

vRealize Orchestrator Workshop

May 12th 2017 – Viktor van den Berg (PQR)



VMUG
VMWARE USER GROUP

Your Link to the VMware Community

Agenda

- Introduction
- vRealize Orchestrator introduction
- Workflow development basics
 - Lab 1: My first workflow
- Actions & configuration elements
 - Lab 2: Using actions in workflows
- Integration with vCenter Server & vRealize Automation



Introduction

Me, myself and I

- Viktor van den Berg
- Solution Architect / Technology Officer
- Focus: SDDC + Hybrid Cloud
- A&O projects



PQR

- Recent merger with 2ML
- System Integrator + Managed Services
- VMware Premier Solutions Provider
- Activities
 - IT Strategy
 - Projects
 - Managed services



Introduction



- Please do a short introduction
- What's your experience with vRO?
- What's your scripting (PS, python, javascript) experience?
- What are your expectation for this workshop?

vRealize Orchestrator Introduction

Introducing vRealize Orchestrator

- **vRealize Orchestrator is an IT process automation tool**
 - Makes IT operations faster and less error-prone
 - Provides a graphical integrated development environment
 - Facilitates the development of workflows
 - Enables workflow to be exported and imported through packages
- **IT processes must be followed in an organization**
 - Create VM
 - Register VM in AD
 - Update a Change Management Database (CMDB)
 - Optain an IP address for an IP Address Management (IPAM) tool
 - Send out notifications

Introducing vRealize Orchestrator

- **vRealize Orchestrator (latest version is 7.2)**
 - Comes as a stand-alone virtual appliance
 - Is included in vRealize Automation

The screenshot shows the VMware vRealize Orchestrator 7.2.0 download page. The page is titled "Download Product" and includes a sidebar with "Product Resources" such as "View My Download History", "Product Info", "Documentation", and "Community". The main content area is divided into tabs: "Product Downloads", "Drivers & Tools", "Open Source", and "Custom ISOs". Under the "Product Downloads" tab, there is a table with three rows of download links. Each row includes the file name, size, type, and a "Download Now" button. The first row is for the "VMware vRealize Orchestrator Appliance 7.2.0 OVA File" (1.034 GB, .ova). The second row is for the "VMware vRealize Orchestrator Appliance 7.2.0 Update Repository" (809.74 MB, .iso). The third row is for the "VMware vRealize Orchestrator plugin for vCenter 6.5" (37.46 MB, .vmoapp). At the bottom of the page, there are links for "MD5 checksums SHA1 checksums and SHA256 checksums" and "End User License Agreement: Accept EULA".

my vmware

Products Accounts Support

Home / VMware vRealize Orchestrator Appliance 7.2.0

Download Product

Version 7.2.0

Documentation [Release Notes](#)

Release Date 2016-11-22

Type Product Binaries

Product Resources

- [View My Download History](#)
- [Product Info](#)
- [Documentation](#)
- [Community](#)

Product Downloads Drivers & Tools Open Source Custom ISOs

File	Information
VMware vRealize Orchestrator Appliance 7.2.0 OVA File	Download Now
File size: 1.034 GB File type: .ova Read More	Download Manager
VMware vRealize Orchestrator Appliance 7.2.0 Update Repository	Download Now
File size: 809.74 MB File type: .iso Read More	Download Manager
VMware vRealize Orchestrator plugin for vCenter 6.5	Download Now
File size: 37.46 MB File type: .vmoapp Read More	Download Manager

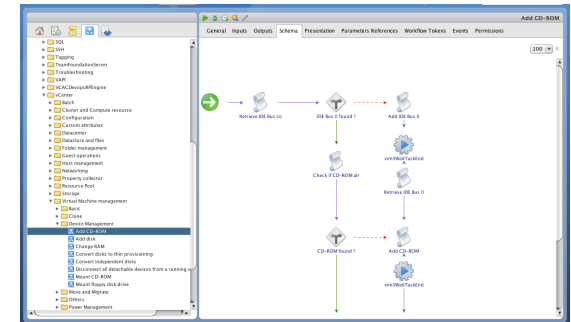
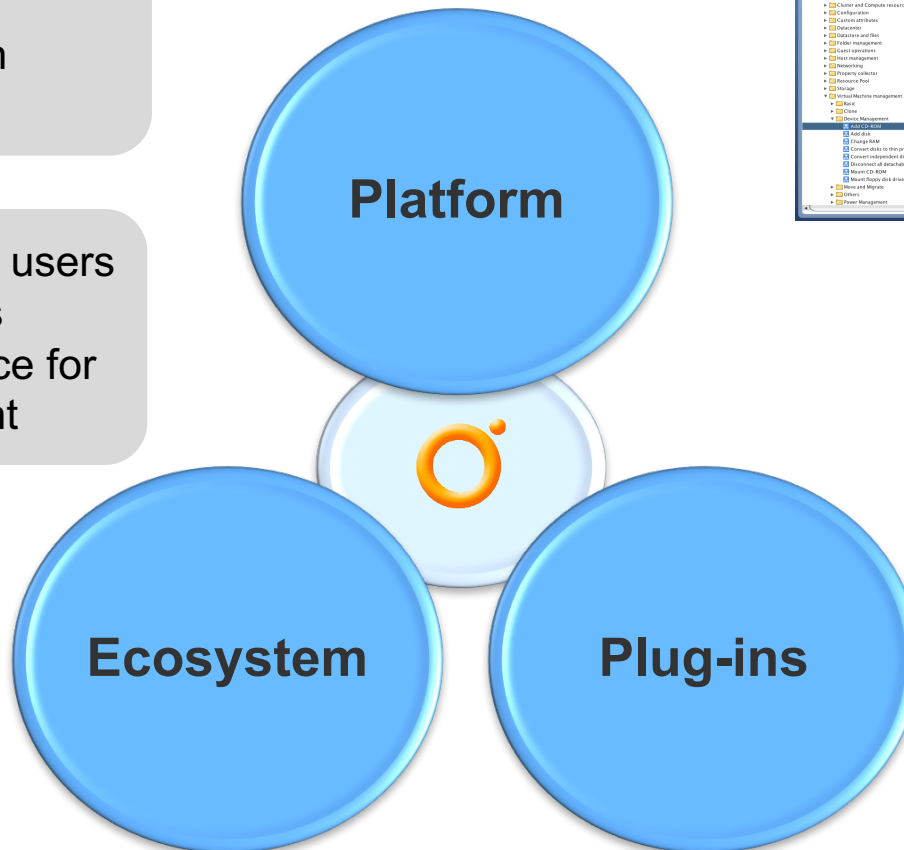
[MD5 checksums](#) [SHA1 checksums](#) and [SHA256 checksums](#).

End User License Agreement: [Accept EULA](#)

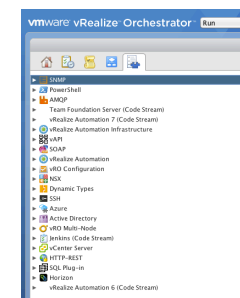
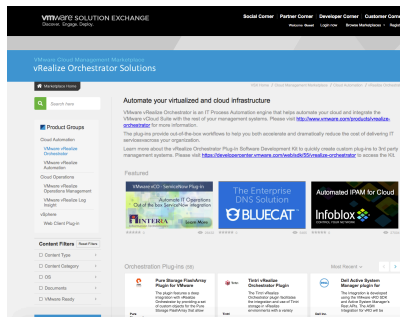
What is VMware vRealize Orchestrator?

- Create, edit, and execute workflows
- Administer solution
- Virtual Appliance

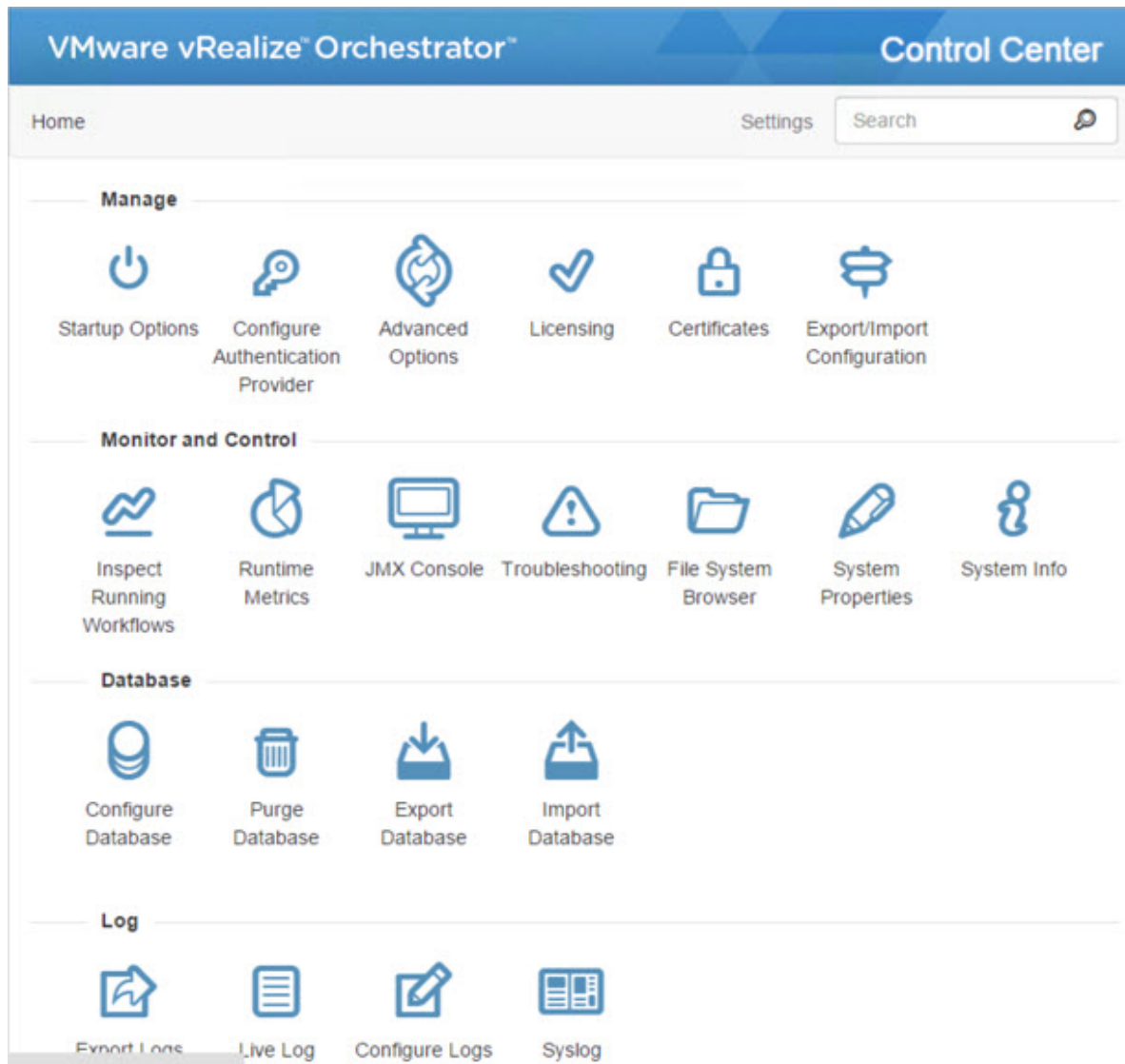
- Allow partners and users to develop plug-ins
- Provide marketplace for hosting that content



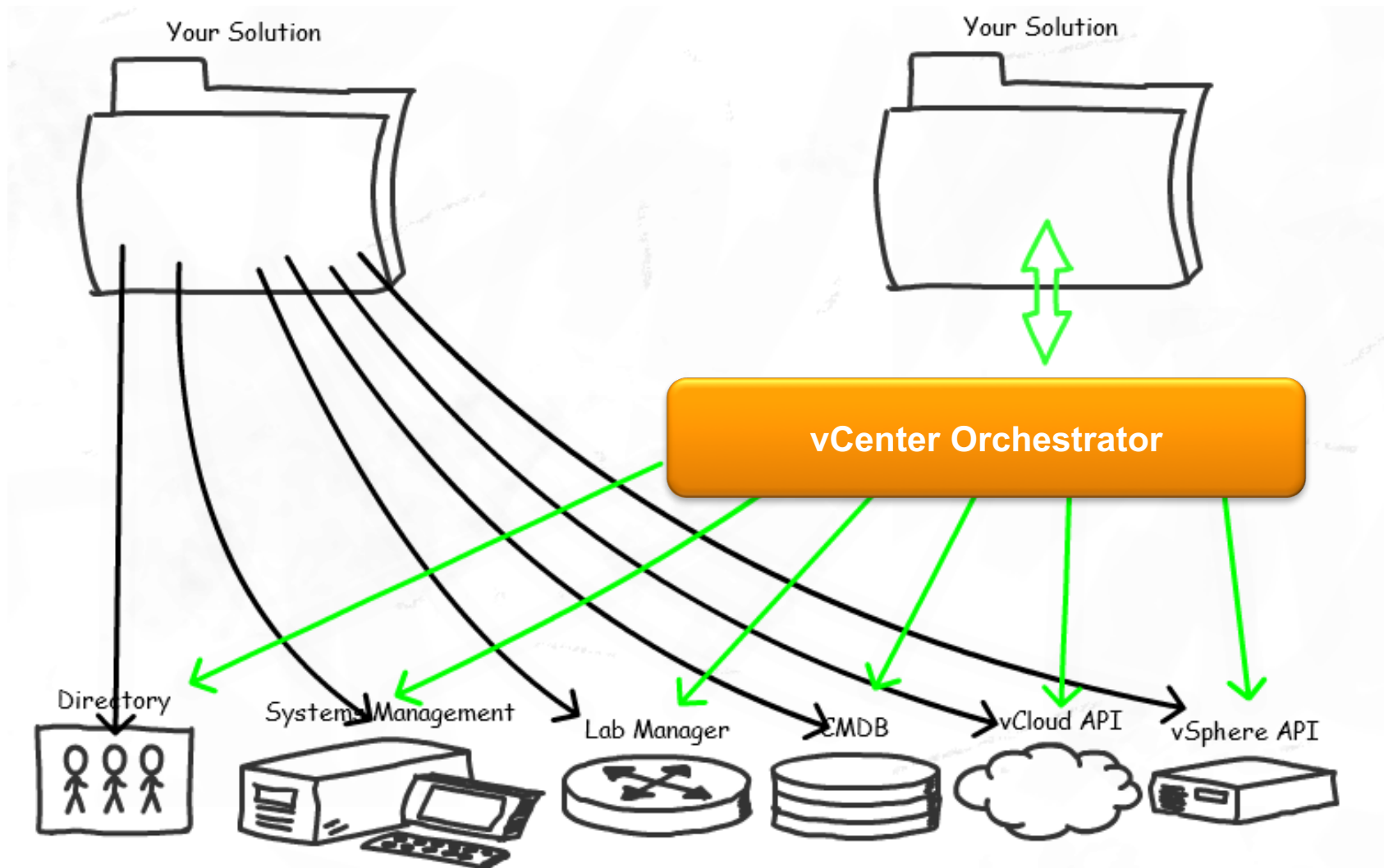
- Integrate VMware stack in your environment
- Prebuilt workflows
- Decoupled from vCO platform releases



vRO Control Center

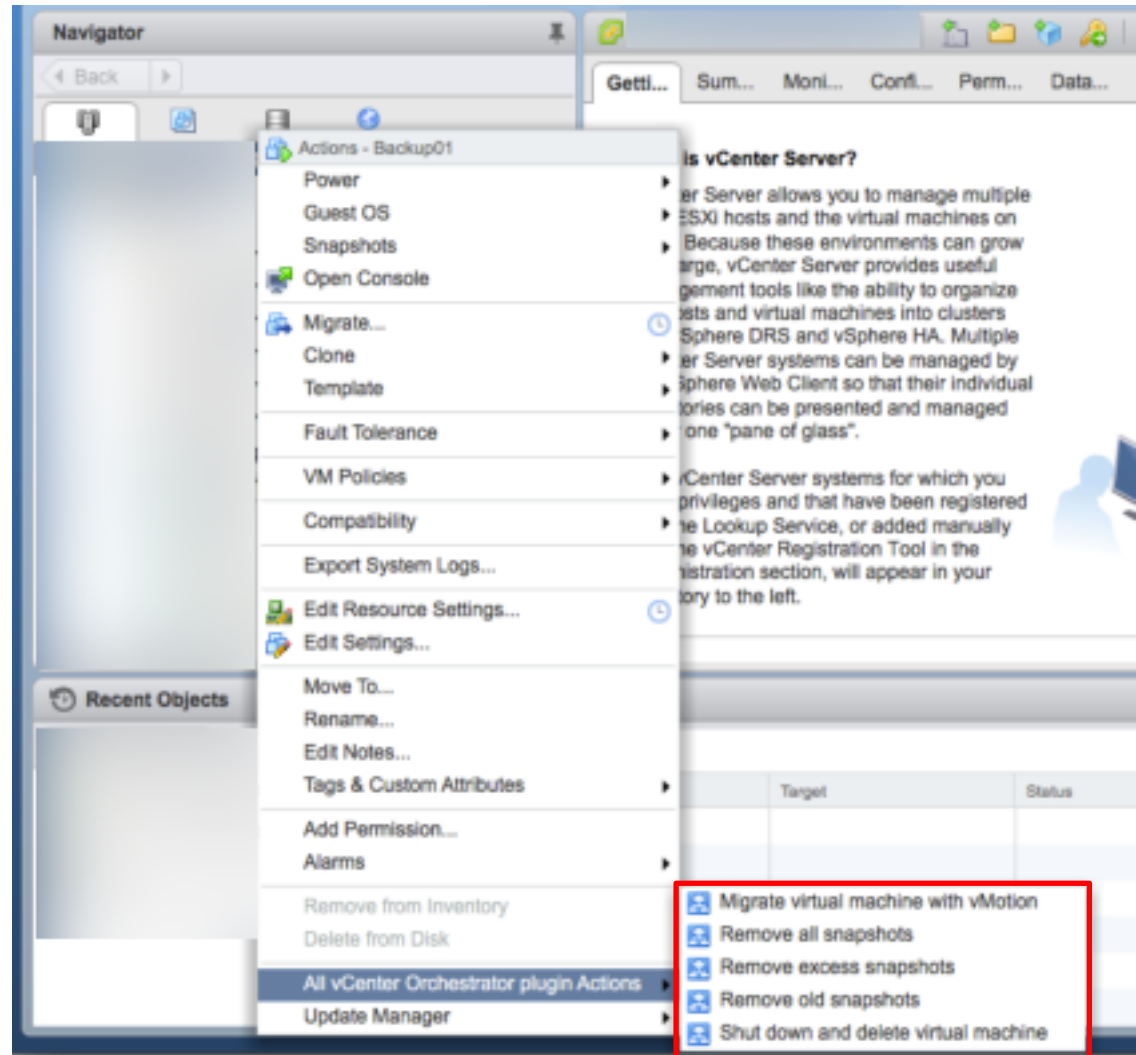


vRO use cases



Integration with...

■ vCenter Server



Integration with...

- vCenter Server
- vRealize Automation

The screenshot displays the 'PQR Experience Center - Self Service Portal' interface. The top navigation bar includes links for Home, Catalog, Items, Requests, Inbox, Design, Administration (selected), Infrastructure, and Containers. A sidebar on the left shows 'Administration' (selected), 'Event Logs', and 'Subscriptions'. The main content area is titled 'Edit Workflow Subscription' and features four tabs: 'Event Topic', 'Conditions', 'Workflow' (selected), and 'Details'. Under the 'Workflow' tab, there is a 'Select a workflow:' section with a tree view showing 'Orchestrator' expanded, containing 'Development', 'Library', 'PQR Automation Package', and 'System'. To the right, the 'Selected workflow:' section shows details for 'buildingSetVmFolder', including its description and input/output parameters. The input parameters table lists 'eventPayloadProperties' as a 'Properties' type. The output parameters section indicates 'No parameters'. At the bottom of the main area are buttons for '< Back', 'Next >', 'Finish', and 'Cancel'. The footer contains the copyright notice '© PQR B.V.', the version 'version 7.2.0 (build 4659752)', and links for 'Privacy Policy' and 'Contact us'.

PQR Experience Center - Self Service Portal

Welcome, [User] | Preferences | Help | Logout

Home Catalog Items Requests Inbox Design Administration Infrastructure Containers

Administration

Event Logs

Subscriptions

Edit Workflow Subscription

Event Topic Conditions Workflow Details

Select a workflow:

- Orchestrator
 - Development
 - Library
 - PQR Automation Package
 - System

Selected workflow:

Name: buildingSetVmFolder

Description:

Input parameters:

Name	Type
eventPayloadProperties	Properties

Displaying 1 - 1 of 1

Output parameters:

Name	Type
No parameters	

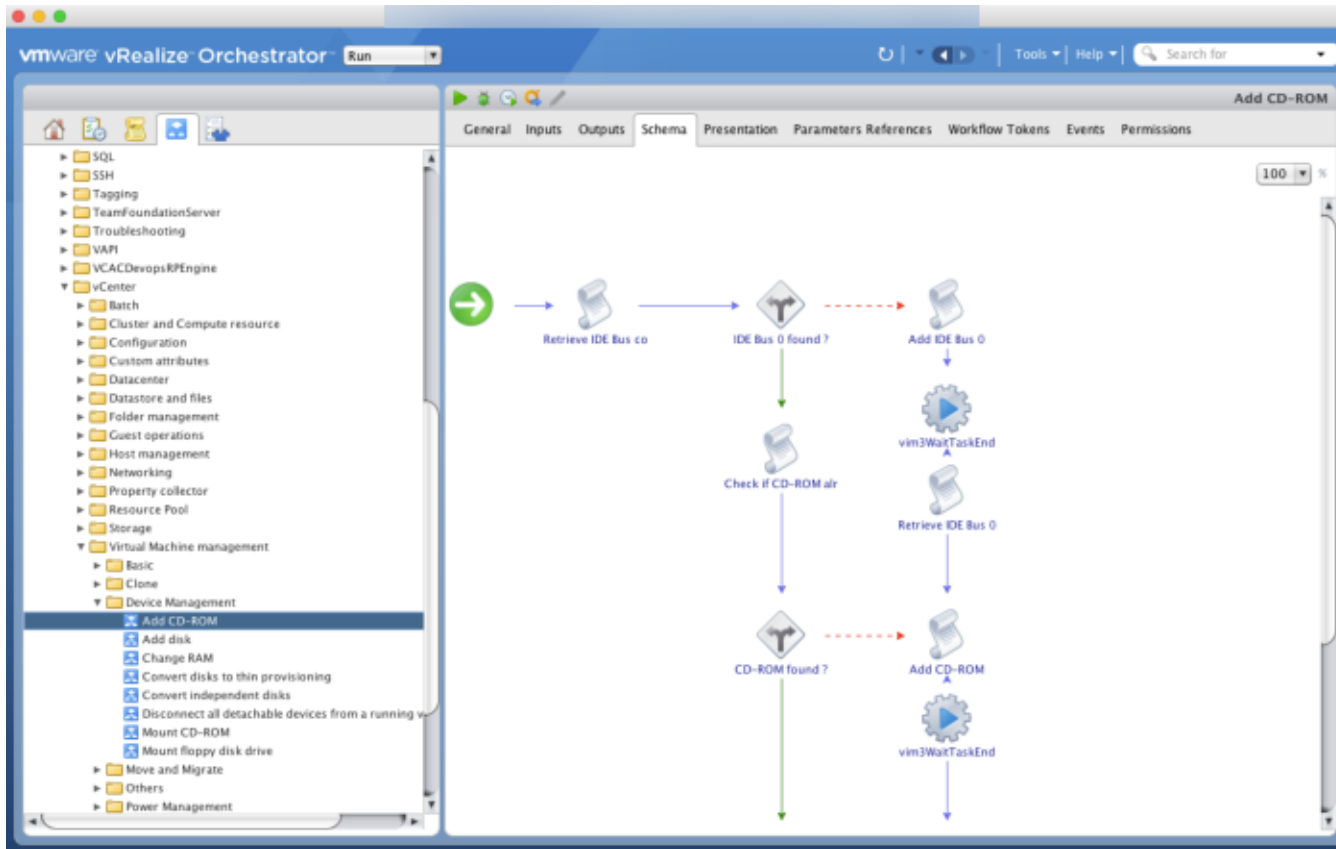
< Back Next > Finish Cancel

© PQR B.V. version 7.2.0 (build 4659752) Privacy Policy Contact us

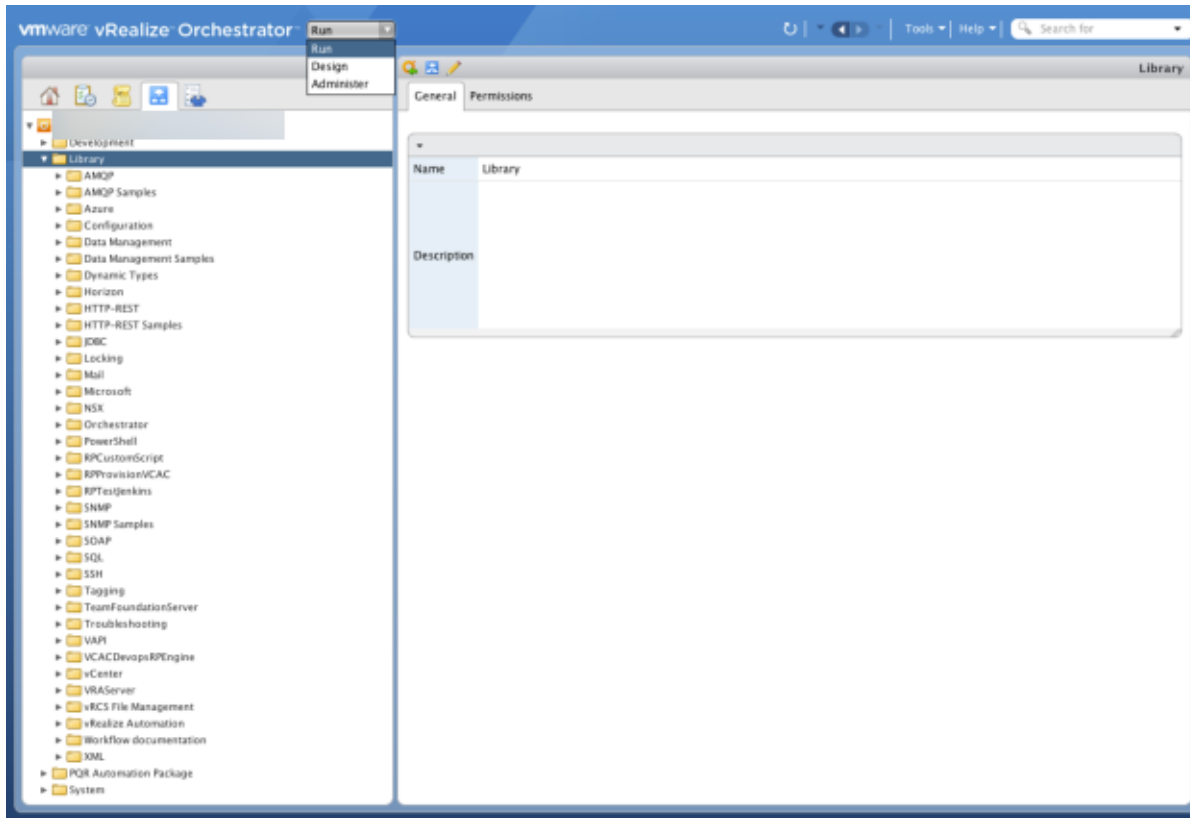
Workflow Development Basics

vRO Client

- Standalone Java client
- Execute and monitor workflows
- Create and develop workflows/actions/etc.



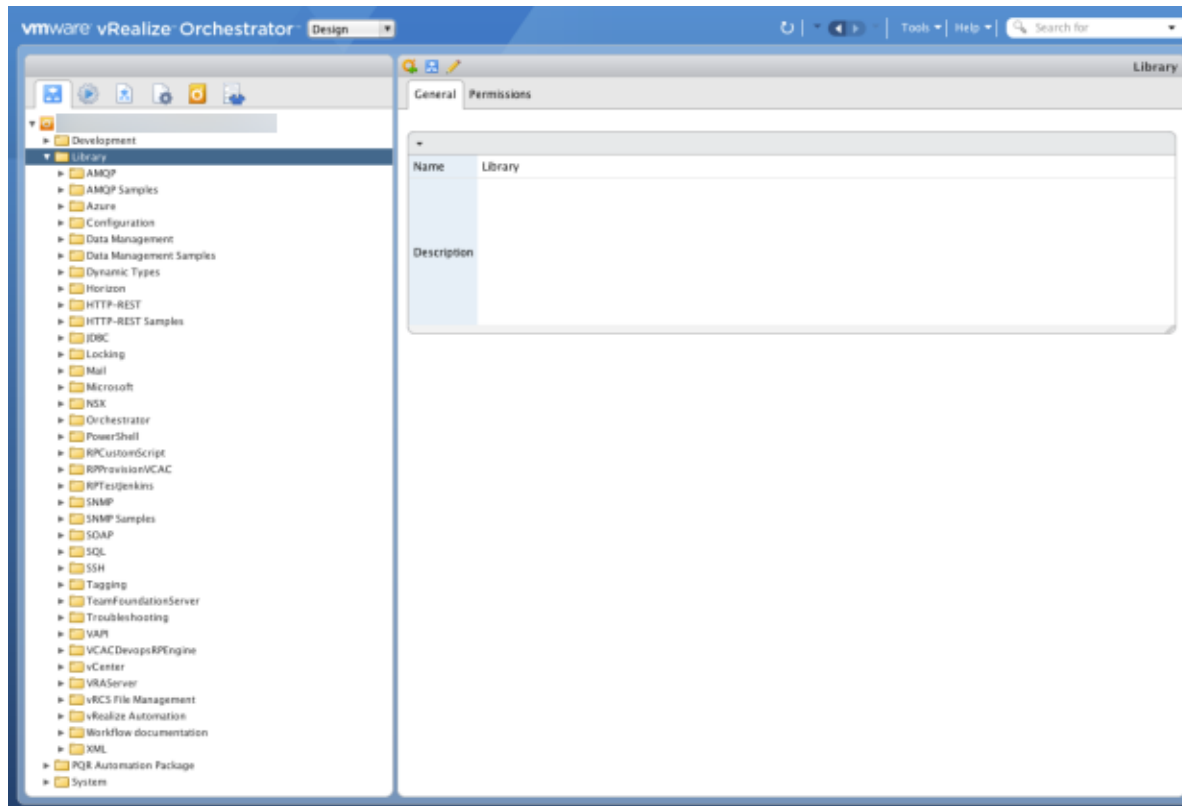
Exploring the Graphical Integrated Development Environment



■ Run

- vRO Overview
- Schedules
- Policies
- Workflows
- Plugins

Exploring the Graphical Integrated Development Environment



- **Design**
 - Workflows
 - Actions
 - Resources
 - Configurations
 - Packages
 - Plugins

Workflow structure

- Defines how to perform an automated task
- Include ready to run workflows/actions
- You can build your own workflows, or reuse existing workflows

My first workflow - Schema

The screenshot displays the 'My first workflow' Schema editor. The interface includes a top navigation bar with tabs: General, Inputs, Outputs, Schema (selected), Presentation, Parameters, References, Workflow Tokens, Events, and Permissions. Below the tabs is a search bar labeled 'Type your filter text here...'. A toolbar contains buttons for Run, Debug, and Validate, along with a progress indicator set to 100%. On the left, a task palette lists various workflow activities under the 'Generic' category, including Scriptable task, Decision, Custom decision, Decision activity, User interaction, Waiting timer, Waiting event, End workflow, Throw exception, Workflow note, Action element, Workflow elem..., Foreach element, and Asynchronous ... The main workspace shows a workflow diagram with a green arrow icon pointing to a 'Scriptable task' icon, which then points to an end workflow icon. The bottom of the window features buttons for Cancel, Revert, Save, and Save and close.

My first workflow - Elements

Workflow elements

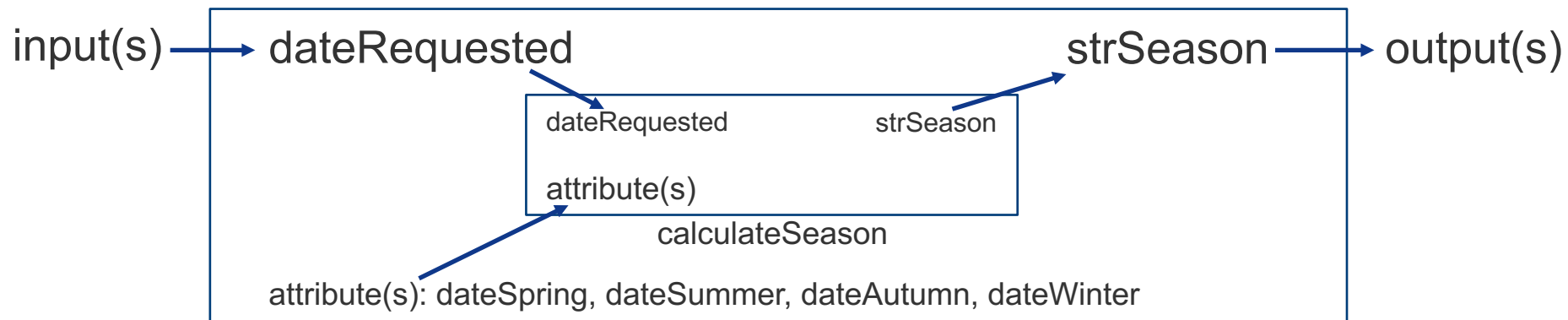
- **Start/end**
- **Scriptable task – Execute a piece of (JavaScript) code**
- **(Nested) Workflow – Run a nested workflow**
- **Actions – JavaScript function with inputs and outputs**
- **Decisions – if (x==y) {do this} else {do that}**
- **Sleep – Just wait**
- **Counter – $n = n + 1$ ($n++$)**
- **Notes - Comments**

My first workflow – Input, output, attributes

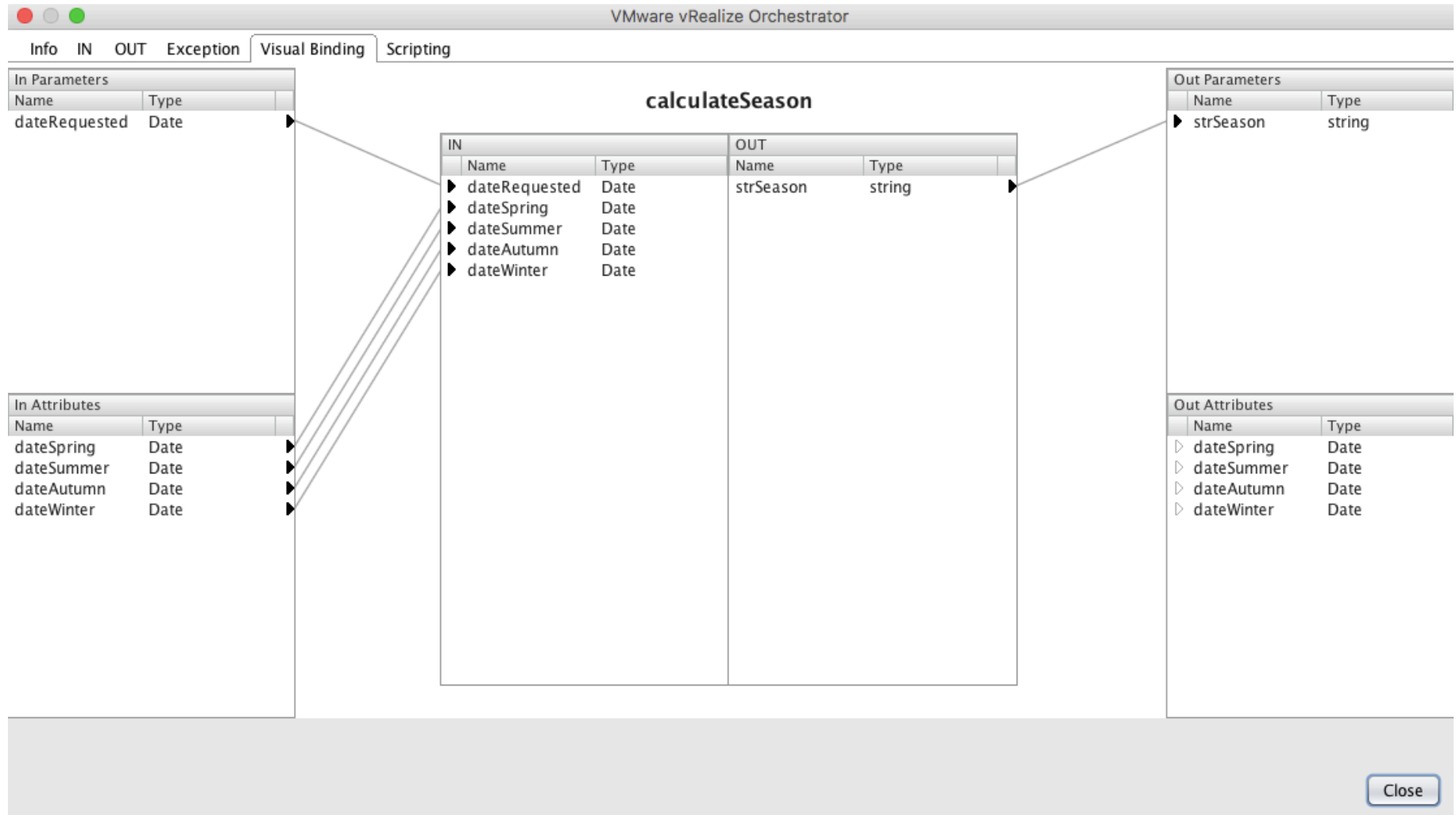
Input, output and attribute

- Input – input variable for a WF/element;
- Output – output variable for a WF/element;
- Attribute – variable for a WF/element.

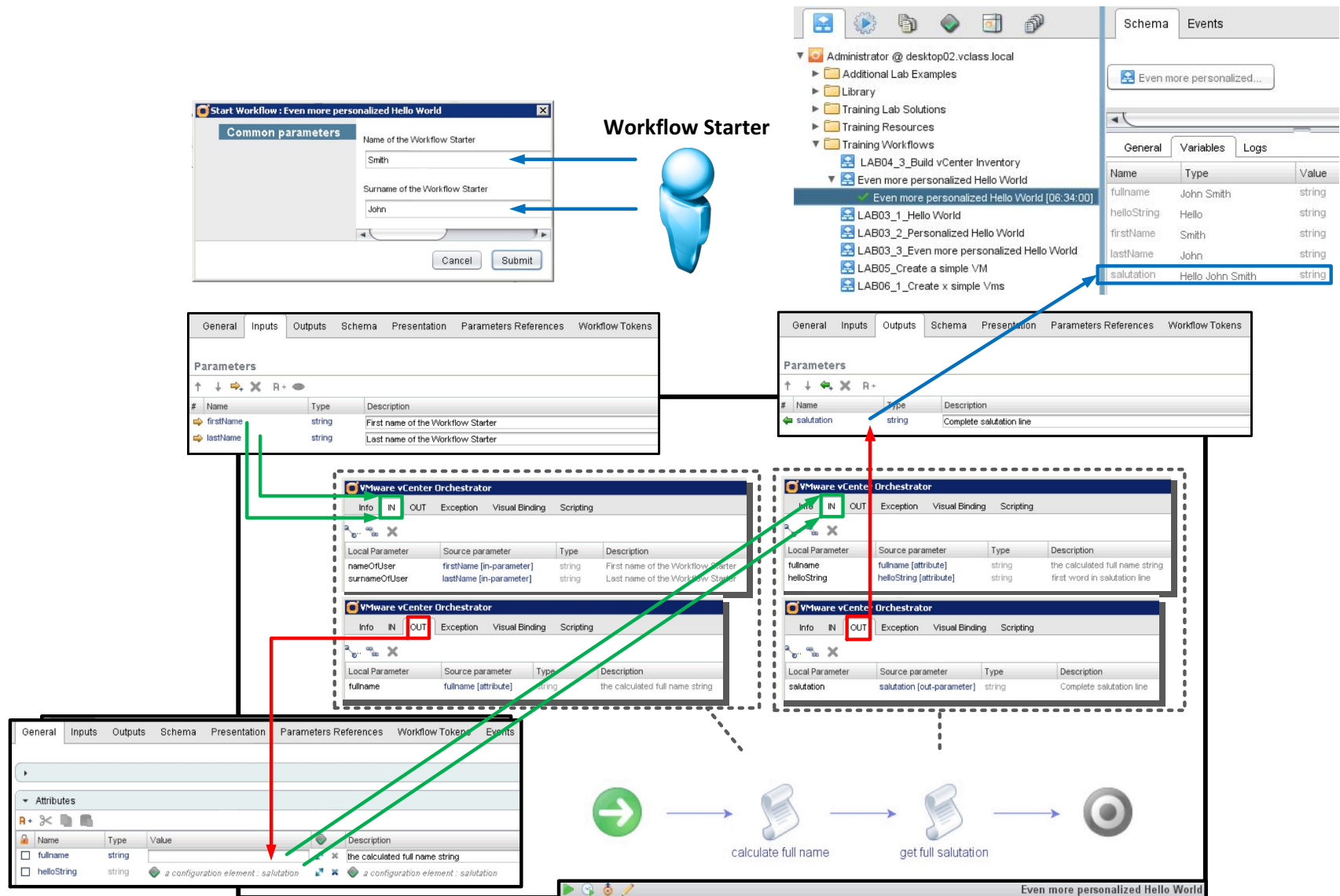
All these variables are "local" for a WF/element



My first workflow – Input, output, attributes



My first workflow – Information flow



My first workflow – Data types (I/II)

- string – For text;
- number – For numbers;
- boolean – True or false;
- Array of string, number, boolean

- Objects:
 - Properties (key/value)
 - VC:VirtualMachine (VM)
 - VC:ClusterComputeResource (Cluster)
 - AD:User (Active Directory user)
 - NSX:VirtualWire (NSX VXLAN network)
 - Azure:VirtualMachine (Azure VM)

- Composite type (define yourself)

My first workflow – Data types (II/II)

The screenshot shows the class hierarchy for `VcVirtualMachine` in the VMware vRealize Orchestrator. The left pane lists the properties of the class, and the right pane lists the methods. A blue arrow points from the word "Properties" to the left pane, and another blue arrow points from the word "Methods" to the right pane.

Properties:

- alarmActionsEnabled
- availableField
- capability
- config
- configIssue
- configStatus
- customValue
- datastore
- declaredAlarmState
- disabledMethod
- effectiveRole
- environmentBrowser
- guest
- guestHeartbeatStatus
- id
- layout
- layoutEx
- name
- network
- overallStatus
- parent
- parentVApp
- permission
- recentTask
- reference
- resourceConfig
- resourcePool
- rootSnapshot
- runtime
- sdkConnection
- snapshot
- storage
- summary
- tag
- triggeredAlarmState
- value
- vimHost
- vimType
- acquireMksTicket() : VcVirtualMachineMksTicket
- acquireTicket(string) : VcVirtualMachineTicket
- answerVM(string,string) : void
- applyEvcModeVM_Task(VcHostFeatureMask[]) : VcTask
- attachDisk_Task(VcID,VcDatastore,number,number) : VcTask
- checkCustomizationSpec(VcCustomizationSpec) : void
- cloneVM_Task(VcFolder,string,VcVirtualMachineCloneSpec) : VcTask
- consolidateVMDisks_Task() : VcTask
- createScreenshot_Task() : VcTask

Methods:

- acquireMksTicket
- acquireTicket
- answerVM
- applyEvcModeVM_Task
- attachDisk_Task
- checkCustomizationSpec
- cloneVM_Task
- consolidateVMDisks_Task
- createScreenshot_Task
- createSecondaryVME_Task
- createSecondaryVM_Task
- createSnapshotEx_Task

Properties

The screenshot shows the VMware vRealize Orchestrator Scripting tab. The script is a `System.log` task that logs the `vm` object. The script is as follows:

```
System.log (vm, "VcVirtualMachine")
```

The script is highlighted in yellow. The left pane shows the `VcVirtualMachine` object type, and the right pane shows the `System.log` task. A blue arrow points from the word "Methods" to the right pane.

Object : VcVirtualMachine

Description

VirtualMachine is the managed object type for manipulating virtual machines, including templates that can be deployed (repeatedly) as new virtual machines. This type provides methods for configuring and controlling a virtual machine.

VirtualMachine extends the ManagedEntity type because virtual machines are part of a virtual infrastructure inventory. The parent of a virtual machine must be a folder, and a virtual machine has no children.

missing name after . operator

Methods

Lab: My first workflow

- Lab: HOL-1721-SDC-5 - Introduction to vRealize Orchestrator
- Module 2: Creating your first workflow (15-20 minutes)
- <http://labs.hol.vmware.com/HOL/catalogs/>

vmware
HOL Online

Welcome, Viktor van den Berg

Lab: HOL-1721-SDC-5 - Introduction to vRealize Orchestrator

TIME REMAINING: 01:27:59

SEND TEXT CTRL+ALT+DEL NOTES POWER DOWN POWER RESET

RECYCLE BIN README.txt

Google Chrome

Mozilla Firefox

VMware vSphere

VMware vSphere

CloudClient

vRealize Orchestrator

VMWARE HANDS-ON LABS

HOL-1721
Host User Main Console Administrator
IP Address 192.168.110.10
Lab Status Ready 05/08 06:14

vmware

6:39 AM

TABLE OF CONTENTS
MORE OPTIONS

1 2 3 4 5 6 336

Lab Module List:

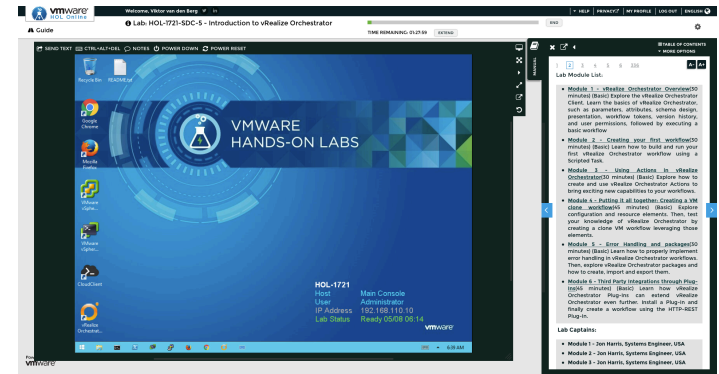
- Module 1 - vRealize Orchestrator Overview(30 minutes) (Basic) Explore the vRealize Orchestrator Client. Learn the basics of vRealize Orchestrator, such as parameters, attributes, schema design, presentation, workflow tokens, version history, and user permissions, followed by executing a basic workflow
- Module 2 - Creating your first workflow(30 minutes) (Basic) Learn how to build and run your first vRealize Orchestrator workflow using a Scripted Task.
- Module 3 - Using Actions in vRealize Orchestrator(30 minutes) (Basic) Explore how to create and use vRealize Orchestrator Actions to bring exciting new capabilities to your workflows.
- Module 4 - Putting it all together: Creating a VM clone workflow(45 minutes) (Basic) Explore configuration and resource elements. Then, test your knowledge of vRealize Orchestrator by creating a clone VM workflow leveraging those elements.
- Module 5 - Error Handling and packages(30 minutes) (Basic) Learn how to properly implement error handling in vRealize Orchestrator workflows. Then, explore vRealize Orchestrator packages and how to create, import and export them.
- Module 6 - Third Party Integrations through Plug-Ins(45 minutes) (Basic) Learn how vRealize Orchestrator Plug-Ins can extend vRealize Orchestrator even further. Install a Plug-In and finally create a workflow using the HTTP-REST Plug-In.

Lab Captains:

- Module 1 - Jon Harris, Systems Engineer, USA
- Module 2 - Jon Harris, Systems Engineer, USA
- Module 3 - Jon Harris, Systems Engineer, USA

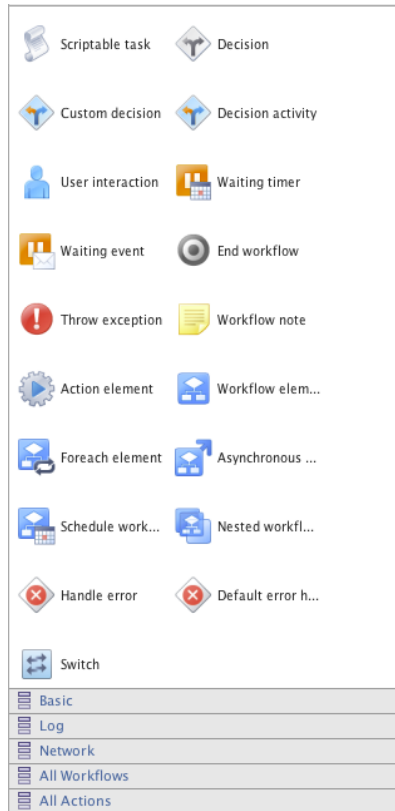
Starting the lab

- Open the VMUG LABS icon
- Open VMware View to connect to the Arrow labs
- User: vmug1-15
- Password: Vmug@2017!
- → Mind the Azerty keyboard 😊
- Open chrome browser
 - <http://labs.hol.vmware.com/HOL/catalogs/>
- Login using your HOL account
- Lab: HOL-1721-SDC-5 - Introduction to vRealize Orchestrator
- Module 2: Creating your first workflow (15-20 minutes)



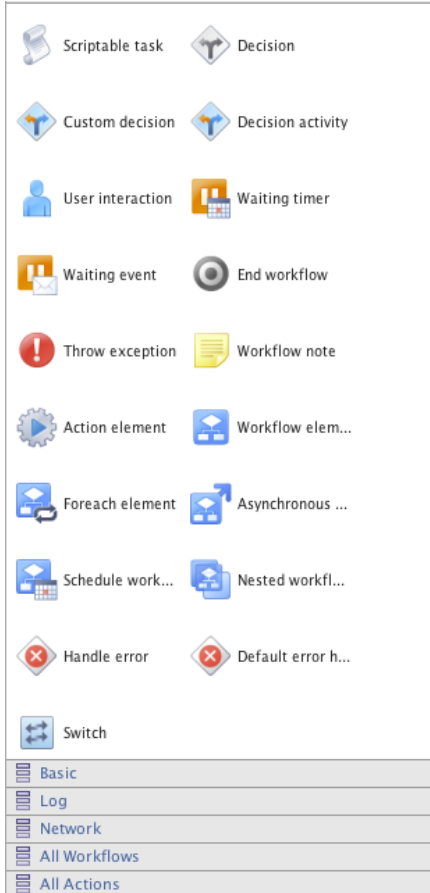
Actions & Configuration Elements

Some guidelines



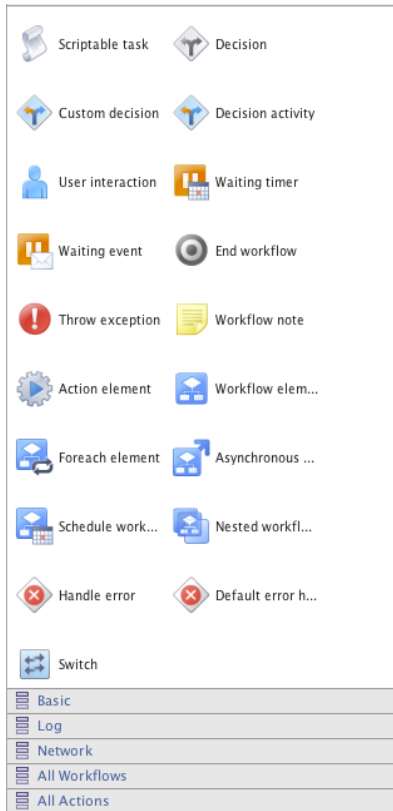
- Workflows can run other workflows.
- In/out parameters are used when calling another workflow.

Some guidelines – Workflow elements



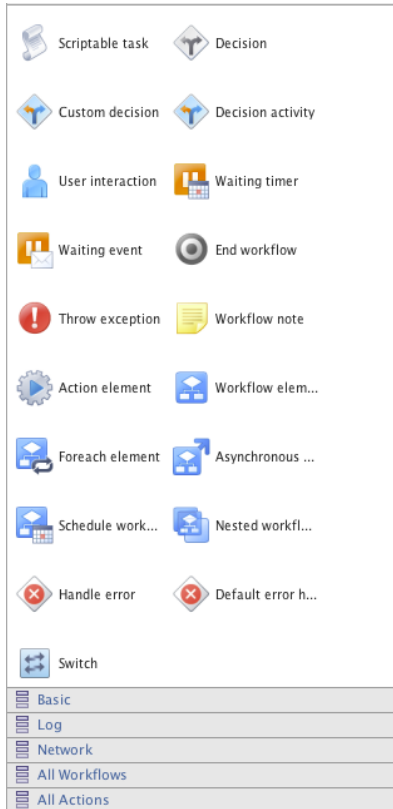
- Action elements are reusable scripts with just one return value.
- Action elements can only contain a script

Some guidelines – Workflow elements



- Decision elements check a bound Boolean parameter directly.
- Custom decisions require a script that returns a Boolean value.
- Decisions have two outgoing connections:
 - Green for true
 - Red for false

Some guidelines – Workflow elements



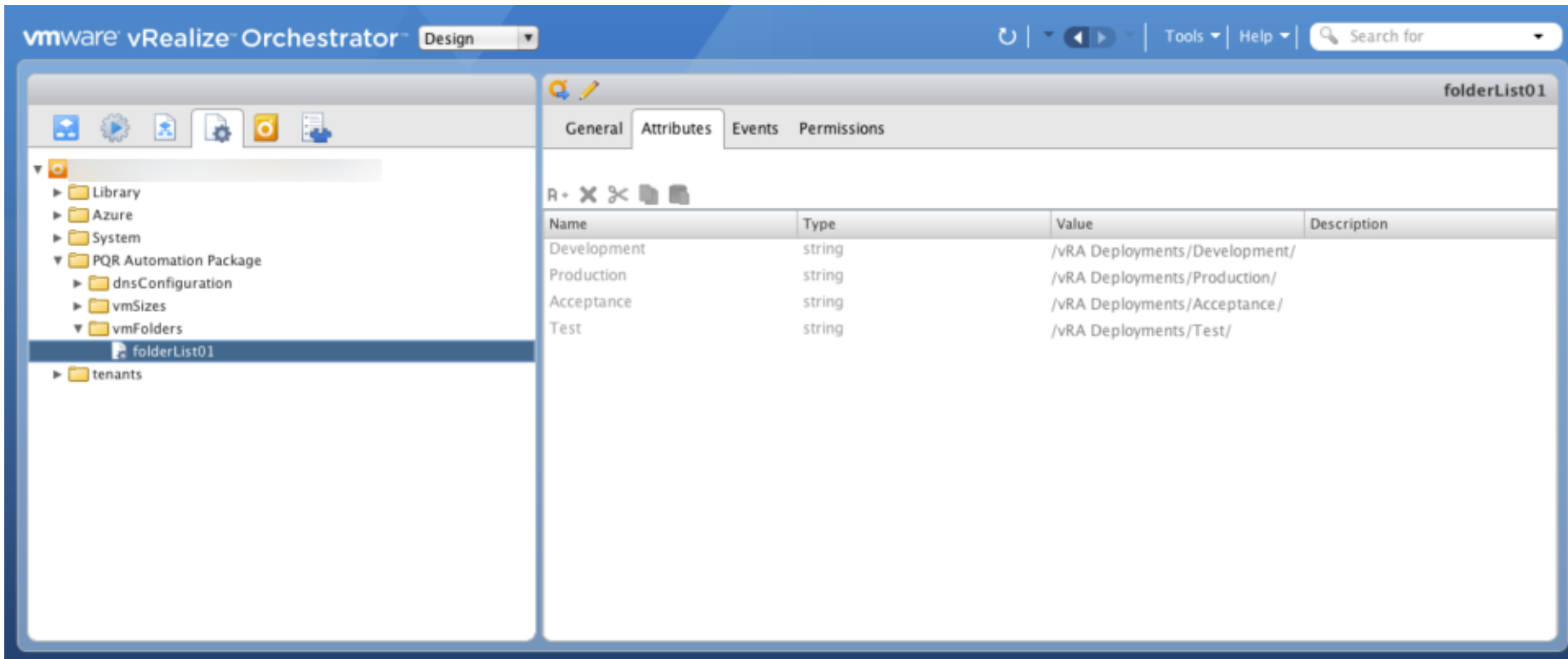
Scriptable tasks:

- Contain JavaScript code that implements a certain functionality
- Can have multiple IN and OUT parameters, available as JavaScript variables in the script.

Give all the elements a descriptive name.

Some guidelines – Configuration elements

- Configuration elements store variables throughout the server.
- A configuration element can have multiple attributes with:
 - [Array] Name Type Value Description



The screenshot shows the VMware vRealize Orchestrator interface in Design view. The left pane displays a tree structure of configuration elements, with 'folderList01' selected under the 'vmFolders' folder. The right pane shows the 'Attributes' tab for 'folderList01', which contains a table of attributes.

Name	Type	Value	Description
Development	string	/vRA Deployments/Development/	
Production	string	/vRA Deployments/Production/	
Acceptance	string	/vRA Deployments/Acceptance/	
Test	string	/vRA Deployments/Test/	

Some guidelines – Configuration elements

- The attributes of a workflow can then be linked to an attribute contained in a configuration element.
- The configuration element can also be used in JavaScript code.

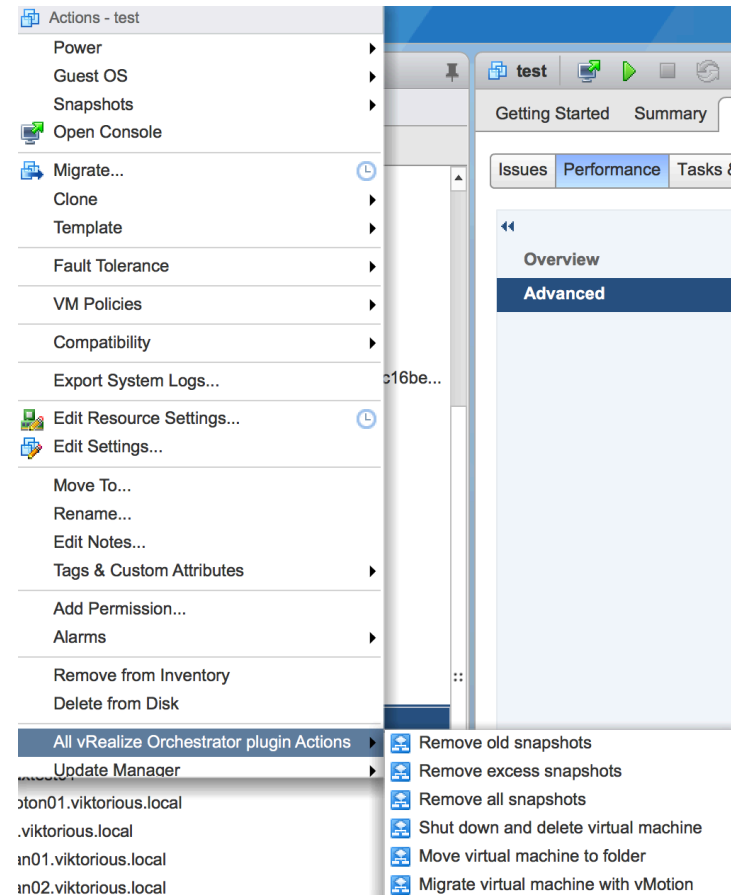
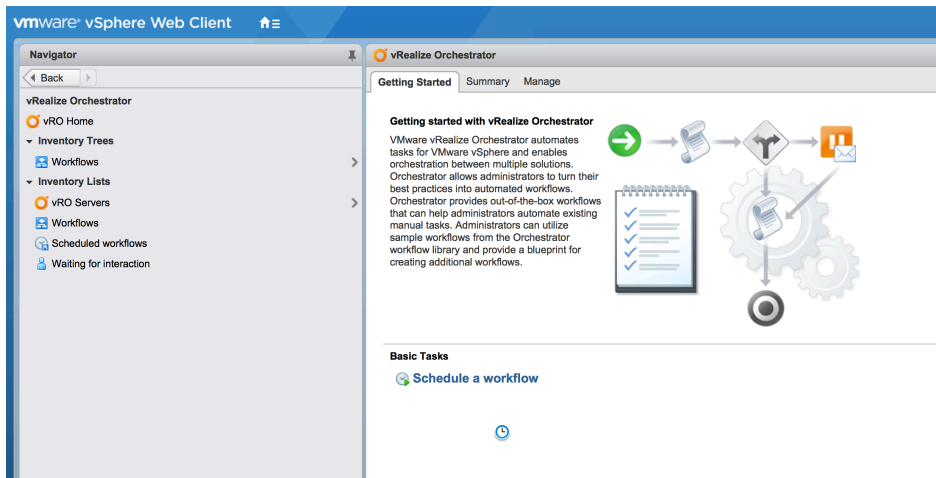
Lab: My first workflow

- Lab: HOL-1721-SDC-5 - Introduction to vRealize Orchestrator
- Module 2: Using actions in vRealize Orchestrator (25 minutes)

The screenshot displays the VMware Hands-On Labs interface. At the top, a navigation bar includes the VMware logo, a welcome message for 'Viktor van den Berg', and links for HELP, PRIVACY, MY PROFILE, LOG OUT, and ENGLISH. Below this, a header bar shows the lab title 'Lab: HOL-1721-SDC-5 - Introduction to vReal...' and a progress bar with a 'TIME REMAINING: 01:16:38' and an 'EXTEND' button. The main area is divided into two panes. The left pane shows a Windows desktop environment with a taskbar containing icons for Recycle Bin, README.txt, Google Chrome, and Internet Explorer. A Start menu is open, displaying a list of applications including On-Screen Keyboard, PuTTY, Remote Desktop Connection, Command Prompt, Windows PowerShell, Internet Explorer, Mozilla Firefox, Google Chrome, Notepad++, and vRealize Orchestrator Client. The right pane shows a manual window titled 'MODULE 3 - USING ACTIONS IN VREALIZE ORCHESTRATOR (30 MINUTES)'. This window contains a table of contents with page numbers 1, 89, 90, 91, 92, 93, and 336. Below the table of contents, it states 'This module contains the following lessons:' and lists three items: 'Introduction', 'Introduction to actions', and 'Create an Action'. The bottom of the interface shows a status bar with the text 'HOL-1721 Host Main Console User Administrator IP Address 192.168.110.10 Lab Status Ready 05/08 06:14' and the VMware logo.

Integration with vCenter Server and vRealize Automation

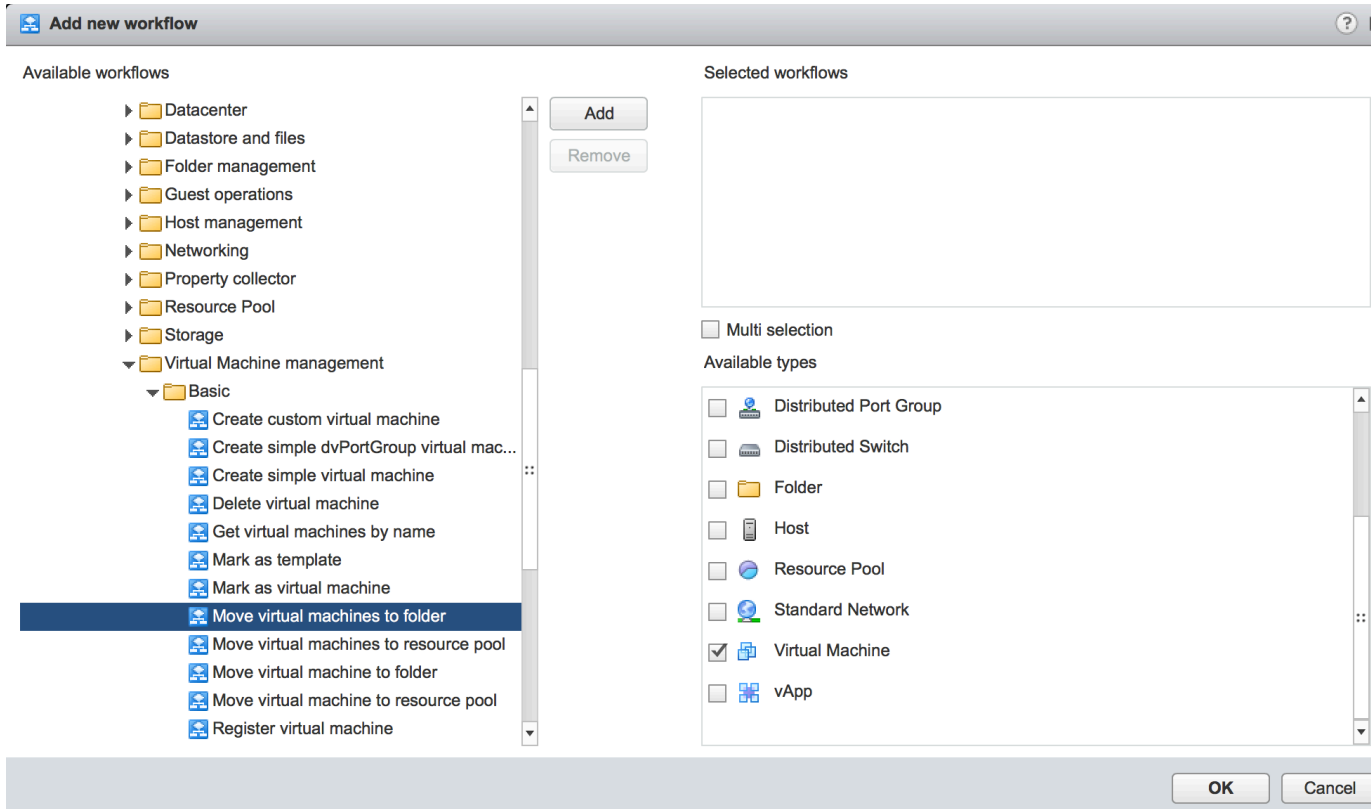
vCenter Integration



vRO + vCenter Server

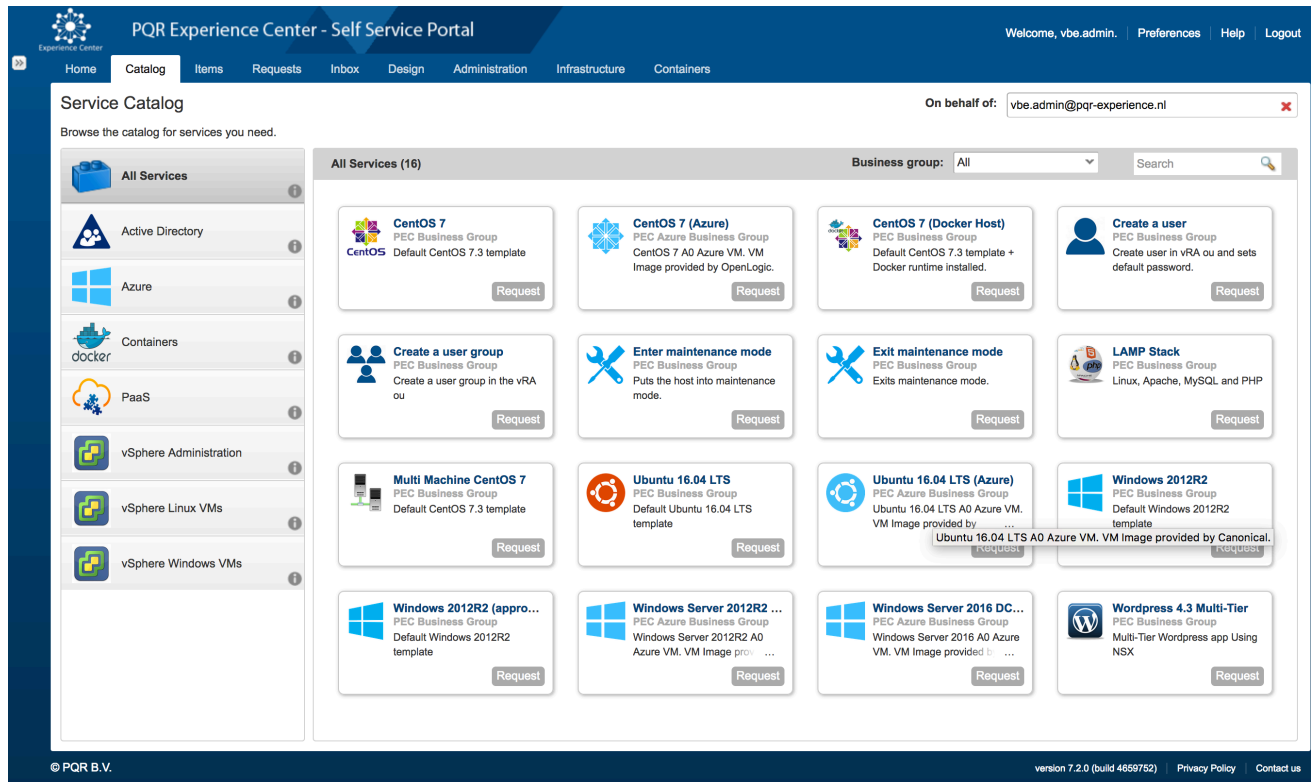
- Seamless integration is provided
- vRO workflows executed on objects

vCenter Integration - Example



- Add a workflow to the WebClient
- Configure the context
- Restart the WebClient

What is vRealize Automation?



vRealize Automation is a tool to implement a self-service catalog that provides IT services:

- Virtual machine LCM;
- Application LCM;
- Anything as a service.

vRealize Automation deployment process

Request Request form Enter options Start request	Deployment Deploy VM First time boot Initial OS configuraiton	Managing VM ready Day 2 operations	Disposing Delete VM Remove objects Update CMDB
--	---	---	--

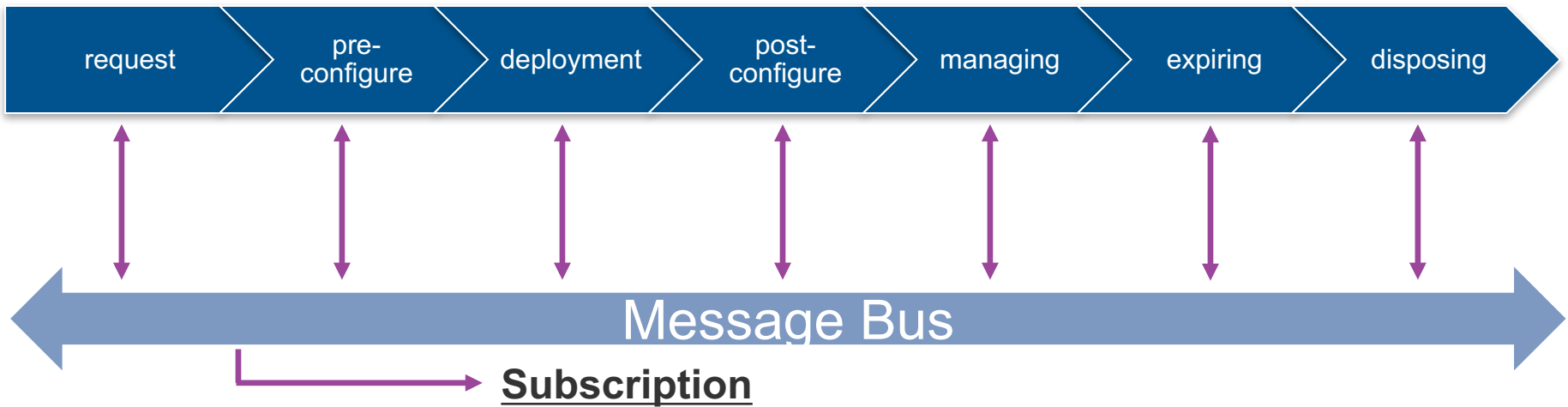


Pre-configure
Preparatory steps
Generate hostname
Create computer account
Request IP address
CMDB registration

Post-configure
Configure VM
Install software
Prepare for use

Expiring
Lease expires
Expire VM
Archive VM

vRealize Automation – Event broker & message bus (I/III)



■ Condition

- Lifecycle state name
- Lifecycle state phase
- ...

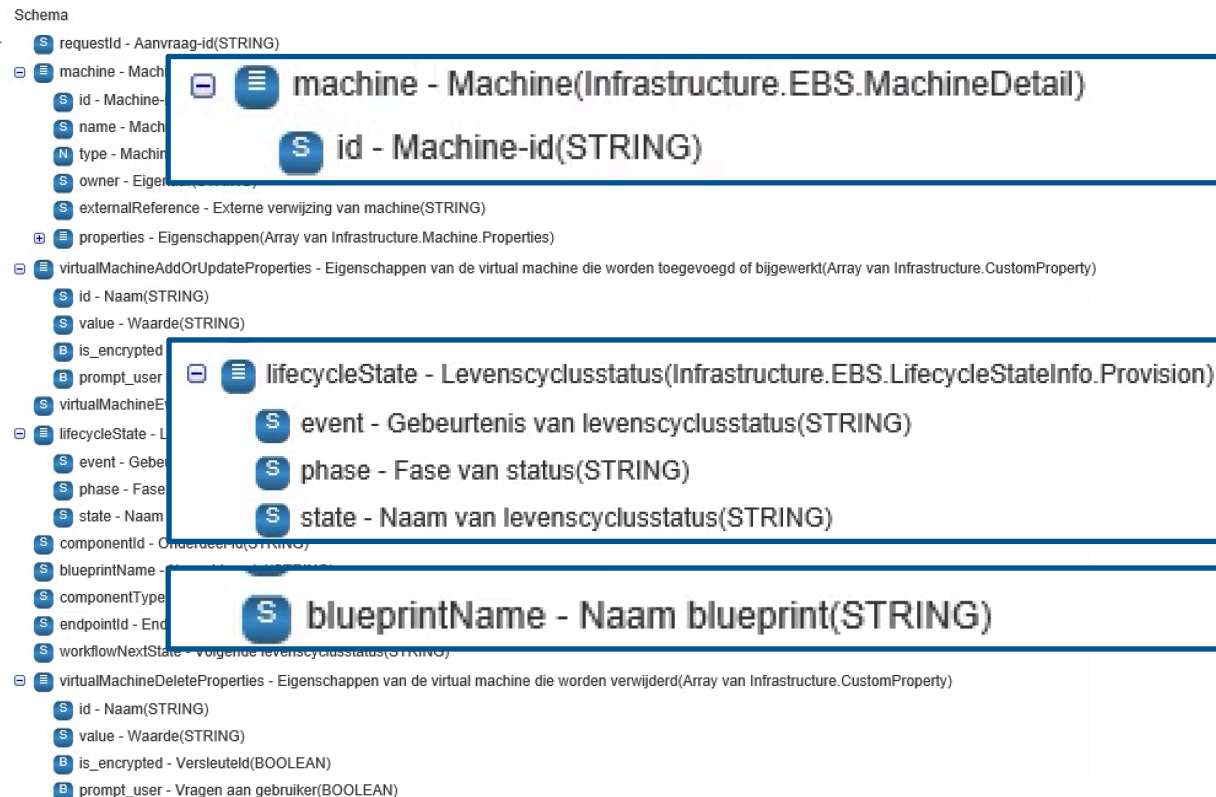
■ Actie: start vRO workflow

■ Blocking/non-blocking

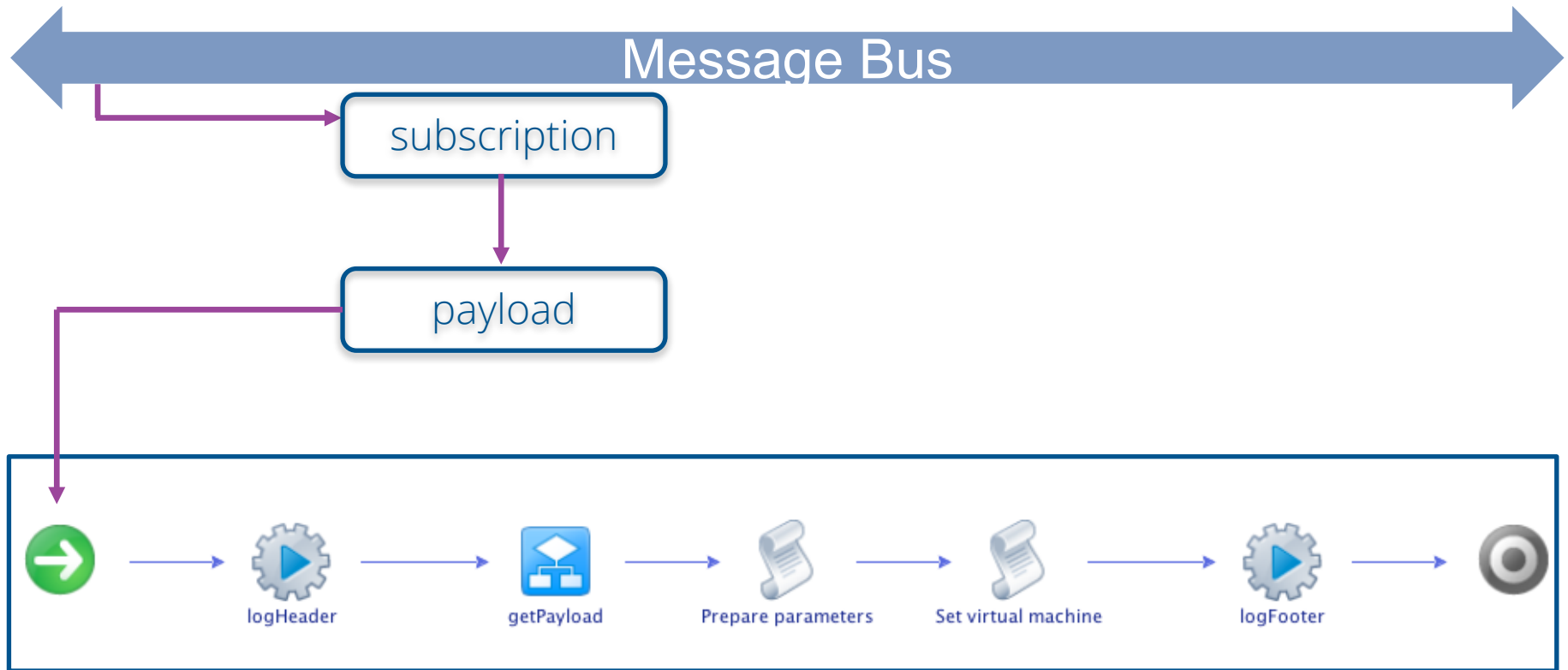
■ Priority

vRealize Automation – Event broker & message bus (II/III)

Message Bus



vRealize Automation – Event broker & message bus (III/III)



Thank you!



VMUG
VMWARE USER GROUP

Your Link to the VMware Community