

Release Automation 6.3 & DevTest Integration

Manjunath Narayan
Engineering Services Architect
SWAT-Advocate Team

Scope

Demonstrate the Release Automation Integration with DevTest

Integration of Release Automation and DevTest enables you to perform Continuous Delivery and Continuous Testing which will help you to identify the defects early on the lifecycle and implement DevOps Best Practices.

DevTest Features

CA Application Test

- Provides an automated testing solution
- design and execute automated tests for multiple layers of a distributed architecture.

CA Service Virtualization

- Removes constraints throughout the SDLC by modeling and simulating unavailable or dependent systems.
- Reduces demand for lab infrastructure and software to avoid costs and reduce configuration effort.

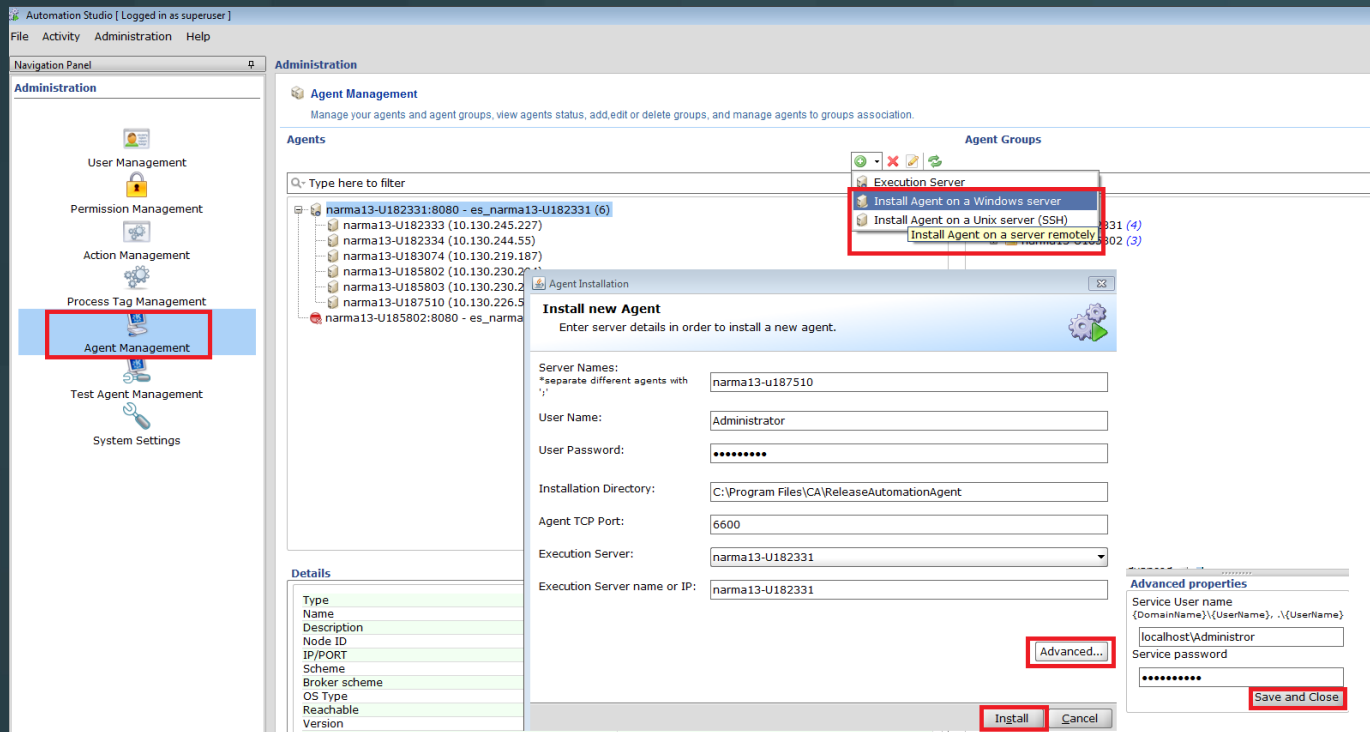
RA integration with DevTest

Procedure

- Install the RA Agent on the Dev Test Server
- Install the DevTest Action Pack
- Adding Action for Service Virtualization
- Adding Action for Application Test
- Sample Flow

Install the RA Agent on the Dev Test Server

From ASAP



Install the RA Agent on the Dev Test Server

From Release Automation (<http://RA Server:8080/nolio-app/>)

The screenshot displays the CA Release Automation web application interface. The browser's address bar shows the URL `http://narma13-u182330:8080/nolio-app/`. The page header includes the CA Technologies logo and a link to explore release planning and management. The main content area is divided into two sections: "Release Operations Center" and "Automation Studio". Below these sections are "Launch" buttons. At the bottom, there is a "Download Agent Setup:" section with a dropdown menu and a "Download" button. The dropdown menu is open, showing a list of operating systems: Windows, Linux x84, Solaris-SPARC, Solaris-x86, AIX, Solaris 9 SPARC, and Solaris 9 x86. The "Agent Setup (Windows)" option is selected and highlighted.

ca technologies

Explore Release Planning & Management
Free license to try CA Release Automation Continuous Delivery Edition [Check it out»](#)

CA Release Automation

Managing enterprise application release operations

Release Operations Center
Central hub to plan, design, execute and track deployments

Automation Studio
Legacy IDE for administration and application deployment modeling
(Automation Studio requires Oracle JRE™ 8 and runs using Java™ Web Start Launcher)


Launch

Download Agent Setup:

- Agent Setup (Windows)
- Agent Setup (Linux x84)
- Agent Setup (Solaris-SPARC)
- Agent Setup (Solaris-x86)
- Agent Setup (AIX)
- Agent Setup (Solaris 9 SPARC)
- Agent Setup (Solaris 9 x86)

Download

Installation of Action Pack (nolio-ca-lisa-actions)

 CA Release Automation

DashboardDesignerReleasesArtifactsEnvironmentsAdministrationDSL

Switch To Express Mode | superuser | Help

Action Management











66* Available Packs
66 new packs available

8* Installed Packs
One updated pack available

Enter search criteria

RefreshDownload

<< < 1 - 10 of 66 > >>

<input type="checkbox"/>	Available Packs	Description	Action List	Version	Date
<input type="checkbox"/>	nolio-actions-f5bigip ★	Nolio F5 Big IP suite actions package		10.5.1410.2	2016-07-26
<input type="checkbox"/>	nolio-actions-vmware ★	Nolio VMWare actions package		10.4.1506.1	2016-04-20
<input type="checkbox"/>	nolio-amazon-actions ★	Nolio Amazon actions package		10.4.1506.1	2015-07-15
<input type="checkbox"/>	nolio-atlassian-jira-actions ★	Nolio JIRA actions package		10.7.1506.1	2015-07-15
<input type="checkbox"/>	nolio-biztalk-actions ★	Nolio Biztalk actions package		10.4.1412.1	2014-12-18
<input type="checkbox"/>	nolio-ca-harvest-actions ★	Nolio CA Harvest actions pack		10.3.1503.1	2015-03-19
<input checked="" type="checkbox"/>	nolio-ca-lisa-actions ★	Nolio CA Service Virtualization actions package		10.9.1504.1	2016-07-31
<input type="checkbox"/>	nolio-ca-servicedesk-actions ★	Nolio CA ServiceDesk actions pack		10.3.1402.1	2014-03-26
<input type="checkbox"/>	nolio-chef-actions ★	Nolio Chef actions pack		10.4.1406.1	2016-11-24
<input type="checkbox"/>	nolio-clearcase-actions ★	Nolio clearcase actions pack		10.3.1411.1	2014-11-04

Installation of Action Pack (nolio-ca-lisa-actions)

- ▲ All Actions (34)
 - ▲ CA Service Virtualization (2)
 - ▲ Application Test (4)
 - SV - Compare Two Test Results
 - SV - Run Model Archives
 - SV - Run Test
 - SV - Run Test Suite
 - ▲ VSE (5)
 - SV - Deploy Virtual Service
 - SV - Redeploy Virtual Service
 - SV - Remove Virtual Service
 - SV - Start Virtual Service
 - SV - Stop Virtual Service
 - ▲ Release Operations Center (13)
 - ▲ Artifacts (23)
 - ROC - Create SVN Artifact

Creating the Action - Deploy Virtual Service

The screenshot displays the CA Service Virtualization interface. On the left, the 'Components' pane shows a tree structure under 'Default Component (2)' > 'Actions (7)' > 'TomCatDeploy (6)'. The 'CarService - Deploy Virtual Service' action is highlighted with a green circle. A context menu is open, showing 'Add Action' as the selected option. Below this, the 'Add Action' dialog is visible, with 'deploy vi' entered in the search field. The 'All Actions' list shows 'SV - Deploy Virtual Service' highlighted. The main pane displays the configuration for 'CarService - Deploy Virtual Service (action) || type: SV - Deploy Virtual Service'. The configuration includes a description, input fields for 'SV Server Name', 'SV Invoke Port', 'SV User Name', 'SV User Password', 'Model Archive (MAR) PATH', and 'Virtual Service Environment'.

Components | **Processes**

Enter search criteria

- Default Component (2)
 - Actions (7)
 - TomCatDeploy (6)
 - CarService - Deploy Virtual Service**
 - CarSe
 - Get W
 - SV - Dv
 - SV - Rce
 - SV - Ra
 - Flows (1) st

Actions Category Options

- Add Action
- New Category
- Copy

Add Action

deploy vi

All Actions

- All Actions (34)
 - CA Service Virtualization (2)
 - VSE (5)
 - SV - Deploy Virtual Service**
 - SV - Redeploy Virtual Service

CarService - Deploy Virtual Service (action) || type: SV - Deploy Virtual Service

Name

CarService - Deploy Virtual Service

Deploy a mar file to VSE as virtual service. (SV version 7.5.1 or higher)

Inputs	Outputs
SV Server Name [String]	
SV Invoke Port [Int]	
SV User Name [String]	
SV User Password [Password]	
Model Archive (MAR) PATH [String]	
Virtual Service Environment [String]	

nama13-u187510

1505

admin

.....

C:\Program Files\CA\DevTest\Projects\Cars-Demo1\VirtualServices\CarService.mar

VSE

Creating the Action - Stop Virtual Service

CarService - Stop Virtual Service (action) || type: SV - Stop Virtual Service

Name

CarService - Stop Virtual Service

Stop a virtual service in a VSE. (SV version 7.5.1 or higher)

Inputs	Outputs
<div>SV Server Name [String]</div> <div>nama13-u187510</div>	
<div>SV Invoke Port [int]</div> <div>1505</div>	
<div>SV User Name [String]</div> <div>admin</div>	
<div>SV User Password [Password]</div> <div>••••••</div>	
<div>Virtual Service Name [String]</div> <div>CarService</div>	
<div>Virtual Service Environment [String]</div> <div>VSE</div>	

Settings for running the Test Case

```
## =====
## IP Spoofing
##
## Interfaces: a comma-separated list
## MAC address (JDK 1.6+), interface n
##
#lisa.ipspoofing.interfaces=12-34-56-7

ds

## =====aces
## Force the use of a native file dial
#lisa.ui.useNativeFileDialog=true

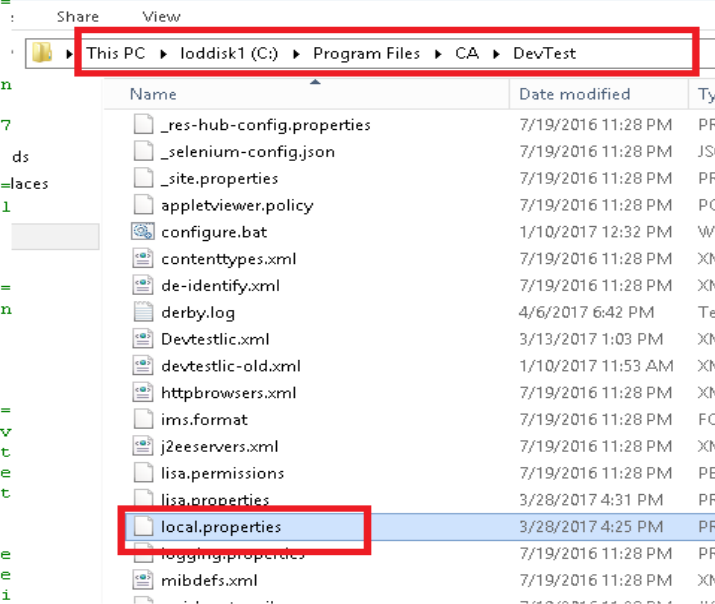
## =====
## Specify the maximum number of recen
## in the Open Recent Quick Start tab
#lisa.prefill.recent=10
#lisa.quickstart.recent=5

##=====lisa-invoke=====
#lisa.portal.invoke.base.url=/lisa-inv
#lisa.portal.invoke.report.url=/report
#lisa.portal.invoke.server.report.dire
#lisa.portal.invoke.test.root=d:/lisat

##=====DevTest Integration Kit example
#asserts=com.mycompany.lisa.AssertFile
#com.mycompany.lisa.AssertFileStartsWi
#filters=com.mycompany.lisa.FilterFileFirstLine
#com.mycompany.lisa.FilterFileFirstLine=com.itko.lisa.editor.FilterController,com.itko.lisa.edito
#nodes=com.mycompany.lisa.node.FTPTestNode
#com.mycompany.lisa.node.FTPTestNode=com.mycompany.lisa.node.FTPTestNodeController,com.mycompany.

# =====
# This property is used as backlog argument while binding Server Socket. Backlog is the requested
# Its exact semantics are implementation specific. In particular, an implementation may impose a
# The value provided should be greater than 0. If it is less than or equal to 0, then by default
# =====
#lisa.serverSocket.bind.backlog=100

#lisa.portal.invoke.test.root={{LISA_HOME}}
```



Creating the Action – Run Test Suite

SV - Run Test Suite (action) || type: SV - Run Test Suite

Name

SV - Run Test Suite

This action runs test from suite file (*.ste) synchronously. User should set the **root** property `lisa.portal.invoke.test.root={{LISA_HOME}}` in **lisa.properties** file before run the action. The paths of files required by this action are relative to the **root**

Inputs

Outputs

Settings

SV Server Name [String]

nama13-u187510



SV Invoke Port [int]

1505

Test Suite Path [String]

examples/Suites/AllTestsSuite.ste



Configuration Path [String]

No Value



Fail If Any Case Failed [boolean]

Default value [false]

Demonstration

Reference:

Release Automation integration with CA Service Virtualization

<https://docops.ca.com/ca-release-automation/integrations/en/optional-action-packs/ca-service-virtualization>

DevTest

<https://docops.ca.com/devtest-solutions/9-5/en>



Manjunath Narayan

Enterprise Service Architect

narma13@ca.com



in