP validates the incoming request and then sends them to the workflows in the network specific value pack.
- 6. The diagnostic results are sent back to the UCA-EBC Foundation value pack through the Automation console.
- 7. Depending on whether the administrator wants to make complex decisions based on the results, the diagnostic results are sent to the Evaluate value packs.
- 8. The Foundation value pack repeats the whole process of retrieving the next action from the decision tree, until it reaches the last action in the branch.

1.1 Design theory

Two key functions are performed by UCA Automation; the problem isolation and problem resolution.

Problem isolation is the responsibility of UCA EBC Problem Diagnosis value pack, which can eliminate event storms, false positives, false negatives, and deduce a single meaningful problem alarm.

This information is then passed to the decide-and-act engine, which identifies the action to be taken for a specific problem. After the action, the evolved knowledge is sent back to the decide-and-act engine for further resolution based on the decision tree or evaluates value-pack optionally, to perform predictive and proactive automation. In addition, diagnostic information is gathered automatically to reduce the MTTR (mean time to resolve).

The UCA Automation system works in the way depicted by the following diagram. It starts with the original problem, performs tests after tests as per the decision tree design, and then either resolves the problem or enriches the problem alarm with complete diagnosis, or can even create a trouble ticket automatically.

In case of manual resolution, the operator is presented with a set of problems, the associated services, and a list of the types of devices which can support such services. Once the above triplet is chosen, the corresponding resolutions are displayed, which can be invoked manually.

In UCA Automation System, the process of problem resolution happens in the way depicted by the following diagram. The administrator or integrator of the system has the option to easily configure the decision tree without the need for any kind of programming. The decideand-act subsystems work based on this configuration. In case the administrator needs to make advanced decisions based on the results of the previous tests, the platform allows him to write his own rules in the evaluate block.

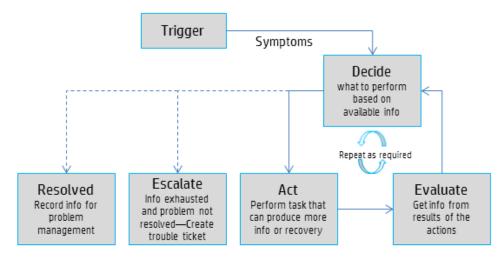


Figure 2 UCA Automation workflow

Accessing UCA Automation

2.1 Launch UCA Automation UI

This section explains the procedure to open the UCA Automation UI.

1. After starting the UCA Automation application, you can open the user interface from the UCA-EBC console by accessing the following URL:

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

- o <hostname or IP address>—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 the localhost as the name of the host to connect to using your web browser.
- o <port #>—Port number of the UCA EBC user interface, which is by default
 8888.
- 2. Click the UCA Automation Foundation VP on the left navigation pane
- 3. Select the UCAAutomation tab on the right-hand side.

The UCA Automation console opens.

🗲) 🕲 hpsatest2.ind.hp.com/8888/uca/#UCA_Automation_Foundati	on_UCA-VI.1-1A:VALUE_PACK	UCAAutomation 🗸 C 🚺 - Google 🖉 🖉 🏚	+ +
UCA for Event Based Correla	tion	Welcome: anonymous (Observer) Login	Help 🔻
UCA-EBC.default		oundation_UCA-V1.1-1A > Value Pack > UCAAutomation tem Treubleshooting UCAvatemation	
UCA_Automation_DomainExample_UCA_EVV1.1-1A UCA_Automation_DomainExample_UCA_PD-V1.1-1A UCA_Automation_Esundation_UCA-V1.1-1A	UCA Automation Monitoring Manual Tests	ALL PALE[0] Watery Enr Operator[0] Period /* them brank /** ** (1) /(1) ** ** 0 ** X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X	2 Ref
Value Pack UCA_Automation_Foundation_UCA requestresponse	Topology View View Reports Settings	Ye kens is show.	
	Auto Refresh ON Global Parameters		

2.2 User interface layout

The Monitoring window, which is the main page of the UCA Automation UI, shows various Actions and their statuses. The Actions are grouped by Originator and sorted by the Originator and the ID columns.

By default, the page is displayed for an Observer role.

hp	UCA for Eve	nt Ba	ased Correla	ation								
	UCA_Automation	Foun	dation_UCA-V	1.1-1A > Value Pa	ck > UCAAu	tomation						
^ 🔶 UC.	Monitoring Confi	guration	Troubleshooting	UCAAutomation								
^ 👘 UC.	UCA Automation		LL FALED	0] Waiting For Operat	or [2]							2 Refr
^ 👘 UG.	Monitoring		Show Details	89 Interact					1	<< << (1)	/(1) >> >>	
🗸 🌒 NC	Manual Tests		D*1	Action Name	Action ID	Problem	Mode	Action Originator	Originator * 1	State	Start Time Service	Resut
0	A			ıca_pbalarm alarm_obj								
1	Topology View	•	120	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca,	Waiting_Operator	12-Apr-14 05:2' MobileServices	
	C View Reports			ica_pbalarm alarm_obj								
	Settings	8	119	test_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_context uca,	Wating_Operator	12-Apr-14 05:21 MobileServices	
	Auto Refresh ON											
	Global Parameters											

Figure 3 UCA Automation User Interface main view

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

To access the UCA Automation UI, contact the UCA EBC administrator who can create a user and associate a role.

On the left panel, the various operations available are Monitoring, Manual Tests, Topology View, and View Reports. You can also set the Global Parameters for the UCA Automation console.

2.3 Logging into UI

Use the following procedure to lo into the UCA Automation UI.

1. Click the **Login** link on upper right corner of the page.

The UCA Automation Console: Login pop-up window appears.

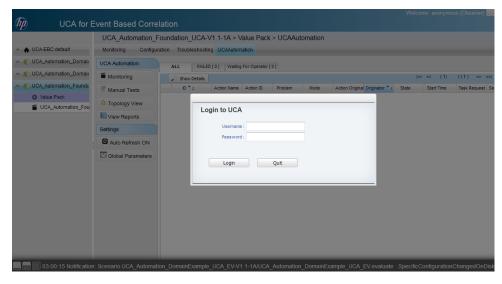


Figure 4 Login panel

When installing the UCA Automation software, the admin user is created with Administrator privileges. The default login credentials are the following:

- o **Username:** admin
- o Password: admin
- 2. Enter the username and password and click the Login button.

After logging in as the administrator, you can create additional users.

Configuring UCA Automation

3.1 Global parameters

In UCA Automation, before the user applies an action and runs a test on an external system, the administrator can decide whether the user can change any settings of an action during the test. The administrator can restrict the user interactions by activating the Global Parameters feature available in the UI.

The available parameters are the following:

- Action mode: This parameter has two attributes.
 - Open Loop— User can interact with an action. With this option, an administrator can decide whether to approve or disapprove an action.
 - Closed Loop—User cannot interact with an action and edit the parameter values for an action.
- Action type:
 - Real—Run the tests in real mode.
 - o Demo—Run the tests in demonstration mode

3.2 Activating Global Parameters

Use the following procedure to activate the Global Parameters feature.

1. Select Settings -> Global Parameters from the menu.

The **Global Parameters** page opens.

	Foundation_UCA-V1.1-1A > Value	Pack > UCAAutomation			
Monitoring Configu	ration Troubleshooting UCAAutomation				
UCA Automation	Action Mode	Action Type		Activate Global Params	
Monitoring	Open Loop	real		true	
🦉 Manual Tests					
🚣 Topology View					
View Reports					
Settings					
Auto Refresh ON					
Global Parameters					
	Update				
		✓ Action [−]	Type : real 🗸	5	Activate Global Params
			Update Reset		

Figure 5 Global Parameters page

2. Select the **Activate Global Params** checkbox.

The Global Parameters feature is activated.

Monitoring

4.1 View status of tests

Use the following procedure to view the status of each test performed.

1. Select **UCA Automation -> Monitoring** from the menu.

The **User Management** page opens.

UCA Automation	ALL	FAILED [2] Waiting For Ope	rator [4]									2 Refre
Monitoring	2 9	how Details							<< <<	(1) /(1)	>> >>		
Manual Tests	0.	2 A	ction Name	Action ID	Problem	Mode	Action Originato	r Originator 🔭 1	State	Start Time	Task Request	Service	Res
A Topology View	🖃 opera	tion_context us	a_pbalarm alarm	object 28									
Topology view	1	24 re	cover_service	106	bsc_interface_down	Open Loop	alarm	operation_contex	Waiting_Operator	13-Apr-14 12:1		MobileServices	
View Reports	E 1	23 list_al_	available_interfaces	105	bsc_interface_down	Open Loop	alarm	operation_contex	Ok	13-Apr-14 12:1	<soapenv:envelo< td=""><td>MobileServices</td><td>Interface</td></soapenv:envelo<>	MobileServices	Interface
Settings	E 1	22 tes	t_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_contex	Failure	13-Apr-14 12:1-	<soapenv:envelo< td=""><td>MobileServices</td><td>PING 10.2</td></soapenv:envelo<>	MobileServices	PING 10.2
-	🖃 opera	tion_context us	ca_pbalarm alarm	_object 26									
Auto Refresh ON	E 1	25 list_al_	available_interfaces	105	bsc_interface_down	Open Loop	alarm	operation_contex	Waiting_Operator	13-Apr-14 12:10		MobileServices	
Global Parameters	1	21 tes	t_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_contex	Failure	13-Apr-14 12:13	<soapenv:envelo< td=""><td>MobileServices</td><td>PING 10.2</td></soapenv:envelo<>	MobileServices	PING 10.2
C Global Parameters	🖃 opera	tion_context u	a_pbalarm alarm;	_object 24									
	1	20 tes	t_bsc_interface	100	bsc_interface_down	Open Loop	alarm	operation_contex	Waiting_Operator	12-Apr-14 06:2		MobileServices	
	🖃 opera	tion_context us	a_pbalarm alarm;	_object 22									
	E 1	19 tes	t_bsc_interface	100	bsc interface down	Open Loop	alarm	operation contex	Waiting Operator	12-Apr-14 06:20		MobileServices	

Figure 6 Monitor view

The page contains the following three tabs which show the status of all tests that are currently running as well as the failed ones in.

- All—Status of all tests that are currently running, failed, awaiting operator's response.
- **Failed**—Lists all tests that have failed.
- **Waiting For Operator**—Lists all tests that are awaiting an action from the operator.

The value displayed next to these tab names represents the number of actions in these tabs. The default view shows the status of the action taken on a specific problem.

Field	Description
ID	Task ID
Action Name	Diagnostic Action taken for a problem.
Action ID	Diagnostic action ID.
Problem	Problem symptom.
Mode	Diagnostic action mode: Whether Open or Closed loop.

Field	Description
Action Originator	The possible values are Alarm and operator.
Originator	The possible values are Alarm ID and Operator ID.
State	The possible values are the following:
	Waiting_Operator
	Request_Sent
	In_Progress
	• Failure
	• 0k
Start Time	The time at which the task started.
Task Request	The request message sent to the activation system.
Service	Service name.
Result	Diagnostic action results.

Table 1 Monitoring

Note

If the status a test is <code>Successful</code>, it is not listed in these lists. The successful reports appear in the **Reports** menu.

2. Select the checkbox for a record and click the **Show Details** button to view the details of that record.

nitoring Config	Juration Troubleshooting UCAAutomation	
int Preview	Prot	t
ID:	122	
Action Name:	test_bsc_interface	
Action ID:	100	
Problem:	bsc_interface_down	
Mode:	Open Loop	
Action Originator:	alarm	
Originator:	operation_context uca_pbalarm_object 28	
State:	Falure	
Status:	FALED	
Action Request:	-Amir vasori 11° ecodes/20172° Bandales/yet7- emay xmla-fib/lip/se su casadomizin hg com?- elader - Actor Spaquell Organder-Natr Organization, Spaquell Spaquell Organder-Natr Organization, Spaquell Spaquell Spaquell Organder-Natr Organization, Spaquell Spaque Spaquell Spaquell S	
Task Response :	-Online service 1.1 # ecoding-1117.** # Intendition-1147.** # I	
Start Time:	4/13/2014, 12:14:38 AM	
End Time:	4/13/2014, 12:15:15 AM	
Service:	MobileServices	
Task Request:	sagen: Christe antis sagen" http://christe.stmisar.go/page/net/ege/" antis mitig/types vs.usadonation to com" sagener interdim caspen (sagener) interdim caspen) inter antisonation (sagener) interdim caspen) interdim caspen (sagener) interdim caspen) interdim caspen) interdim caspen) interdim caspen (sagener) interdim caspen) interdim caspen) interdim caspen) interdim caspen (sagener) interdim caspen) interdim caspen (sagener) interdim caspen) interdim caspendim caspen) interdim caspen) interdim caspen) interdim cas	ly>

Figure 7 Monitor show details

3. (Optional) Hover mouse over the record's **Result** cell to view the results.

UCA Automation	AL	L FAILED	[2] Waiting For	Operator [3]									💦 Refre
Monitoring		Show Details							jes.	<< (1)	/(1) >>	**	
Manual Tests		D*:	Action Name	Action ID	Problem	Mode	Action Originator	Originator * 1	State	Start Time	Task Request	Service	Result
	🖃 оре	eration_context	uca_pbalarm ala	rm_object 28									
A Topology View		124	recover_service	106	bsc_interface_dov	Open Loop	alarm	operation_context	Waiting_Operator	13-Apr-14 12:1		MobileServices	
View Reports	V	123	list_al_available_ir	105	bsc_interface_dov	Open Loop	alarm	operation_context	Ok	13-Apr-14 12:1	<pre><soapenv:envelop< pre=""></soapenv:envelop<></pre>	MobileServices	Interface IP-Addres
		122	test_bsc_interface	100	bsc_interface_dov	Open Loop	alarm	operation_context	Failure	13-Apr-14 12:1	<soapenv:envelop< td=""><td>MobileServices</td><td>PING 10 De control</td></soapenv:envelop<>	MobileServices	PING 10 De control
Settings	🗏 оре	eration_context	uca_pbalarm ala	rm_object 26									IP-Address
Auto Refresh ON		125	list_all_available_ir	105	bsc_interface_dov	Open Loop	alarm	operation_context	Waiting_Operator	13-Apr-14 12:1		MobileServices	Method Star Protocol
Global Parameters		121	test_bsc_interface	100	bsc_interface_dov	Open Loop	alarm	operation_context	Failure	13-Apr-14 12:1	<soapenv:envelop< td=""><td>MobileServices</td><td>PING 10 Ethernet1/0</td></soapenv:envelop<>	MobileServices	PING 10 Ethernet1/0
Giobal Parameters	😑 оре	eration_context	uca_pbalarm ala	rm_object 24									192.168.13 YES unset
		120	test_bsc_interface	100	bsc_interface_dov	Open Loop	alarm	operation_context	Waiting_Operator	12-Apr-14 06:2		MobileServices	Ethernet1/1
													unassigned
													administrati

Figure 8 Monitor view result

4.2 Updating the Waiting for Operator records

If a test shows the status as ${\tt Waiting}\ {\tt for}\ {\tt Operator},$ the operator should provide inputs to complete that test successfully.

- 1. Select the Waiting For Operator tab.
- 2. Select the checkbox for the required record and click the Interact button.

The Specify parameters for Action: <Name of the test> window opens.

- 3. Enter the required inputs if the field is open for editing.
 - If the values are pre-configured or the parameter has a default value, the field displays those values. You can edit the value before approving it.
 - If the value is pre-configured and restricted from editing, the field is displayed as non-editable and you cannot edit it before approving.
- 4. Click Approve if the value is correct or click Disapprove.

hp	Specify parameters for Action: test_bsc_interface
ip address of the interface	ip address configured on the interace
	Approve Disapprove

Figure 9 Monitor interact page

4.3 Managing Auto refresh

When a new event occurs, the records are displayed immediately at runtime on the **Monitoring** page.

The page is automatically refreshed to fetch the new records. You can manage refreshing the page in the following two ways:

- Configure the UI_AUTO_REFRESH_INTERVAL attribute in the
 \$ {UCA_EBC_DATA}/instances/<your
 instance>/deploy/UCA_Automation_Foundation_UCA-V1.1 1A/conf/UCAAutomation.properties.
- By default the auto refresh timer is set to 10 seconds (10000ms).
- Click the Refresh button on the right-hand side of the Monitoring page.
- You can also toggle the Auto Refresh value by clicking the Settings -> Auto Refresh option.

Starting tests manually

Manually, you can start an Action or a test using this interface. An action can be performed on a selected resource for a specific service type or a problem.

1. Select **UCA Automation** -> **Manual Tests** options from the menu.

The page displays a list of Service Type problems and corresponding actions.

You can populate the values on this page using the UCA Automation inventory system.

	Foundation_UCA-V1.1-1A > Value Pack ration Troubleshooting UCAAutomation	<pre>c > UCAAutom</pre>	ation							
CA Automation	Service Type	D.*	Action Name	Action Type	Description	Action Mode	Output Parser	Dispatch Type	TT Action	Alarm Action
Monitoring	MobileServices	-	10001110	, total type		oblem to load corres		Dispaton type		
Manual Tests										
Topology View										
View Reports										
ttings										
Auto Refresh ON										
Global Parameters										
	Problem Setect a Service to bad corresponding Problems									
						Start	Test Reset			

Figure 10 Manual Tests main page

2. Click the service you want from the list in the Service Type pane.

All the associated Problems for the selected service are listed in the **Problem** table.

3. Select a problem from the list.

For the selected service and problem combination, all the possible actions are listed in the **UCA Actions** table.

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

Field	Description
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 2 UCA Actions field descriptions

4. Select the checkbox for the appropriate action that you want to run and click the **Start Test** button.

CA_Automation_F	ound	dation_UCA-V1	1-1A > Value P	ack > U	CAAutor	nation							
nitoring Configur	ation	Troubleshooting	UCAAutomation										
A Automation		Service Type			D*	Action Name	Action Type	Description	Action Mode	Output Parser	Dispatch Type	TT Action	Alarm Action
Monitoring		MobileServices			100	test_bsc_interfac		test_bsc_interfa		regex	HPSA		
Manual Tests													
Topology View													
View Reports													
tings													
Auto Refresh ON													
Global Parameters													
		Problem											
		bsc_interface_down											
										1			
									Start	Test Reset			

Figure 11 Manual Tests – Selected action for a problem

After the test starts, the task waits for the user interaction which is the <code>Request_Sent</code> state and then moves to <code>In_Progress</code> state, and later to other states.

You can monitor the progress of the tests in the **Monitor** page.

Launching Topology view

The Topology View in UCA Automation provides a graphical presentation of the actions performed during a particular test.

1. To open the Topology view, edit the following attributes in the

```
${UCA_EBC_DATA}/instances/<your
instance>/deploy/UCA_Automation_Foundation_UCA-V1.1-
1A/conf/UCAAutomation.properties file.
ucaebc_tomsawyer_port=http://<Neo4J server
hostname>:8888/graphdisplay/?username=root&nodeId=0&profile
=ucaatm
```

2. Click the UCA Automation -> Topology View.

The Topology viewer of UCA-EBC opens with the ucaatm profile that is defined in the \${UCA_EBC_DATA}/instances/<your instance>/deploy/conf/GraphDisplayProfiles.xml file.

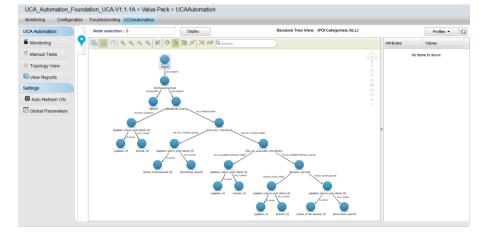


Figure 12 Topology View – launch Neo4j data browser

Generating reports

You can generate the required reports in the **Reports** window.

After a test is run and it displays the status as successful or Failed, this test is moved from the Monitoring window to the Reports window.

1. Select **UCA Automation -> View Reports** from the menu.

The **Reports** page appears with the options to search and generate reports.

UCA_Autor	nation_l	Found	dation_U	CA-V1.1	1-1A > Valu	ue Pack	k > UCA/	Automation								
Monitoring	Configu	ration	Troubles	hooting	UCAAutomatic	n										_
UCA Automatic	00		asic Search			_										
			From Date :			-	To Date :			E	Archive					
Monitoring		~ A0	dvanced Sea													
Manual Tes			Status :	(Any)		~		Typ	e: (Any)		~	Orig	inator : (Any)	~		
Manual Tes	ts		Problem :	(Any)		~		Service Typ	e: (Any)		~					
🚣 Topology V	iew									Search	Re	set				
Ciew Repor	ts												ee ee (1		>> >>	
Settings			ort Report		int Report	_	Archive									
							n Mod		Originator *	State	Status	Action Regue Service			Start Time	End Time
-			D*:	Action Na	me Action ID	Problem	n Moo	e Action Ungin	Unginator	State	Status	Action Reque Service	Task Reques' Task Res	pon: Resur	Start Time	
Auto Refre	sh ON	8	D*:	Action Na	me Action ID	Problem	1 100	e Action Ungir	Unginator	No items		Actual Reduc Service	lask Reques lask Res	pon: Hesut	Start Time	
-			D*:	Action Na	me Action ID	Problem	1 100	e Action Ungir	Unginator			Actual Reduct Service	lask Heques lask Hes	pon Result	Start Time	

Figure 13 Report search criteria

2. Search the reports based on the following criteria.

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
Туре	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
Archive	Searches for archived records, if selected

Table 3 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

Report actions	Description
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 4 Report actions

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

Monitoring Configu	ration Trouble	eshooting UCAAutom	ation												
UCA Automation	V Basic Searc	h													
OCA Automation	From Date			To Date :				Arc	hive						
Monitoring	Advanced S														
Manual Tests	Statu	is: (Any)	~			Type : (Any)	~		Origi	nator : (Any)		~		
Manual Tests	Proble	m: (Any)	~		Service	Type : (Any)	~							
A Topology View							Search		Reset						
View Reports															
E View Reports	Export Report	Print Report		Archive							ec	(1)	/(4)	>> >>	
Settings															
C Auto Refresh ON	D*:	Action Name	Action ID	Problem	Mode	Action Origin	Originator *	State	Status	Action Request	Service	Task Reques	Task Respon	Result	Start Time
Auto Refresh UN		context uca_pbalarm	- '												
Global Parameters	129	recover_service	106	bsc_interface		alarm	operation_co	Ok	PASSED	xml version="1.0</td <td></td> <td></td> <td></td> <td></td> <td></td>					
	128 lis	t_all_available_interfaces	105	bsc_interface	Open Loop	alarm	operation_co	Ok	PASSED	xml version="1.0</td <td>MobileServices</td> <td><soapenv:en< td=""><td><?xml version</td><td>Interface IP-A</td><td>di 13-Apr-14 01:1:1</td></td></soapenv:en<></td>	MobileServices	<soapenv:en< td=""><td><?xml version</td><td>Interface IP-A</td><td>di 13-Apr-14 01:1:1</td></td></soapenv:en<>	xml version</td <td>Interface IP-A</td> <td>di 13-Apr-14 01:1:1</td>	Interface IP-A	di 13-Apr-14 01:1:1
	127	test_bsc_interface	100	bsc_interface	Open Loop	alarm	operation_co	Failure	FAILED	xml version="1.0</td <td>MobileServices</td> <td><soapenv:en< td=""><td><?xml version</td><td>PING 10.20.30</td><td>4 13-Apr-14 01:10</td></td></soapenv:en<></td>	MobileServices	<soapenv:en< td=""><td><?xml version</td><td>PING 10.20.30</td><td>4 13-Apr-14 01:10</td></td></soapenv:en<>	xml version</td <td>PING 10.20.30</td> <td>4 13-Apr-14 01:10</td>	PING 10.20.30	4 13-Apr-14 01:10
	operation_	context uca_pbalarm	alarm_obj	ect 28											
	124	recover_service	106	bsc_interface	Open Loop	alarm	operation_co	Ok	PASSED	xml version="1.0</td <td>MobileServices</td> <td><soapenv:en< td=""><td><?xml version</td><td>00:02:57: %LI</td><td>N 13-Apr-14 12:1:1</td></td></soapenv:en<></td>	MobileServices	<soapenv:en< td=""><td><?xml version</td><td>00:02:57: %LI</td><td>N 13-Apr-14 12:1:1</td></td></soapenv:en<>	xml version</td <td>00:02:57: %LI</td> <td>N 13-Apr-14 12:1:1</td>	00:02:57: %LI	N 13-Apr-14 12:1:1
	📄 123 lis	t_all_available_interfaces	105	bsc_interface	Open Loop	alarm	operation_co	Ok	PASSED	xml version="1.0</td <td>MobileServices</td> <td><soapenv:en< td=""><td><?xml version</td><td>Interface IP-A</td><td>di 13-Apr-14 12:1! 1</td></td></soapenv:en<></td>	MobileServices	<soapenv:en< td=""><td><?xml version</td><td>Interface IP-A</td><td>di 13-Apr-14 12:1! 1</td></td></soapenv:en<>	xml version</td <td>Interface IP-A</td> <td>di 13-Apr-14 12:1! 1</td>	Interface IP-A	di 13-Apr-14 12:1! 1
	122	test_bsc_interface	100	bsc_interface	Open Loop	alarm	operation_co	Failure	FAILED	xml version="1.0</td <td>MobileServices</td> <td><soapenv:en< td=""><td><?xml version</td><td>PING 10.20.30</td><td>.« 13-Apr-14 12:1/1</td></td></soapenv:en<></td>	MobileServices	<soapenv:en< td=""><td><?xml version</td><td>PING 10.20.30</td><td>.« 13-Apr-14 12:1/1</td></td></soapenv:en<>	xml version</td <td>PING 10.20.30</td> <td>.« 13-Apr-14 12:1/1</td>	PING 10.20.30	.« 13-Apr-14 12:1/1
	operation_	context uca_pbalarm	alarm_obj	ect 26											
	126	recover_service	106	bsc_interface	Open Loop	alarm	operation_co	Ok	PASSED	xml version="1.0</td <td>MobileServices</td> <td><soapenv:en< td=""><td><?xml version</td><td>00:02:57: %LJ</td><td>N 13-Apr-14 01:0!</td></td></soapenv:en<></td>	MobileServices	<soapenv:en< td=""><td><?xml version</td><td>00:02:57: %LJ</td><td>N 13-Apr-14 01:0!</td></td></soapenv:en<>	xml version</td <td>00:02:57: %LJ</td> <td>N 13-Apr-14 01:0!</td>	00:02:57: %LJ	N 13-Apr-14 01:0!
	🗐 125 ks	t_all_available_interfaces	105	bsc_interface	Open Loop	alarm	operation_co	Ok	PASSED	xml version="1.0</td <td>MobileServices</td> <td><soapenv:en< td=""><td><?xml version</td><td>Interface IP-A</td><td>di 13-Apr-14 12:1i</td></td></soapenv:en<></td>	MobileServices	<soapenv:en< td=""><td><?xml version</td><td>Interface IP-A</td><td>di 13-Apr-14 12:1i</td></td></soapenv:en<>	xml version</td <td>Interface IP-A</td> <td>di 13-Apr-14 12:1i</td>	Interface IP-A	di 13-Apr-14 12:1i
	121	test bsc interface	100	bsc interface	Open Loop	alarm	operation co	Failure	FAILED	xml version="1.0</td <td>MobileServices</td> <td><soapenv:en< td=""><td><?xml version</td><td>PING 10.20.30</td><td>4 13-Apr-14 12:1:</td></td></soapenv:en<></td>	MobileServices	<soapenv:en< td=""><td><?xml version</td><td>PING 10.20.30</td><td>4 13-Apr-14 12:1:</td></td></soapenv:en<>	xml version</td <td>PING 10.20.30</td> <td>4 13-Apr-14 12:1:</td>	PING 10.20.30	4 13-Apr-14 12:1:

Figure 14 Report default search

- 3. (Optional) Perform the following based on your requirement.
 - **Archive**—Select a record from the search result and click the **Archive** button to archive it.
 - **Print Report**—Click the **Print Report** button to print a report.
 - **Export Report**—Click the **Export Report** button to export the results into CSV format.

7.1 Deleting a report

Use the following procedure to delete a report.

1. Select UCA Automation -> View Reports from the menu.

The Reports page appears with the options to search and generate reports. You can delete only the archived reports.

2. Search the reports with the required criteria and select the checkbox for Archive.

The archived records appear in the bottom panel.

Monitoring Config	uration	Troubles	hooting UCA	Automation												
UCA Automation		sic Search from Date :			То Г	Date :				Archive						
Monitoring	^ Ad	vanced Se	arch													
Manual Tests									Search	Ret	et					
Topology View																
Topology View View Reports																
View Reports	Ехро	rtReport	Print Re	sport	Delete	2						e4	<< (1)	/(1)	>> >>	
View Reports	Expo		Print Re Action Name		Delete	Mode	Action Origin	Originator *	State	Status	Action Reque Service		<< (1) s Task Resp		>> >> Start Time	End Time
View Reports Settings		D * :		Action ID	Problem	Mode	Action Origin	Originator *	State	Status	Action Requir Service					End Time
View Reports Settings Auto Refresh ON		D * :	Action Name A	Action ID	Problem	Mode		Originator *	State	Status	Action Reque Service	Task Reque	s Task Resp	on Result	Start Time	
	Эор	D*: eration_c	Action Name A	alarm alar	Problem m_object 26	Mode Open Loop						Task Reque	s Task Resp n xml versi</td <td>on Result</td> <td>Start Time</td> <td>1:0! 13-Apr-14 01</td>	on Result	Start Time	1:0! 13-Apr-14 01

Figure 15 Report search for archived records

3. Select the checkbox for the report you want to delete and click the **Delete** button.

The report is deleted.

UCA Automation Orchestrator

The UCA Automation Orchestrator is a tool which you can use for defining and modeling the three entities: Service, Problem, Action, and Action Outcome in the form of a decision tree. You can upload this decision tree to a neo4j graph database and use it to identify the Actions to be executed based on the Problem or Action Outcome. The graph can be designed as an N-ary tree. This tool is made available in the form of a plug-in in Eclipse.

The plug-in provides an interface to define these model entities and retain them in the automation inventory of the DB.

8.1 Prerequisites

The following prerequisites apply to installing and using the Automation Orchestrator:

- The HPSA Foundation value pack is deployed.
- The tables in the inventory are created.

For more details, refer to the UCA Automation Installation Guide.

8.2 Create new decision tree project

After installing the UCA Automation Orchestrator plug-in and restarting eclipse, the plug-in is added to the Eclipse tool bar.

1. Click the 🖾 icon in the ribbon.

The Create New Automation Orchestrator Project Wizard appears.

New Automation Orchestrator Project						
Create New Automation Orchestrator Project Wizard						
Create New Automation	Create New Automation Orchestrator Project Wizard					
<u>P</u> roject name:						
File Name						
Inventory DB Driver	com.edb.Driver	•				
Inventory DB URL	jdbc:edb:// <hostname>:<port>/<service></service></port></hostname>	•				
Inventory DB Username	<username></username>					
Inventory DB Password						
Neo4j DB Protocol	http					
Neo4j DB Host	<hostname></hostname>					
Neo4j DB Port	<port></port>					
Neo4j DB	db					
Neo4j DB Data	data					
✓ Use <u>d</u> efault location						
Location: C:\Users\srira	amub\workspace	B <u>r</u> owse				
Choose file system: default 👻						
?	Einish Cancel					

Figure 16 Create New Decision Tree Project Wizard

- 2. Specify the following details:
 - o Project name
 - o File name
 - o Inventory DB Driver
 - Inventory DB URL
 - o Username
 - o Password
 - Neo4j Host
 - o Neo4j Host Port
- 3. Click the **Finish** button.

The UCA Automation Orchestrator supports two databases: Enterprise DB and Oracle. The project is created with the following folder structure.

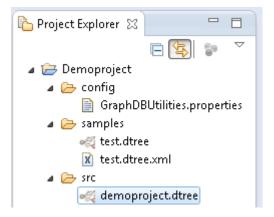


Figure: UCA Automation Orchestrator project folder structure

 GraphDBUtilities.properties file—Contains the inventory properties and Neo4j properties.

```
inventory.db.driver=com.edb.Driver
inventory.db.url=jdbc:edb://<hostname>:<port>/<service>
inventory.db.user=<username>
inventory.db.password=<password>
neo4j.db.protocol=http
neo4j.db.host=<hostname>
neo4j.db.port=<port>
neo4j.db.ob=db
neo4j.db.data=data
org.neo4j.rest.batch_transaction=true
```

Parameter	Description
inventory.db.driver	Inventory database driver name.
inventory.db.url	Inventory database URL.
inventory.db.user	Inventory database username.
inventory.db.password	Inventory database password.
neo4j.db.protocol	The neo4j graph database protocol. By default, the value is http.
neo4j.db.host	The neo4j graph database hostname.
neo4j.db.port	The neo4j graph database port.
neo4j.db.db	Default value is db.
neo4j.db.data	Default value is data.
org.neo4j.rest.batch_transaction	Default value is true.

• . dtree file—Created under the src directory. This file contains the Automation Orchestrator.

When defining the model objects in the inventory, the system uses the Inventory database configuration. When you want to upload the decision tree to neo4j graph database, the system uses the neo4j graph database configuration.

8.3 UCA Automation Orchestrator UI layout

The following picture shows the UCA Automation Orchestrator main screen.

The following functions are available on the UI:

- Definition button—Automation definition view where you can define the model objects.
- Orchestration button—An editor using which you can define the decision tree.
- Deployment button— You can upload the decision tree to Neo4j graph database. This button is enabled only after the decision tree is orchestrated and saved. By default the button is disabled.

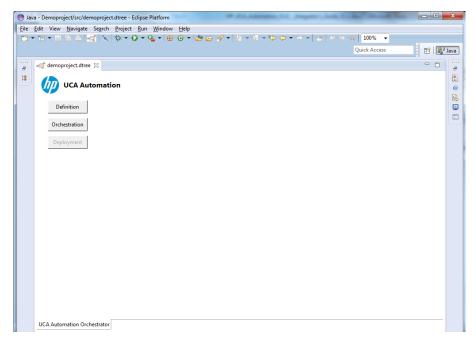


Figure 17 UCA Automation Orchestrator main page in Eclipse

Note

Use the Eclipse Console View to see the errors and log messages.

8.4 Define Model Objects

- 1. Click the **Definition** button.
- 2. The Automation Definition View with the following three tabs appears.
 - Service Definition tab—To define Service types.
 - **Action Definition** tab—To define the Actions.
 - **Problem Definition** tab—To define the Problems.

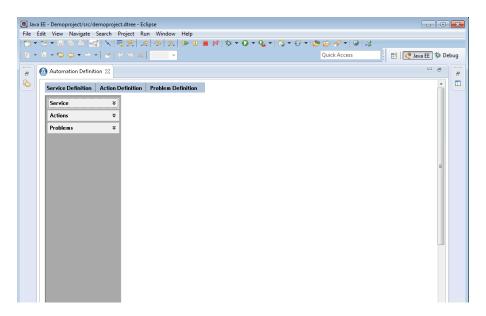


Figure 18 Automation Definition View

8.4.1 Create a new service type

Follow this procedure to create a new Service type.

1. Click the **Definition** button.

The Automation Definition View appears.

2. Click the **Service Definition** tab.

The **CREATE A SERVICE** pane opens.

		moproject.dtree - Eclipse Search Project Run Window Hel	p		
102 -	12 - H R A 🜌	' X 🗟 🛠 Le 👁 B 🕩		₨ ▾ ଓ ▾ 😕 🗁 🛷 ▾ 💩 🖧 –	
12 -	$a \star c + \star c \to \star$	≝ 4/ % X 💿 🔻		Quick Access	😰 😤 Java EE 🕸 Debug
8	🔄 Console 🙆 Automa	ation Definition 🔀			- e -
b	Service Definition	Action Definition Problem Definit	ion		
	Service	×	CREATE A SERVICE		
	Actions	*			
	Problems	Service Type ★			
			Create		
					E
	•		m		

Figure 19 Create a new service

- 3. Enter a name for the Service type in the text box.
- 4. Click the **Create** button.

8.4.2 Create a new problem

The following procedure shows how to create a new Problem.

1. Click the **Definition** button.

The Automation Definition View appears.

2. Click the **Problem Definition** tab.

The **CREATE A PROBLEM** pane opens.

_		E - Demoproject/src/o					
				roject Run Window Help			
					📕 🕅 🎋 * 🔘 * 💁 * 🖏 * 😚 * 🖉 🗁 🔗 * 🚱 🖧		
1 2	- 1	$S \rightarrow \leftrightarrow \leftrightarrow \rightarrow$	• 🖻 🗸	/ 🌭 🗶 📔 👻	Quick Access	📑 😰 😵 Java EE 🏂 I	Debug
é	, G	🛛 Console 🕢 Auto	mation Defin	nition 🛙		- 8	
ľ	5	Service Definition	Action De	efinition Problem Definition		^	
		Service	¥		CREATE A PROBLEM		
		Actions	×				
		Problems	¥	Problem Name *			
				Service Name *	ADSL 👻		
						=	
					Create		
						_	
		•					

Figure 20 Create a new Problem

- 3. Specify the name of the problem in the **Problem Name** text box.
- 4. Select the service type for which the problem is being created, from the **Service Name** drop-down list and click the **Create** button.

8.4.3 Create a new action

The following procedure shows how to create a new Action.

1. Click the **Definition** button.

The Automation Definition View appears.

2. Click the **Action Definition** tab.

The **CREATE AN ACTION** pane opens.

1983 I.S.		- Demoproject/src/demoprojec	t dteres - Fallings		
		: View Navigate Search P			
				■ 14 ☆ - 0 - 4 - 3 - 6 - 2 - 4 - 9 4	
					🗈 😰 Java EE 🕸 Debug
8		Console 🕢 Automation Defi	inition 🛛		
<u></u>		Service Definition Action D	efinition Problem Definition		≙
		Service ¥		CREATE AN ACTION	
		Actions ¥	Action Name *		
		Problems ¥	Description		
			Туре	recover/resolve	
			Action Mode	Open Loop 🔹	=
			Output Parser	None	
			Cost		
			Dispatch Type	Trouble Ticket	
			Trouble Ticket Action	create_tt •	
			Creat	e	
	•		1	III	

Figure 21 Create a new Action

- 3. Enter the required details in the **CREATE AN ACTION** pane.
 - Action Name—Name of the action.
 - Description—Enter a description for the action.
 - Type—Select an action type from the drop-down list, which contains recover/resolve, test, audit, read-only-test, internal, and escalate options.
 - Action Mode—Select an option from the drop-down list, which contains Open Loop, Closed Loop, or None options.
 - OutputParser—Select an option from the drop-down list, which contains None, regex, and xpath options.
 - o Cost
 - Dispatch Type—Select an option from the drop-down list, which contains HPSA, OO, ShellScript, Alarm, and Trouble Ticket options.
 - Trouble Ticket Action—Select the Trouble Ticket Action from the drop-down list.
 - Alarm—Select the alarm action from the drop-down list.
- 4. Click the **Create** button to create a new action.

When you create an Action, the system creates two Action Outcomes by default. The default Action outcomes contain the Action name with the following suffixes <code>_passed</code> and <code>failed</code>.

An example of the formats is <ActionName>_passed and <ActionName>_failed.

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.

Action Type	Description
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 5 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 the value is set to <code>None</code> , the value set in the Parameters -
	> Global Parameter -> ActionMode is used.

Table 6 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.

Table 7 Output Parsers

Dispatch Type	Description
HPSA	Action is dispatched to HP Service Activator for execution.
00	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 are creating alarms and updating alarms.
Trouble Ticket	An internal trouble ticket action.
	Possible operations are creating Trouble Ticket, associating a trouble ticket, dissociating a trouble ticket, and closing a trouble ticket.

Table 8 Dispatch Types

8.5 Update or delete model objects

8.5.1 Modify or delete a Service type

The following procedure shows how to update the name of a service type or delete the Service type.

1. Click the **Definition** button.

The Automation Definition View appears.

2. Click the **Service** tab or the *icon* on the left-hand side.

The **Service** tab expands and shows the configured service types.

3. Select a service type.

The **UPDATE A SERVICE** pane appears. All the configured problems associated with the service type are displayed under the **Problems** tab. You can see them by clicking the **Problem** tab.

Java EE - Demoproject/src/demoproject ile Edit View Navigate Search Pr	roject Run Window Help				×
ar ar a sa s		∎ ¥ ☆ • Q • Q •] 73 • 63 •	Quick Access	🖹 🖹 😭 Java EE 🏇 Di	ebug
Console Automation Defin					8
Service Demittion Action of Actions and Ac	Service Type * PTW Update	UPDATE A SERVICE		5	
•		m		Þ	

Figure 22 Update a Service

- 4. Modify the name in the Service Type text box.
- 5. Click the **Update** button to update the service type or click the **Delete** button to delete the service type.

8.5.2 Modify or delete an Action

The following procedure shows how to update or delete an Action.

1. Click the **Definition** button.

The Automation Definition View appears.

2. Click the **Actions** tab or icon on the left-hand side.

The Actions tab expands and shows the configured Actions.

3. Select the action you want to modify from the actions listed under the **Action** tab.

The **UPDATE AN ACTION** pane appears.

💓 Java EE - Demoproject/src/demoproject.dt			
File Edit View Navigate Search Proje			
		I■₩ ☆ ▼ Q ▼ Q ▼ ☆ ▼ ☆ ▼ ☆ ▼ ☆ ▼ ☆ 2	_
$ \underline{a} + \overline{a} + \overline{c} + \overline{c} + \overline{c} + \overline{c} \underline{a} \mathcal{A} \rangle$	> ×	Quick Access 🗈 😰 Java EE 🕸	Debug
🕝 📮 Console 🙆 Automation Definiti	ion 🕱	- <i>e</i>	
Service Definition Action Defi	inition Problem Definition		
Service *		UPDATE AN ACTION	
ADSL Cable_Service LTE	Action Name *	terminate_alarm	
MobileServices MPLS_VPN PTW	Description		
RAN_REM	Туре	recover/resolve	
Actions checkHeadEnd	Action Mode	Open Loop 👻	
create_tt dissociate_tt/close_tt	Output Parser	None 👻	
list_all_avail_interface LTE_RECOVER Perform_DetermineCutPoint	Cost		
RAN_RECOVER recover service	Dispatch Type	Alarm	
RemediateHeadEnd remediate_TTP_Failure terminate_alarm	Alarm Action	terminate_alarm •	
test_bsc_interface update_alarm_check_tt update_tt	Upda	Delete	
Problems 🌣			
Determine_Cut_Point PostCheck PreCheck			
•		III >	

Figure 23 Update an Action

- 4. Modify the required fields on this pane.
- 5. Click the **Update** button to update the action or click the **Delete** button to delete the action.

When you delete an action, all action outcomes and all parameters associated with the action are deleted.

8.5.3 Modify or delete a configured Problem

The following procedure shows how to update or delete a Problem.

1. Click the **Definition** button.

The Automation Definition View appears.

2. Click the **Problems** tab or the icon on the left-hand side.

The **Problems** tab expands and shows the configured Problems.

3. Select the problem you want to modify from the problems listed under the **Problems** tab.

The **UPDATE A PROBLEM** pane appears.

If you did not select a service type before trying to expand the **Problems** tab, the Please select a Service first! message appears.

a EE - Demoproject/src/o Edit View Navigate		ltree - Eclipse ject Run Window Help		-	×
 📃 Console 🙆 Auto	mation Definit	tion 🛙			
Service Definition	Action Def	finition Problem Definition		_	
Service ADSL Cable,Service LTE MPLSyVPN PTW RAN,REM Actions Problems (sic_interface_down unknown_symptor		Problem Name * Service Name *	UPDATE A PROBLEM bsc_interface_down MobileServices		
•	P.				

Figure 24 Update a Problem

- 4. Modify the name of the problem and click the **Update** button to update the problem.
- 5. (Optional) Click the **Delete** button to delete the problem.

8.6 Define parameters

8.6.1 Create a parameter

The following procedure shows how to create a parameter.

1. Click the **Definition** button.

The Automation Definition View appears.

2. Click the **Actions** tab or ***** icon on the left-hand side.

The **Actions** tab expands and shows the configured Actions.

3. Select an Action for which you want to create a parameter, right-click and select the **Create a Parameter** option from the pop-up list.

The **CREATE A PARAMETER** pane opens.

	Project Run Window Help	■ \$\$ \$\$ • 0 • % •]\$\$ • 6 •]\$ © @ @ \$\$ •] @]\$	
] • ∰ • (+ + + + ≤ [<	×	Quick Access	🖹 😭 Java EE 🕸 Debug
📮 🔁 Console 🔕 Automation Def	inition 🛙		
Service Definition Action E	Perinition Problem Definition		
Service \$ ADSL Cable_Service LTE		CREATE A PARAMETER	
MobileServices MPLS_VPN	Parameter Name 🛛 *		
PTW RAN_REM	Type *	String	
Actions *	Label		E
checkHeadEnd create_tt dissociate_tt/close_tt list all avail interface	Description		
LTE_RECOVER Perform_DetermineCutPoint RAN_RECOVER	Default Value		
recover_service RemediateHeadEnd	Editable?		
remediate_TTP_Failure terminate_alarm test_bsc_interface update alarm check tt	Action Name	test_bsc_interface v	
update_alarm_check_tt update_tt	Parameter Type 🛛 *	Input 👻	
Problems 🌣			
Determine_Cut_Point PostCheck PreCheck	Cre	ate	
•		III	

Figure 25 Create a Parameter

- 4. Enter the required information in the fields.
 - o Parameter Name
 - \circ $\hfill Type—The available values are String, Integer, Float, and Boolean$
 - Parameter Type—Select a value from the drop-down list. The available values are Input and Output.
- 5. Click the **Create** button to create the parameter.

8.6.2 View and update existing parameters

The following procedure shows how to view and update an existing parameter for an action.

1. Click the **Definition** button.

The Automation Definition View appears.

2. Click the **Actions** tab or the **i**con on the left-hand side.

The **Actions** tab expands and shows the configured Actions.

3. Select an Action for which you want to modify the parameter, right-click and select the **View Parameters** option from the pop-up list.

The **LIST OF PARAMETERS** pane opens with the existing input and output parameters listed separately.

Service *		
Service ADSL	I	IST OF PARAMETERS
Cable_Service	INPUT	OUTPUT
LTE MobileServices	interface_ip_address	
MPLS_VPN		
PTW		
RAN_REM		
Actions		
checkHeadEnd create_tt	U	PDATE A PARAMETER
dissociate_tt/close_tt		
list_all_avail_interface LTE_RECOVER	Parameter Name 🔺	interface_ip_address
Perform_DetermineCutPoint	Type *	
RAN_RECOVER recover service	Type	String •
RemediateHeadEnd	Label	
remediate_TTP_Failure terminate_alarm		
test_bsc_interface update_alarm_check_tt	Description	IP Address
update_alarm_cneck_tt		
Problems *	Default Value	0.0.0.0
Determine_Cut_Point		
PostCheck	Editable?	
PreCheck		
	Action Name	test_bsc_interface v
	Parameter Type *	Input
	Parameter Type	where .

Figure 26 List of parameters

4. Select the parameter you want to modify.

The **UPDATE A PARAMETER** pane appears.

- 5. Update the required fields for the parameter and click the **Update** button.
- 6. The parameter is updated.

8.6.3 Delete a parameter

The following procedure shows how to delete an already created parameter for an action.

1. Click the **Definition** button.

The Automation Definition View appears.

2. Click the **Actions** tab or the **i**con on the left-hand side.

The **Actions** tab expands and shows the configured Actions.

3. Select an Action for which you want to delete the parameter, right-click and select the **View Parameters** option from the pop-up list.

The **LIST OF PARAMETERS** pane opens with the existing input and output parameters listed separately.

10.00							
💽 Jav	Java EE - Demoproject/src/demoproject.dtree - Eclipse						
File	Edit	View Navigate Search P	roject Run	Window Help			
		(-					
8	٢	Console 🙆 Automation Defi	inition 🛙				
		Service			LIST OF PARAMETERS		
		ADSL					
		Cable_Service		INPUT	OUTPUT		
		LTE					
		MobileServices	In	erface_ip_address			
		MPLS_VPN					
		PTW RAN REM					
		KAN_REM					
		Actions \$					
		checkHeadEnd		U	JPDATE A PARAMETER		
		create_tt					
		dissociate_tt/close_tt					
		list_all_avail_interface LTE_RECOVER		Parameter Name 🔺	interface ip address		
		Perform DetermineCutPoint					
		RAN RECOVER		Type *	String		
		recover service					
		RemediateHeadEnd		Label			
		remediate_TTP_Failure					
		terminate_alarm					
		test_bsc_interface update_alarm_check_tt		Description	IP Address		
		update_tt					
		aposto_tt		Default Value	0.0.0		
		Problems 🛠		Derduit Volde			
		Determine Cut Point					
		PostCheck		Editable?			
		PreCheck					
				Action Name	test_bsc_interface 👻		
				Action Name			
				Parameter Type 🛛 *	Input -		
				Update	Delete		

Figure 27 List of parameters

4. Select the parameter you want to delete.

The **UPDATE A PARAMETER** pane appears.

5. Click the **Delete** button.

The parameter is deleted.

8.7 Create an Action Outcome

The following procedure shows how to view the already created Action outcomes for an Action.

1. Click the **Definition** button.

The Automation Definition View appears.

2. Click the **Actions** tab or the **i**con on the left-hand side.

The **Actions** tab expands and shows the configured Actions.

3. Select an Action for which you want to create an action outcome, right-click and select the **Create an Action Outcome** option from the pop-up list.

The **CREATE AN ACTION OUTCOME** pane opens.

	project/src/demoprojec Navigate Search F	t.tdree-Eclipse Project Run Window Help		×
🔐 📮 Consol	e 🙆 Automation Def	inition 🛛		8
Service	Definition Action D	Definition Problem Definition	-	
Servi	e ¥	CREATE AN ACTION OUTCOME		
Actio	15 \$			
check create dissoc list_all LTE_R Perfor RAN_I recov Reme remec termin	HeadEnd tt iste_tt/close_tt avail_interface ECOVER m_DetermineCutPoint ECOVER r_service liateHeadEnd iste_TTP_Failure late_TTP_Failure late_TTP_Failure late_TTP_failure _alarm_check_tt tt	Action Outcome Name * Action Name test_bsc_interface + Create	E	
		m	* }	

Figure 28 Create an Action Outcome

4. Enter a name for the action outcome in the **Action Outcome Name** and click the **Create** button.

A new Action outcome is created with the given name.

8.7.1 View and update Action Outcomes

The following procedure shows how to view the already created Action outcomes for an Action.

1. Click the **Definition** button.

The Automation Definition View appears.

2. Click the **Actions** tab or the **i**con on the left-hand side.

The Actions tab expands and shows the configured Actions.

3. Select the Action for which you want to modify an action outcome, right-click and select the **View Action Outcomes** option from the pop-up list.

The **LIST OF ACTION OUTCOMES** pane opens with a list of outcomes for the selected Action in the **ACTION OUTCOMES** box.

	EE - Demoproject/src/demoproject.dtree - dit View Navigate Search Project	•	- • ×
	📮 Console 🙆 Automation Definition 🖇		
<u></u>	Service Definition Action Definition	Problem Definition	ĵ □
	Service ×	LIST OF ACTION OUTCOMES	
	Actions CheckHoadEnd checkHoadEnd creats_tt dissociate_tt//close_tt list_all_avail_interface LTE_RECOVER Perform_DetermineCutPoint RAN_RECOVER recover_service RemediateHeadEnd	ACTION OUTCOMES test_bsc_interface_passed	
	KemediateHeadthd remediate_TTP_Failure terminate_alarm test_bsc_interface update_alarm_check_tt update_tt	UPDATE AN ACTION OUTCOME Action Outcome Name * test_bsc_interface_failed	E
	Problems ¥	Action Name test_bsc_interface v Update Delete	
	•	11	•

Figure 29 Update an Action Outcome

4. Select an Action outcome you want to update from the list.

The **Update an Action Outcome** pane appears. You can modify only the name.

5. Modify the Action outcome name and click the **Update** button.

The Action outcome is modified.

8.7.2 Delete an action outcome

The following procedure shows how to view the already created Action outcomes for an Action.

1. Click the **Definition** button.

The Automation Definition View appears.

2. Click the **Actions** tab or the icon on the left-hand side.

The **Actions** tab button expands and shows the configured Actions.

3. Select the Action for which you want to delete the action outcome, right-click the action and select the **View Action Outcomes** option from the pop-up list.

The **LIST OF ACTION OUTCOMES** pane opens with a list of outcomes for the selected Action in the **ACTION OUTCOMES** box.

		- Demoproject/src/demoprojec t View Navigate Search P			×
8		Console 🙆 Automation Defi		- 8	8
b		Service Definition Action D	efinition Problem Definition	^	
		Service ¥ Actions \$	LIST OF ACTION OUTCOMES		
		checkHeadEnd create_tt dissociate_tt/close_tt list_all_avail_interface LTE_RECOVER Perform_DetermineCutPoint RAN_RECOVER	ACTION OUTCOMES		
		recover_service RemediateHeadEnd remediate_TTP_Failure terminate_alarm <u>test_bsc_interface</u> update_alarm_check_tt update_tt	UPDATE AN ACTION OUTCOME Action Outcome Name * test_bsc_interface_failed	н	
		Problems ¥	Action Name tesit,bsc_interface +		
	•		W	•	

Figure 30 Update an Action Outcome

4. Select the action outcome you want to delete from the list.

The **Update an Action Outcome** pane appears.

- 5. Click the **Delete** button.
- 6. Click **Yes** to confirm deleting the action outcome.

The action outcome is deleted.

8.8 Orchestrating Model Objects

In the UCA Automation Orchestrator UI main window, the **Orchestration** button opens an editor, where you design a decision tree by using the entities from the database (enterprise or Oracle).

8.8.1 Launch Orchestration

1. In the UCA Automation Orchestrator project folder structure, click the .dtree file under the src directory.

The Automation Orchestrator home page opens.

2. In the Automation Orchestrator home page, click the **Orchestration** button.

A palette with the following options appears:

- Select—To select a Problem link or Action Outcome link.
- Root—To draw a root node. In a decision tree, you can have only one root node.
- Service—To draw a Service node. All the Service nodes are linked to the root node at top.
- Action—To draw an Action node.
- Problem—To define links or relationships. The Service nodes are always linked to the Action nodes using Problem link.

• Action Outcome— To define links or relationships. The Action nodes are linked to another Action node by action outcome link.

	🕲 Java EE - Demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject/src/demoproject							
	Edit View Navigate Search Project Run Window Help							
ő	UCA Automation							
	Definition Image: Select Society of the select se							
	UCA Automation Orchestrator							

Figure 31 UCA Automation Orchestration

Each option is associated with an icon. To the right of the palette is the editor.

The editor contains a decision tree with root node as **UCAAutomation** and all Service nodes appear as linked to the root node.

A new file with extension . dree.xml is generated under the src folder. This file contains all Problems, Actions, and the XML format of the skeleton decision tree.

8.8.2 Add a Service to decision tree

The following procedure describes how to add a Service node to a decision tree.

1. In the Automation Orchestrator home page, click the Orchestration button.

The palette with options and the editor appears.

2. Select the **Service** option from the palette and click in the editor.

The service icon is placed in the editor. The link with the root node is established automatically and the **Properties** pane opens with the following two columns:

- o Property
- Value

The serviceName property appears in the Property column.

Java EE - Demoproject/src/demoproject.dtree - Eclip	ose	
File Edit View Navigate Search Project Run	Window Help	
📑 • 🖻 • 🗄 🕲 🗠 😽 🔍 🔍 🖈	🔿 R II = M 🌣 + 🔾 + 🖓 + 😚 + 👉 💋	> 🖋 + ❷ 2≱ ½ + 1½ + 1½ + 1↓ (↓ + → + ∠ ℓ ∨ ⋈
		Quick Access 🛛 😰 😰 Java EE 🕸 Debug
🎦 Project Explorer 🛛 🗖 🗖	🐗 *demoproject.dtree 🔀 📄 GraphDBUtilities.properties	
= 😨 😨 🏹		
a 🥽 Demoproject	UCA Automation	
a 🗁 config		
GraphDBUtilities.properties	Definition Select	
 b b samples a b src 	Root	
a demoproject.dtree	Orchestration Service	
demoproject.dtree.xml	Action	
	Deployment Problem	
	ActionOutcome	
		TTE / MobileServices ADSL
		- 🖌 🔁 🗌
	UCA Automation Orchestrator	
	🔲 Properties 🙁 📮 Console 🕢 Automation Definition	🛃 🖬 🏇 🔍 🍸 🗖 🗖
	Property	Value
	serviceName	
		·
		Þ

Figure 32 Add Service node to orchestrator editor

3. Click the row in the Value column corresponding to the serviceName.

A drop-down list appears with all the service names.

4. Select a service name and click inside the orchestrator editor.

The service name is added to the editor against the Service node. If you do not give a name to the node, the <code>Please specify the Service name!</code> message appears when you save the decision tree.

- 5. (Optional) Select **Edit** from the menu or right-click the node to perform the following operations:
 - Undo— To undo a previous operation, select Undo <operation name>. Multiple undo operations are possible.
 - Redo— The **Redo <operation name>** is enabled after an operation is undone.
 - Zoom In—Click View -> Zoom In from the menu to magnify the decision tree in the editor.
 - Zoom Out—Click View -> Zoom Out from the menu to diminish the decision tree. This option is available only for the decision tree editor.
 - Delete—To delete a node. When you delete a Service node, the Service node and the incoming link and outgoing link are deleted.

Note

You cannot delete the Root node and the link between the Root node and the Service node.

When you delete a Service node, UCA Orchestrator automatically deletes the link to the Root node.

8.8.3 Delete a Service from the decision tree using CLI

You can use the command line utility to delete a specific branch in the decision tree for a Service.

1. Run the following command:

./decisionTree.sh -d MobileServices

2. To Delete all the Services run the following run

./decisionTree.sh -d ALL

8.8.4 Add an Action to decision tree

The following procedure describes how to add an action to a decision tree.

1. In the Automation Orchestrator home page, click the Orchestration button.

The palette with options and the decision tree editor appear.

2. Select the Action option from the palette and click in the editor.

The icon for the Action option is placed in the editor. The **Properties** pane opens with the following two columns:

- o Property
- Value

The actionName property appears in the Property column.

Java EE - Demoproject/src/demoproject.dtree - Eclipse Ie Edit View Navigate Search Project Run Window I	Hala	
<pre>demoproject.dtree 2</pre>	ν. το φ	- <i>g</i>
Definition Orchestration Deployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceployment Ceploym	UCAutomation UCAutomation TTE MobileServices ADSL	
UCA Automation Orchestrator		ren ta ka ⊻ e a
🔲 Properties 🔀 📮 Console 🙆 Automation Definition		1 🖬 🐎 🖾 🔻 🗆 🖬
Property actionName	Value	-

Figure 33 Add an Action node to orchestrator editor

3. Click the row in the Value column corresponding to the actionName.

A drop-down list appears with all the action names.

4. Select an action name and click inside the orchestrator editor.

The action name is added to the editor against the Action node. If you do not give a name to the node, the <code>Please specify the Action name!</code> message appears when you save the decision tree.

- 5. (Optional) Select **Edit** from the menu or right-click the node to perform the following operations:
 - Undo— To undo a previous operation, select Undo <operation name>. Multiple undo operations are possible.
 - Redo— The **Redo <operation name>** is enabled after an operation is undone.
 - Zoom In—Click View -> Zoom In from the menu to magnify the decision tree in the editor.

- Zoom Out—Click View -> Zoom Out from the menu to diminish the decision tree. This option is available only for the decision tree editor.
- Delete—To delete a node. When you delete an Action node, the Action node and the incoming link and outgoing link of the Action node are deleted.

8.8.5 Link Service node to Action node

After you add an Action node, the Service node and Action node are linked using the Problem link. Problems are associated with Services. A service can have more than one Problem associated to it. If you want to associate a problem to two different service types, you should create the problem for each service type.

Follow the procedure to link the Action and Service nodes.

1. In the Automation Orchestrator home page, click the Orchestration button.

The palette with options and the decision tree editor appear.

- 2. Select the **Problem** option from the palette and in the editor, click the Service node and then the Action node.
- 3. Select the **Select** option in the palette and click the problem link to associate the problem name to the link.

The **Properties** pane opens with the following two columns:

- Property
- Value

The problemName property appears in the Property column.

4. Click the row in the Value column corresponding to the problemName.

A drop-down list appears with all problems associated with the Service type.

🔘 Ja	🛿 Java EE - Demoproject/src/demoproject.dtree - Eclipse 💿 🔳 👔							
File	le Edit View Navigate Search Project Run Window Help							
 	✓ *demoproject.dtree ⋈ UCA Automation		- 7					
	Definition Orchestration Deployment Deployment → ActionOutcome	LTE Mobile Services ADSL						
	UCA Automation Orchestrator							
	Properties 🙁 😑 Console 🙆 Automation Definition	년 (1)	≱ ⊠					
	Property	Value						
	problemName	Bic interface down unknown_symptom						

5. Select a value and click inside the orchestrator editor for the value to get updated in the editor.

The Service node is linked to the Action node via the Problem link. If you do not give a name to the node, the <code>Please specify the Problem name!</code> message appears when saving the decision tree.

6. (Optional) Select **Edit** from the menu or right-click the node or link, you can perform the following operations:

- Undo— To undo a previous operation, select Undo <operation name>. Multiple undo operations are possible.
- Redo— The **Redo <operation name>** is enabled after an operation is undone.
- Zoom In—Click View -> Zoom In from the menu to magnify the decision tree in the editor.
- Zoom Out—Click View -> Zoom Out from the menu to diminish the decision tree. This option is available only for the decision tree editor.
- Delete—To delete a node or a link. When you delete a link, the system deletes only the link.

8.8.6 Link Action nodes

You can link two Action nodes using the Action outcome link. Follow the procedure to link the Action nodes.

1. In the Automation Orchestrator home page, click the Orchestration button.

The palette with options and the decision tree editor appear.

- 2. Add and Action node in the editor.
- 3. Select the **Action Outcome** option from the palette and in the editor, click the Action node from which the link should be defined and then the other Action node.
- 4. Select the **Select** option in the palette and click the Action outcome link to associate the action outcome name to the link.

The **Properties** pane opens with the following two columns:

- o Property
- Value

The actionOutcomeName property appears in the Property column.

5. Click the row in the Value column corresponding to the actionOutcomeName.

A drop-down list appears with all action outcomes of the Action from which the link is drawn.

	va EE - Demoproject/src/demoproject.dtree - Eclipse Edit View Navigate Search Project Run Window Help		
# 6	*demoproject.dtree 🛛		- <i>e</i>
	Definition	UCAAutomation UCAAutomation TTE bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic bic bic bic bic bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic_ bic bic bic bic bic bic bic bic	
	UCA Automation Orchestrator		
	🔲 Properties 🕴 📮 Console 🕢 Automation Definition		🖻 📑 🎘 🖾 🔍 🗖 🗖
	Property	Value	
	actionOutcomeName	Test, by: interface failed tast, bs: interface, passed	

Figure 34 Action Outcome link

6. Select the action outcome and click inside the orchestrator editor for the value to get updated in the editor.

The Action nodes are linked via the action outcome link. If you do not give a name to the node, the <code>Please specify</code> the Action outcome name! message appears when you save the decision tree.

- 7. (Optional) Select **Edit** from the menu or right-click the node or link, you can perform the following operations:
 - Undo— To undo a previous operation, select **Undo <operation name>**.
 Multiple undo operations are possible.
 - Redo— The **Redo <operation name>** is enabled after an operation is undone.
 - Zoom In—Click View -> Zoom In from the menu to magnify the decision tree in the editor.
 - Zoom Out—Click View -> Zoom Out from the menu to diminish the decision tree. This option is available only for the decision tree editor.
 - Delete—To delete a node or link. When you delete a link, only the link is deleted.

8.8.7 Save decision tree

1. After drawing the decision tree, click the **Save** button in the UI to save the decision tree.

The **.dree.xml** file is updated. After you successfully save the decision tree, the option for deployment is enabled.

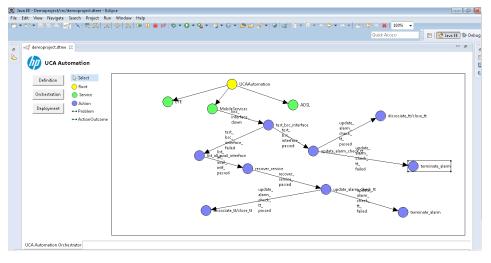


Figure 35 Final Decision Tree

After orchestrating the decision tree, if you click the **Orchestration** button again, a confirmation message appears: Do you want to re-draw the decision tree?

2. Click **Cancel** to save the current graph.

If you click **Ok**, the decision tree skeleton with the Root node and Service nodes is regenerated and the all updates to the decision tree are lost.

8.9 Deploy decision tree

After you deploy the final decision tree, you can view it in the UCA EBC console's **Topology View**.

Follow the procedure to deploy the decision tree into neo4j graph database.

- Specify the Neo4j graph database credentials in the GraphDBUtilities.properties file.
- 2. Click the **Deployment** button to deploy the decision tree.

A confirmation message appears.

If the decision tree contains services without any further links to problems or actions, the orchestrator uploads those services as well.

Hence, if the decision tree contains services without any links to Actions, the confirmation message displays the list of services which do not have any problem linked and the message checks whether you still want to upload the decision tree.

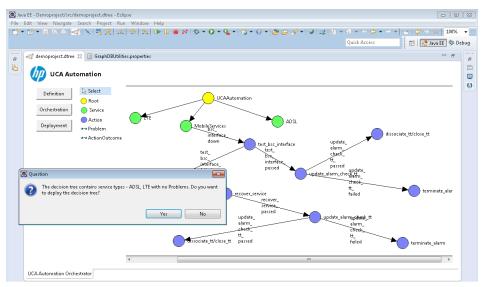


Figure 36 Confirmation message

- 3. Select Yes to upload the decision tree into the neo4j database.
- 4. Log into UCA EBC, after the new graph is uploaded into the neo4j graph database.
- Select UCA-EBC:default -> Topology Management -> Display to view the uploaded graph.

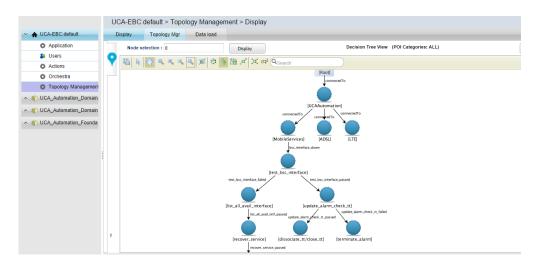


Figure 37 Decision tree after uploading into Neo4j graph database

Note

Only the nodes that have links in the decision tree are saved in the .dtree.xml. Hence, you cannot deploy the nodes which do not have links in the decision tree. However, when you save, the orchestrator editor saves all nodes and you can continue editing the graph.

8.10 Generate skeleton decision tree using CLI

You can Use the command line utility to generate a skeleton decision tree from the database.

- 1. Modify the GraphDBUtilities.properties file to specify the database URL, username, and password.
- 2. Run the following command to generate the skeleton decision tree XML.
 ./decisionTree.sh -e text.xml
- 3. Modify and upload the XML to the Neo4j database.

Modify the XML only if you are familiar with the syntax of the decision tree.

8.10.1 Deploy using command Line utility

You can use the command line utility to deploy the decision tree developed using the automation orchestrator to Neo4j database.

 Modify the GraphDBUtilities.properties file available under the /opt/UCA_Automation/Utilities/DecisionTree/conf to specify the location of the Neo4j host and port database.

```
#UCA Automation Inventory database connection details
#Oracle jdbc driver :
oracle.jdbc.pool.OracleConnectionPoolDataSource
#Oracle url :
jdbc:oracle:thin:@<hostname>:<port>:<service>
#Postgres jdbc driver : com.edb.Driver
#Postgres url : jdbc:edb:@<hostname>:<port>:<service>
inventory.db.driver=com.edb.Driver
inventory.db.url=jdbc:edb://<hostname>:<port>/<service>
inventory.db.url=jdbc:edb://<hostname>:<port>/<service>
inventory.db.user=<username>
inventory.db.password=<password>
#Neo4j database connection details
```

```
neo4j.db.protocol=http
neo4j.db.host=<hostname>
neo4j.db.port=<port>
neo4j.db.db=db
neo4j.db.data=data
#enables batch transaction of inserts into neo4j db
org.neo4j.rest.batch transaction=true
#enables http streaming
org.neo4j.rest.stream=true
```

The default value for the <host> is localhost and <port> is 7474.

2. Copy the Decision Tree XML developed using the orchestrator from projects src folder to any location in the server.

For example, copy the XML file to
/opt/UCA_Automation/Utilities/DecisionTree/etc.

3. Run the following command available at /opt/UCA_Automation/Utilities/DecisionTree/bin.

./decisionTree.sh -u etc/<Name of the decision tree xml>