

A vRO Plugin for Azure

Stephen Mak
Product Value Engineering, Broadcom
Version 0.24, March 6, 2024

Introduction

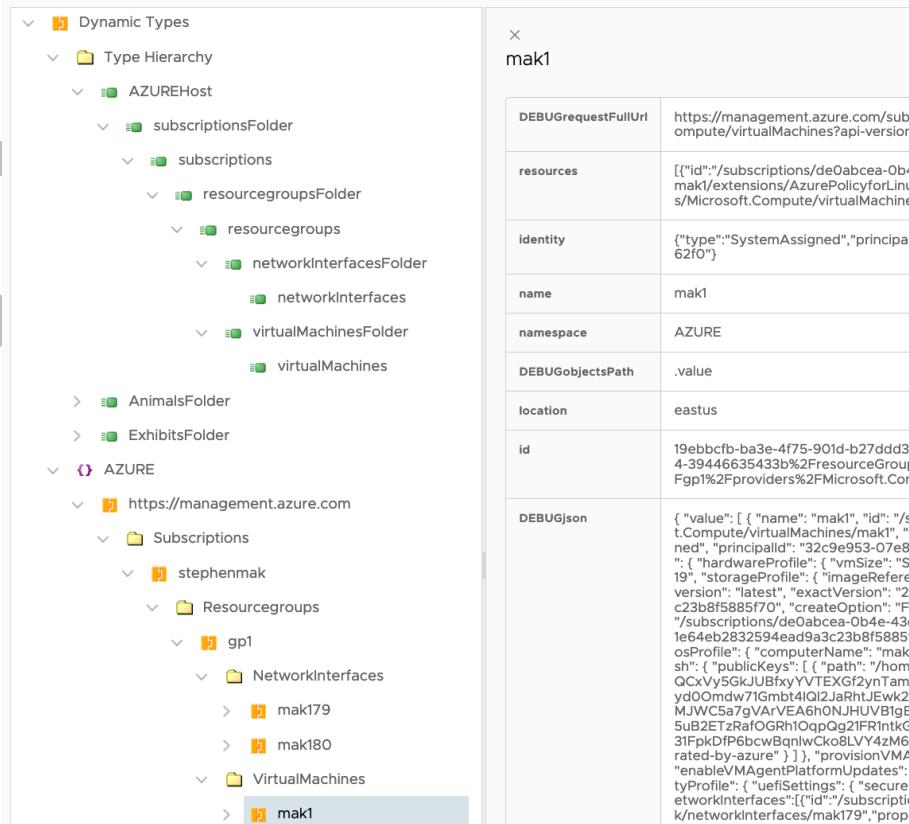
This document lists instructions to create a vRO plugin for Azure. We define vRO Dynamic Types to interface with Azure public cloud for the purpose of consuming native cloud services.

The plugin offers below capabilities.

- 1) A custom REST header action to authenticate with Azure
- 2) Collect subscriptions, resource groups, network interfaces and virtual machines via REST calls and store them in vRO as inventory objects
- 3) A workflow to create a virtual machine using input form actions to look up required resources
- 4) A workflow to delete a virtual machine with the VM object as input
- 5) A workflow to delete a virtual machine with a VM name, a subscription ID and a resource group name as inputs
- 6) Instruction to convert a REST call to a vRO workflow. Our tool creates a vRO workflow with inputs and Azure authentication.

The plugin has a Type Hierarchy below with a sample of inventory objects.

Inventory 



The screenshot shows the vRO Inventory interface. On the left, there's a tree view of 'Dynamic Types' under 'Type Hierarchy'. It includes categories like 'AZUREHost', 'AZURE', and specific resources like 'subscriptions', 'resourcegroups', 'networkInterfaces', 'virtualMachines', and 'Resourcegroups' for a user named 'stephenmak'. On the right, there's a detailed view of a specific inventory object for a virtual machine named 'mak1'. The object has various properties listed:

Property	Value
DEBUGRequestFullPath	https://management.azure.com/subscriptions/de0acea-0b4e-43c3-b2df-5885f70
resources	[{"id": "subscriptions/de0acea-0b4e-43c3-b2df-5885f70/extensions/AzurePolicyForLinux/Microsoft.Compute/virtualMachines/mak1"}]
identity	{"type": "SystemAssigned", "principalId": "62f0"}
name	mak1
namespace	AZURE
DEBUGObjectsPath	.value
location	eastus
id	19ebbcfb-ba3e-4f75-901d-b27ddd4-39446635433b%2FresourceGroups%2Fproviders%2FMicrosoft.CognitiveServices%2Fmachines/mak1
DEBUGJson	{ "value": [{ "name": "mak1", "id": "/subscriptions/de0acea-0b4e-43c3-b2df-5885f70/extensions/AzurePolicyForLinux/Microsoft.Compute/virtualMachines/mak1", "type": "Microsoft.Compute/virtualMachines", "principalId": "32c9e953-07e8-4f1d-8500-164eb2832594ead9a3c23bbf5885f70", "hardwareProfile": { "vmSize": "Standard_D2_v2", "storageProfile": { "imageReference": { "version": "latest", "exactVersion": "2023-09-01", "createOptions": "FromImage" }, "osDisk": { "computerName": "mak1", "publicKeys": [{ "path": "/home/stephenmak/.ssh/id_rsa" }] } }, "provisionVMAgent": true, "enableVMPlatformUpdates": true, "vnetProfile": { "subnetSettings": [{ "id": "/subscriptions/de0acea-0b4e-43c3-b2df-5885f70/resourceGroups/mak1/providers/Microsoft.Network/networkInterfaces/mak1" }] } }] }

The plugin stores pre-defined workflows in the AZURE/Default folder.

Embedded-VRO

Dashboard

Library

Workflows

Actions

Policies

Activity

Workflow Runs

Workflows

AZURE

AZURE CRUD Operations

Default

- Invoke 'CREATE SQL Managed Instance: PUT https:__manage...' (Icon)
- Invoke 'Create VM: PUT https:__management.azure.com_sub...' (Icon)
- Invoke 'DELETE vm2 object' (Icon)
- Invoke 'DELETE vm: DELETE https:__management.azure.com_...' (Icon)
- Test DT actions (Icon)

The plugin stores pre-defined ‘get’ actions in two modules, com.vmware.pve.azure and com.vmware.coe.dynamicTypes.pluginGeneratorV3.

Embedded-VRO

Dashboard

Library

Workflows

Actions

Policies

Activity

Workflow Runs

Scheduled

Waiting for Input (0)

Policy Runs

Actions 4 of 697

Add filter... NEW ACTION

Name	Tags	Description
getAzureCustomHeaders	com.vmware.coe.dynamicTypes.plugin...	Get authorization header for Azures
getAzureSubscriptionIds	com.vmware.pve.azure	
getAzureNetworkInterfaceNames	com.vmware.pve.azure	
getAzureResourceGroupNames	com.vmware.pve.azure	

Dynamic Type Plug-in generator

Reader will use Dynamic Type Plug-in generator v3.1 to create Dynamic Types for the plugin. The generator is consisted of a set of workflows and actions.

Embedded-VRO

The screenshot shows the 'Library' section of the Embedded-VRO interface. The left sidebar has a tree structure with the following items:

- Dashboard
- Library
 - Workflows
 - Actions
 - Policies
- Activity
 - Workflow Runs
 - Scheduled
 - Waiting for Input (0)
 - Policy Runs
- Assets
 - Packages
 - Configurations
 - Resources
 - Environments
- Administration
 - Groups
 - Inventory
 - Audit Logs
 - Git Repositories
 - Git History
- Deleted Items
- API Explorer

The 'Assets' section is currently selected, and its sub-sections are visible on the right. The 'CSE' section is expanded, showing the 'Dynamic Types Plug-in generator v3.1' sub-section, which is also expanded. The 'Helper' sub-section under 'CSE' is highlighted with a light blue background.

CSE

Dynamic Types Plug-in generator v3.1

- Create a plug-in no swagger
 - 3- Repeat for each object type
 - 3a- Add a REST operation
 - 3b- Invoke a REST operation with headers from action
 - 3c- Create a type from a REST operation
 - 4- Remove a REST operation (Optional)
 - 1- Create new plug-in
 - 2a- Create a REST host
 - 2b- Add REST Host as DT Host
- Create a plug-in with Swagger

Helper

- CRUD operations creation
 - Create a type - Helper
 - Create plug-in from swagger string and REST Host
 - Get swagger properties as JSON
 - Update a type findById

Test

- Create DT for vRA Services
- Delete vRA DT services
- Get workflow tokens duration
- Test plug-ins types
- Delete a type
- Delete plug-in
- Export a plug-in as a package
- Set up first - Set vRO credentials

Install

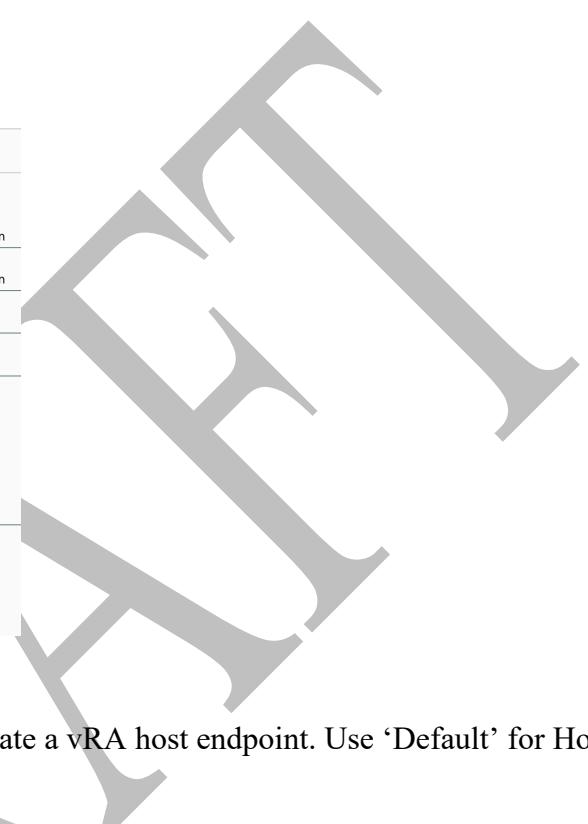
Below are the steps.

Create REST host – https://login.microsoftonline.com

Run below workflow with no authentication to create a REST endpoint for Azure login. Use the input values listed.

Add a REST host

Adds a REST host to the plug-in's inventory.



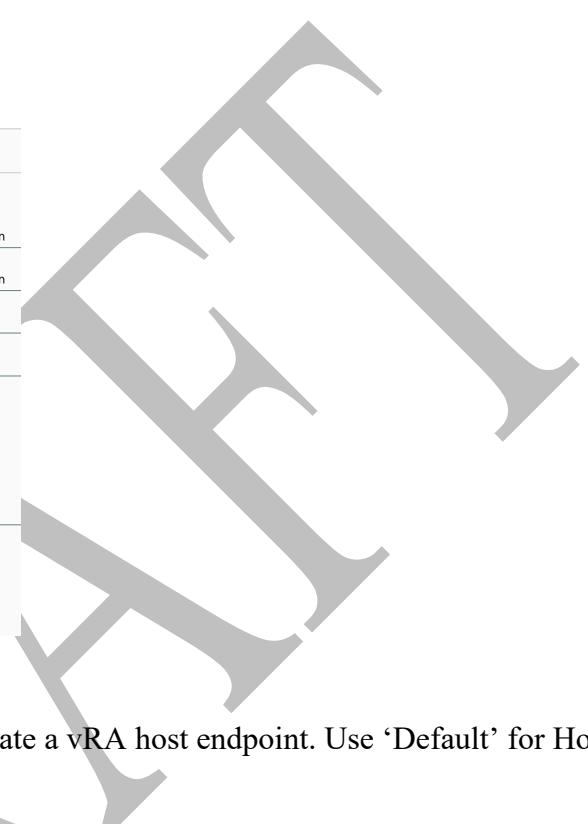
Host Properties		Host Authentication	Proxy Settings	SSL
Properties to create a new host. The name is the host's unique identifier.				
Name *	https://login.microsoftonline.com			
URL *	https://login.microsoftonline.com			
Connection timeout (seconds)	30			
Operation timeout (seconds)	60			
If set to true, the certificate is accepted silently and the certificate is added to the trusted store.	<input checked="" type="checkbox"/>			
Support for parallel request executions	<input checked="" type="checkbox"/>			
Redirect strategy	defaultRedirect			
RUN	CANCEL			

Create vRA host endpoint

Run below workflow with authentication to create a vRA host endpoint. Use 'Default' for Host name.

Add vRA Host

Adds the configuration of the VMware Aria Automation Host in the plugin's inventory



Host properties		User credentials
Connection Type *	vra-onprem	
Host Name *	Default	
Host URL *	https://cava-n-82-056.eng.vmware.com	
Automatically install SSL certificates	<input checked="" type="checkbox"/>	
RUN	CANCEL	

Add vRA Host

Adds the configuration of the VMware Aria Automation Host in the plugin's inventory

Host properties User credentials

Session Mode * Shared Session Per User Session

User Name * fritz@coke.sqa-horizon.local

Password * *****

Inventory

- >  SNMP
- >  PowerShell
- <  VMware Aria Automation
 - <  Default: https://cava-n-82-056.eng.vmware.com

Set Configuration Elements

Update below values for your environment. Hostname is 'Default' matching vRA host object name in inventory.

Orchestrator credentials DELETE FIND USAGES FIND DEPENDENCIES

General Variables Version History Audit

View, edit, and create local variables for your configuration.

NEW DELETE COPY PASTE

<input type="checkbox"/>	Variable	Value	Type	Description
<input type="checkbox"/>	password	*****	SecureString	
<input type="checkbox"/>	username	fritz@coke.sqa-horizon.local	string	
<input type="checkbox"/>	hostname	Default	string	host description

microsoftonline [DELETE](#) [FIND USAGES](#) [FIND DEPENDENCIES](#)

General [Variables](#) Version History Audit

View, edit, and create local variables for your configuration.

NEW	DELETE	COPY	PASTE
<input type="checkbox"/>	Variable	▼	Value
<input type="checkbox"/>	url		*****
<input type="checkbox"/>	azureAuthenticationContent		*****
<input type="checkbox"/>	restHost	[object REST:RESTHost]	
	redirectStrategy	defaultRedirect	
	operationTimeout	60	
	displayName	https://login.microsoftonline.com: https://logi...	
	@type	RESTHost	
	proxyAuthentication	NONE	
	type	RESTHost	
	proxyHost		
	url	https://login.microsoftonline.com	
	proxyPort	0	
	dunesId	becc9baa-dd3f-45c1-8a72- c8f15641d175	
	hostVerification	true	
	name	https://login.microsoftonline.com	
	id	becc9baa-dd3f-45c1-8a72- c8f15641d175	
	@fullType	REST:RESTHost	
	connectionTimeout	30	
	authentication	NONE	

Where url is a string of below format:

/ [TENANT_ID] /oauth2/token

azureAuthenticationContent is a string of below format:

grant_type=client_credentials&client_id=[CLIENT_ID]&client_secret=[SECRET]&re
source=https://management.azure.com

See reference section [Create Application Secret](#) for detail.

Test getAzureCustomHeaders action

Run below action to obtain a bearer token verifying values of microsoftonline configuration element.

getAzureCustomHeaders

RUN DEBUG DELETE FIND USAGES FIND DEPENDENCIES

General Script Version History Audit

Runtime Environment

JavaScript

```

1 System.log("Getting configuration element");
2 for each(var configElement in Server.getConfigurationElementCategoryWithPath("Dynamic Types Plug-in Generator")) {
3     System.log(configElement.name);
4     if (configElement.name == "microsoftonline") {
5         restHost = configElement.getAttributeWithKey("restHost").value;
6         url = configElement.getAttributeWithKey("url").value;
7         content = configElement.getAttributeWithKey("azureAuthenticationContent").value;
8         break;
9     }
10 }
11 if (restHost == null) throw "REST Host, username, password not set in Extensibility Samples/vRAHost configuration";
12
13 var restOp = new RESTOperation("Get access token"); // Define the name of the new operation
14 restOp.method = "POST"; // How is data sent to the api when called. POST mean we send data to the api.
15 restOp.urlTemplate = url; // Define how to call api. Three parameters get sent when we call the api.
16 restOp.defaultContentType = "application/x-www-form-urlencoded";
17 var inParamtersValues = []; // Add input parameters and variables to array
18 operation = restHost.addOperation(restOp); // Add defined operation to rest host defined in "restHost" attribute
19 var request = operation.createRequest(inParamtersValues, content); // Create request to api
20
21 var response = request.execute(); // Execute request and store returndata in "response"
22 // Code below is for debugging purposes and will output data in Orchestrator client when run. Output is not part of the actual workflow
23 System.log("Returned status code: " + response.statusCode);
24 var contentAsString = response.contentAsString; // Convert response to plain string
25 //System.debug("Content as string: " + contentAsString);
26
27 var jsonResponse = JSON.parse(contentAsString);
28 tokenId = jsonResponse.access_token; // Extract the token variable form the response
29 //System.log ("access token id = " + tokenId);
30 var headersProp = new Properties();
31 headersProp.put("Authorization", "Bearer " + tokenId);
32 //headersProp.put("Authorization", "Bearer " + token);
33 return headersProp;

```

API Explorer

Filter objects

- System
 - Objects
 - Enums
 - Types
- Plugins
 - AD

Properties

Return type Properties

Inputs

ADD NEW INPUT

getAzureCustomHeaders

RUN DEBUG DELETE FIND USAGES FIND DEPENDENCIES Completed

General Script Version History Audit

Name getAzureCustomHeaders

Module com.vmware.coe.dynamicTypes.pluginGenerator

Result / Inputs Logs

Name	Value	Type
Action Result	[Properties]	Properties
Authorization	Bearer eyJ0eXAiOiJKV1QiLCJhbGciO... iJSUzI1NilsIn1g1dCl6lhSdmvO... FA3QTNVYVdTbIU3Yk05blQ... wTWpoQSlsImtpZC16lhSdmv... OFA3QTNVYVdTbIU3Yk05bl...	string

Create namespace – AZURE

Run workflow (1) in the CSE/Dynamic Types Plug-in generator v3 folder to create a namespace.
Use the input values below.

-1- Create new plug-in



Plug-in Name *	AZURE
Host icon *	default-16x16.png
Action module to create	com.vmware.cse.dt.azure
Workflow category	AZURE
Action to get custom headers	getAzureCustomHeaders

RUN **CANCEL**

Create REST host - <https://management.azure.com>

Run workflow (2a) with no authentication to add the AZURE management host.

-2a- Create a REST host

Adds a REST host to the plug-in's inventory.

Host Properties	Host Authentication	Proxy Settings	SSL
Properties to create a new host. The name is the host's unique identifier.			
Name *	https://management.azure.com		
URL *	https://management.azure.com		
Connection timeout (seconds)	30		
Operation timeout (seconds)	60		
If set to true, the certificate is accepted silently and the certificate is added to the trusted store.	<input checked="" type="checkbox"/>		
Automatically URL Redirection	<input checked="" type="checkbox"/>		
Support for parallel request executions	<input checked="" type="checkbox"/>		

Create DT Host from REST Host – AZUREHost

Run workflow (2b) to create a Dynamic Type (DT) host. Use the input values below.

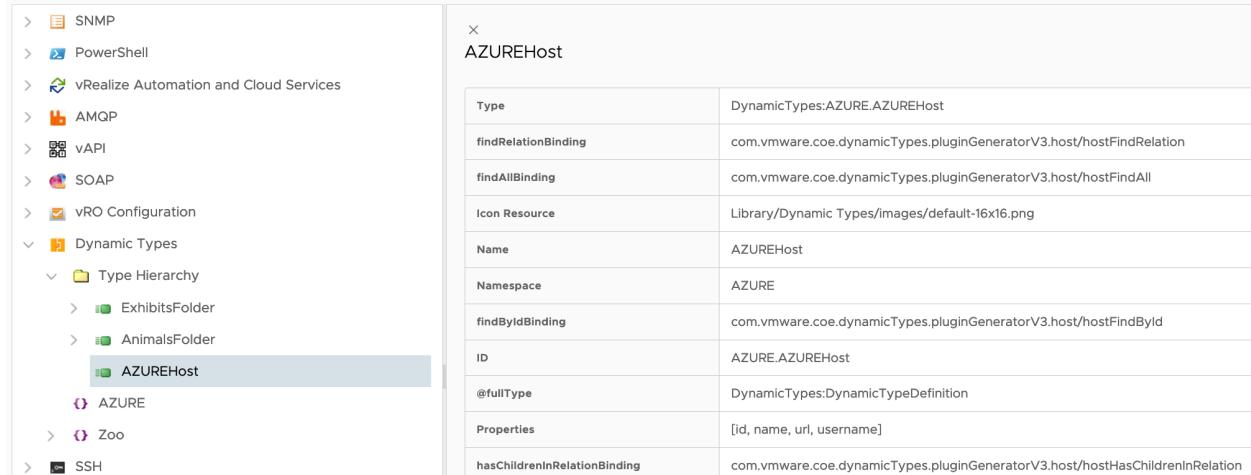
-2b- Add REST Host as DT Host

restHost *	https://management.azure.com/: https://manageme...
namespace *	AZURE
authenticationConfigurationElement	Q

RUN **CANCEL**

Examine AZUREHost in Inventory

Inventory 



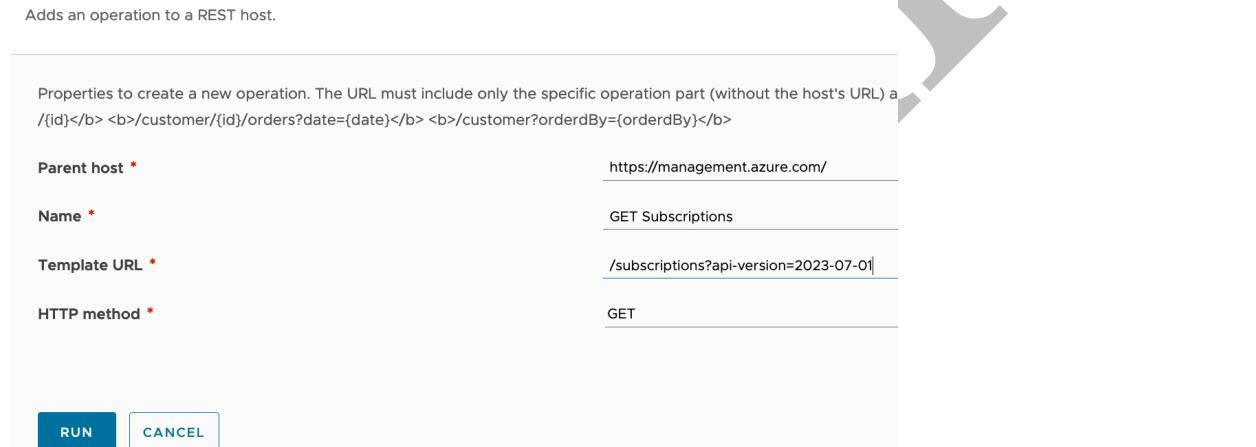
The screenshot shows the vRealize Automation and Cloud Services inventory interface. On the left, there is a tree view of various connection types and a 'Dynamic Types' section. Under 'Dynamic Types', the 'Type Hierarchy' is expanded, showing 'ExhibitsFolder' and 'AnimalsFolder', with 'AZUREHost' selected and highlighted in grey. To the right of this tree view is a detailed properties panel for 'AZUREHost'. The properties listed are:

Type	DynamicTypes:AZURE.AZUREHost
findRelationBinding	com.vmware.coe.dynamicTypes.pluginGeneratorV3.host/hostFindRelation
findAllBinding	com.vmware.coe.dynamicTypes.pluginGeneratorV3.host/hostFindAll
Icon Resource	Library/Dynamic Types/images/default-16x16.png
Name	AZUREHost
Namespace	AZURE
findByIdBinding	com.vmware.coe.dynamicTypes.pluginGeneratorV3.host/hostFindById
ID	AZURE.AZUREHost
@fullType	DynamicTypes:DynamicTypeDefinition
Properties	[id, name, url, username]
hasChildrenInRelationBinding	com.vmware.coe.dynamicTypes.pluginGeneratorV3.host/hostHasChildrenInRelation

GET Subscriptions

Run workflow (3a) to add the REST operation. Use the input values below.

-3a- Add a REST operation



Adds an operation to a REST host.

Properties to create a new operation. The URL must include only the specific operation part (without the host's URL) a /{id} /customer/{id}/orders?date={date} /customer?orderdBy={orderdBy}

Parent host *	<input type="text" value="https://management.azure.com/"/>
Name *	<input type="text" value="GET Subscriptions"/>
Template URL *	<input type="text" value="/subscriptions?api-version=2023-07-01 "/>
HTTP method *	<input type="text" value="GET"/>

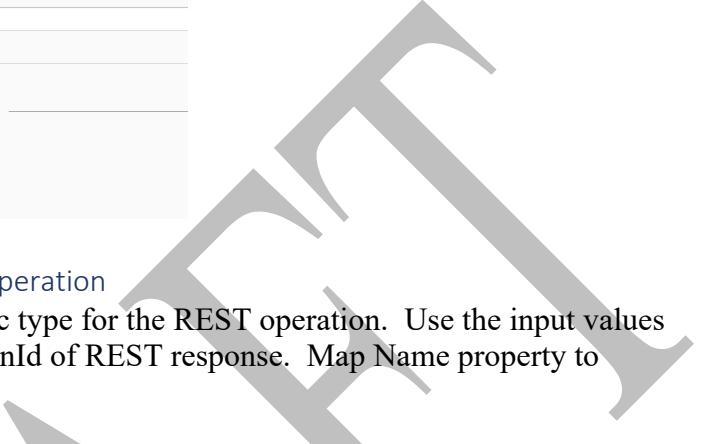
RUN **CANCEL**

Template URL: /subscriptions?api-version=2023-07-01

Test REST Operation

Run workflow (3b) to test the operation.

-3b- Invoke a REST operation with headers from action



REST Operation *

getCustomHeadersAction

inParametersValues

+

content

GET Subscriptions

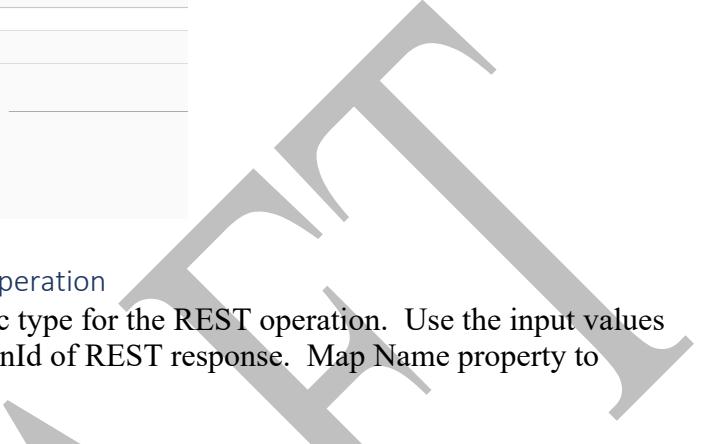
getAzureCustomHeaders

RUN CANCEL

Create subscriptions type from REST operation

Run workflow (3c) to create a dynamic type for the REST operation. Use the input values below. Map ID property to subscriptionId of REST response. Map Name property to displayName.

-3c- Create a type from a REST operation



List objects operation *

Type name *

Icon *

Parent type *

Parent name and ID

Folder label *

hasCustomHeadersAction

Action to get custom headers

Content preview

Objects path

Element index

Object properties

+

Property	Value
<input type="checkbox"/> displayName	stephenmak

GET Subscriptions

subscriptions

default-16x16.png

AZUREHost

<https://management.azure.com/> : 19ebbcfb-ba3e-4f75-901d-b27ddd31fa5a

Subscriptions

getAzureCustomHeaders

{
 "value": [
 {
 ".value":
 0
 }
]
}

<input type="checkbox"/>	authorizationSource	RoleBased
<input type="checkbox"/>	tags	{"Environment":"INDIVIDUAL","CostCenter":"US1071310","ShortCostCenter":"71310","ManagementType":"OPEX","OwnerEmail":"smak@vmware.com","SecondaryKey": "D-1336769"}
Show Or Hide Columns		
ID property *	subscriptionId	
Name property *	displayName	

In case of an error, correct the error, delete the type and its actions then retry. Type delete instruction is provided in a section below.

Examine Subscriptions in inventory

Examine the DT object in inventory for error.

Inventory [C](#)

- > SNMP
- > PowerShell
- > vRealize Automation and Cloud Services
- > AMQP
- > vAPI
- > SOAP
- > vRO Configuration
- < Dynamic Types
 - < Type Hierarchy
 - > ExhibitsFolder
 - < AZUREHost
 - < subscriptionsFolder
 - < subscriptions
 - < AnimalsFolder
- < AZURE
 - < https://management.azure.com
 - < Subscriptions
 - > Error - click me for more information

x Error - click me for more information

fixAction	Check subscriptionsFolderFindRelation action
name	Error - click me for more information
namespace	AZURE
DEBUGRequestFullPath	https://management.azure.com/subscriptions
DEBUGobjectsPath	.value
id	error
DEBUGjson	{"error":{"code":"MissingApiVersionParameter","message":"The api-version query parameter (?api-version=) is required for all requests."}}
@fullType	DynamicTypes:AZURE.subscriptions
error	{"error":{"code":"MissingApiVersionParameter","message":"The api-version query parameter (?api-version=) is required for all requests."}}

Update subscriptionsFolderFindRelation action

Error: MissingApiVersionParameter

Solution: update urlTemplate as below

```

1 // Cache;
2 var cacheTimeout = 60 * 1; // 5 minutes cache
3 var objectId = "com.vmware.cse.dt.azure.subscriptionsFolderFindRelation(" + parentType + "," + parentId + "," + relationName +
4 var objects = System.getModule("com.vmware.coe.dynamicTypes.pluginGeneratorV3.caching").getFromCache(objectId);
5 if (objects != null) return objects;
6 System.log("SubscriptionFolderFindRelation()");
7 // REST Operation variables;
8 var operationName = "folder findRelation";
9 var method = "GET";
10 var urlTemplate = "/subscriptions?api-version=2023-07-01";
11 var content = "";
12 var defaultContentType = "";
13 var objectsPath = ".value"; // Edit this line if the objectsPath property is wrong
14 var cache = true;
15 var idProp = new Properties();
16 var contentAsStringLink;
17 var namespace = parentType.split(".")[0];
18 var childType = System.getModule("com.vmware.coe.dynamicTypes.pluginGeneratorV3").getChildType(parentType, parentId, relationName);

```

Inputs:

- parentType : string
- parentId : string
- relationName : string

Properties:

- Enums
- Types
- Plugins
 - AD

Return type: DynamicTypes:DynamicObj

Inputs:

ADD NEW INPUT

parentType	string
parentId	string
relationName	string

GET Resource Groups

Run workflow (3a) to add the REST operation. Use the input values below. Note: an input parameter {id} is used in URL.

-3a- Add a REST operation

Adds an operation to a REST host.

Properties to create a new operation. The URL must include only the specific operation part (without the host's URL) and placeholders for parameters that are provided at request run time. Examples: /customer/{id} /customer/{id} /customer?orderBy={orderBy}

Parent host *	https://management.azure.com: https://managemen...	×
Name *	GET Resource Groups	
Template URL *	/subscriptions/{id}/resourcegroups?api-version=2023-07-01	
HTTP method *	GET	

Template URL: /subscriptions/{id}/resourcegroups?api-version=2023-07-01

Test REST Operation

Run workflow (3b) to test the operation. Supply a subscription id as input.

-3b- Invoke a REST operation with headers from action

REST Operation *	GET Resource Groups
getCustomHeadersAction	getAzureCustomHeaders
inParametersValues	
<input style="width: 20px; height: 20px; border: none; background-color: #0072bc; color: white; font-size: 14px; border-radius: 50%;" type="button" value="+"/>	

Create resourcegroups type from REST operation

Run workflow (3c) to create a dynamic type for the REST operation. Use the input values below. Update parent type value. Map ID and Name to the corresponding property in REST response.

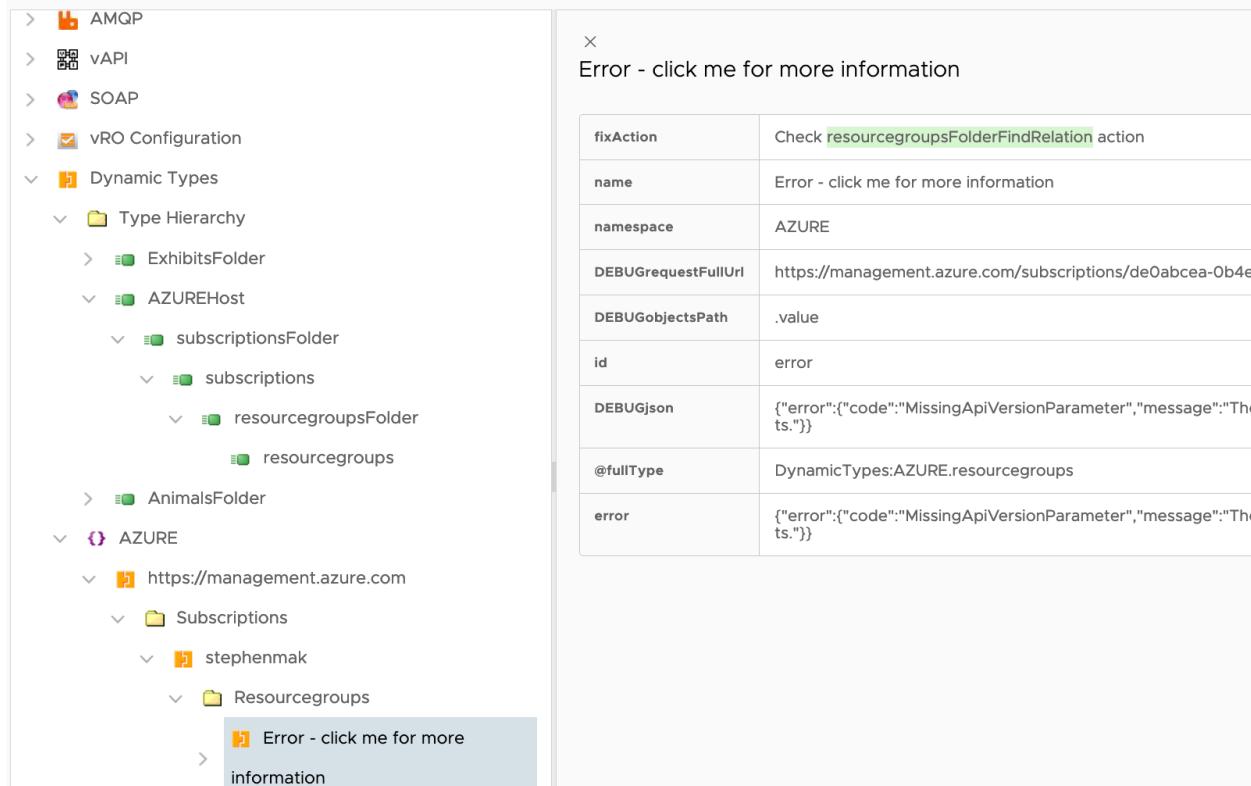
-3c- Create a type from a REST operation

List objects operation *	GET Resource Groups: GET /subscriptions/{id}/resourceGroups	
Type name *	resourcegroups	
Icon *	default-16x16.png	
Parent type *	subscriptions	
Parent name and ID	stephenmak : 19ebbcfb-ba3e-4f75-901d-000000000000	
Folder label *	Resourcegroups	
hasCustomHeadersAction	<input checked="" type="checkbox"/>	
Action to get custom headers	getAzureCustomHeaders	
Content preview	{ "value": [{ "key": "Content-Type", "value": "application/json" }, { "key": "Content-Length", "value": "123" }] }	
Objects path	.value	
Element index	0	
Object properties	<ul style="list-style-type: none"> <input style="border: 1px solid #ccc; border-radius: 50%; width: 1em; height: 1em; margin-right: 0.5em;" type="button" value="+"/> 	
<input type="checkbox"/> Property	<input style="border: 1px solid #ccc; border-radius: 50%; width: 1em; height: 1em; margin-right: 0.5em;" type="button" value="▼"/>	Value
<input type="checkbox"/> name		gp1
<input type="checkbox"/> location		eastus
<input type="checkbox"/> id		/subscriptions/deOabcea-0000-0000-0000-000000000000

Examine Resourcegroups in inventory

Examine the DT object in inventory for error.

Inventory



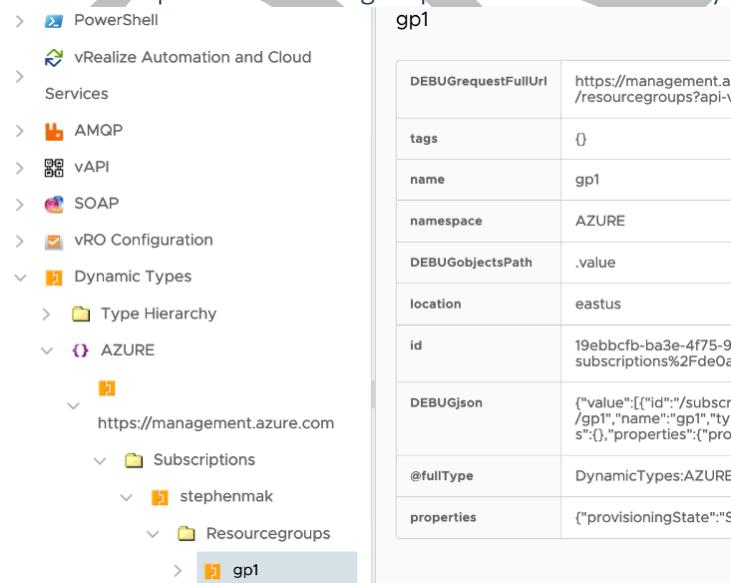
fixAction	Check resourcegroupsFolderFindRelation action
name	Error - click me for more information
namespace	AZURE
DEBUGrequestFullUrl	https://management.azure.com/subscriptions/de0abcea-0b4e...
DEBUGobjectsPath	.value
id	error
DEBUGjson	{ "error": { "code": "MissingApiVersionParameter", "message": "The..."}}
@fullType	DynamicTypes:AZURE.resourcegroups
error	{ "error": { "code": "MissingApiVersionParameter", "message": "The..."}}

Update resourcegroupsFolderFindRelation action

Error: MissingApiVersionParameter

Solution: update urlTemplate

Examine a specific resourcegroups instance in inventory



DEBUGrequestFullUrl	https://management.azure.com/resourcegroups?api-version=2017-05-01
tags	{}
name	gp1
namespace	AZURE
DEBUGobjectsPath	.value
location	eastus
id	19ebbcfb-ba3e-4f75-91.../subscriptions%2Fde0a...
DEBUGjson	{ "value": [{ "id": "/subscriptions%2Fde0a.../resourcegroups%2Fgp1", "name": "gp1", "type": "ResourceGroup", "properties": { "provisioningState": "S" } }] }
@fullType	DynamicTypes:AZURE
properties	{"provisioningState": "S"}

GET Network Interfaces

Run workflow (3a) to add the REST operation. Use the input values below. Note: two input parameters, {id} and {rg}, are used in URL.

-3a- Add a REST operation

Adds an operation to a REST host.

Properties to create a new operation. The URL must include only the specific operation part (without the host's URL) and can contain placeholders for parameters that are provided at run time.

Parent host *

Name *

Template URL *

HTTP method *

RUN **CANCEL**

Template URL:
`/subscriptions/{id}/resourceGroups/{rg}/providers/Microsoft.Network/networkInterfaces?
api-version=2023-09-01`

Test REST Operation

Run workflow (3b) to test the operation. Supply a value to each parameter in the order as listed in URL.

3b- Invoke a REST operation with headers from action

REST Operation *

getCustomHeadersAction

inParametersValues

<input type="checkbox"/>	
<input type="checkbox"/>	e0abced
<input type="checkbox"/>	gp1

content

RUN **CANCEL**

Create interfaces type from REST operation

Run workflow (3c) to create a dynamic type for the REST operation. Use the input values below. Update parent type value to resourcegroups. Map ID and Name to the corresponding property in REST response.

-3c- Create a type from a REST operation

List objects operation *

Type name *

Icon *

Parent type *

Parent name and ID

Folder label *

hasCustomHeadersAction

Action to get custom headers

Content preview

Objects path

Element index

Object properties

+ 

<input type="checkbox"/>	Property	Value
<input type="checkbox"/>	kind	Regular
<input type="checkbox"/>	name	mak179
<input type="checkbox"/>	etag	W/"ff7958e0-4a00-4795-819b-25e10c98c10e"
<input type="checkbox"/>	location	eastus

GET Network Interfaces

networkInterfaces

default-16x16.png

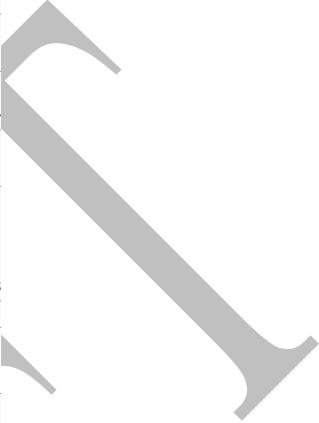
resourcegroups

gp1 : 19ebbcfb-ba3e-4f75-

NetworkInterfaces

getAzureCustomHeaders

{
 "value": [
 {
 ".value":
 0
 }
]
}



Examine NetworkInterfaces in inventory
Examine the DT object in inventory for error.

Inventory

- > SNMP
- > PowerShell
- > vRealize Automation and Cloud Services
- > AMQP
- > vAPI
- > SOAP
- > vRO Configuration
- > Dynamic Types
 - > Type Hierarchy
 - > AZURE
 - > https://management.azure.com
 - > Subscriptions
 - > stephenmak
 - > Resourcegroups
 - > gp1
 - > Networkinterfaces
 - > Error - click me for more information

Error - click me for more information

fixAction	Check networkInterfacesFolderFindRelation action
name	Error - click me for more information
namespace	AZURE
DEBUGrequestFullUrl	https://management.azure.com/subscriptions/de0abce4-0b4e-433b/resourceGroups/%2Fsubscriptions%2Fde0abce4-0b4e-433b/providers/Microsoft.Network/networkInterfaces
DEBUGobjectsPath	.value
id	error
DEBUGjson	{"error":{"code":"MissingApiVersionParameter","message":"The api-version query parameter (?api-version) is required for all requests."}}
@fullType	DynamicTypes:AZURE.networkInterfaces
error	{"error":{"code":"MissingApiVersionParameter","message":"The api-version query parameter (?api-version) is required for all requests."}}

Update networkInterfacesFolderFindRelation action

Error 1: MissingApiVersionParameter

Solution: update urlTemplate

networkInterfacesFolderFindRelation

RUN DEBUG DELETE FIND USAGES FIND DEPENDENCIES

General Script Version History Audit

Runtime Environment

JavaScript

Inputs:

parentType : string parentId : string relationName : string

```

1 // Cache;
2 var cacheTimeout = 60 * 1; // 5 minutes cache
3 var objectId = "com.vmware.cse.dt.azure.networkInterfacesFolderFindRelation(" + parentType + "," + parentId + "," + relationName + ")";
4 var objects = System.getModule("com.vmware.coe.dynamicTypes.pluginGeneratorV3.caching").getFromCache(objectId);
5 if (objects != null) return objects;
6
7 // REST Operation variables;
8 var operationName = "folder findRelation";
9 var method = "GET";
10 var urlTemplate = "/subscriptions/{id}/resourceGroups/{rg}/providers/Microsoft.Network/networkInterfaces";
11 // Parameters : ?api-version=2023-09-01";
12 var content = "";
13 var defaultContentType = "";
14 var objectsPath = ".value";// Edit this line if the objectsPath property is wrong
15 var cache = true;
16 var idProp = new Properties();
17 var contentAsStringLink;
18 var namespace = parentType.split(".")[0];
19 var childType = System.getModule("com.vmware.coe.dynamicTypes.pluginGeneratorV3").getChildType(parentType, parentId, relationName);
20 if (childType != null) {
21   try {
22     var shortType = childType.split(",")[1];
23     var fixAction = "Check " + shortType + "FolderFindRelation action";
24     var params = parentId.split("/");
25     var restHostId = params.shift(); // Removing the host ID
26     params = getParams(urlTemplate); // This is only needed when parent folder object URL has parameters in different order
  
```

API Explorer

Filter objects

System

- > Objects
- > Enums
- > Types

Plugins

- > AD

Properties

Return type

DynamicTypes:DynamicObject

Inputs

ADD NEW INPUT

parentType
parentId
relationName

Error 2: invalid subscription Id and resource group name

×

Error - click me for more information

fixAction	Check networkInterfacesFolderFindRelation action
name	Error - click me for more information
namespace	AZURE
DEBUGrequestFullUrl	https://management.azure.com/subscriptions/de0abcea-0b4e-433b-93f1-36d733c1-0bc5-4b48-adf4-f83dae459bd1/resourceGroups/%2Fsubscriptions%2Fde0abcea-0b4e-433b-93f1-36d733c1-0bc5-4b48-adf4-f83dae459bd1/resourceGroups%2Fgp1/providers/Microsoft.Network/networkInterfaces?api-version=2023-09-01
DEBUGobjectsPath	.value
id	error
DEBUGjson	{"error":{"code":"AuthorizationFailed","message":"The client '36d733c1-0bc5-4b48-adf4-f83dae459bd1' with object id '36d733c1-0bc5-4b48-adf4-f83dae459bd1' does not have authorization to perform action 'Microsoft.Resources/subscriptions/resourceGroups/de0abcea-0b4e-433b-93f1-36d733c1-0bc5-4b48-adf4-f83dae459bd1/read' over scope '/subscriptions/de0abcea-0b4e-433b-93f1-36d733c1-0bc5-4b48-adf4-f83dae459bd1/resourceGroups/subscriptions/de0abcea-0b4e-433b-93f1-36d733c1-0bc5-4b48-adf4-f83dae459bd1/resourceGroups/gp1/providers/Microsoft.Network/networkInterfaces' or the scope is invalid. If access was recently granted, please refresh your credentials."}}
@fullType	DynamicTypes:AZURE.networkInterfaces
error	{"error":{"code":"AuthorizationFailed","message":"The client '36d733c1-0bc5-4b48-adf4-f83dae459bd1' with object id '36d733c1-0bc5-4b48-adf4-f83dae459bd1' does not have authorization to perform action 'Microsoft.Resources/subscriptions/resourceGroups/de0abcea-0b4e-433b-93f1-36d733c1-0bc5-4b48-adf4-f83dae459bd1/read' over scope '/subscriptions/de0abcea-0b4e-433b-93f1-36d733c1-0bc5-4b48-adf4-f83dae459bd1/resourceGroups/subscriptions/de0abcea-0b4e-433b-93f1-36d733c1-0bc5-4b48-adf4-f83dae459bd1/resourceGroups/gp1/providers/Microsoft.Network/networkInterfaces' or the scope is invalid. If access was recently granted, please refresh your credentials."}}

Solution: Update subscriptionId and resourceGroupName value in action as below

```

networkInterfacesFolderFindRelation
RUN DEBUG DELETE FIND USAGES FIND DEPENDENCIES

General Script Version History Audit
Runtime Environment
JavaScript

Inputs:
parentType : string parentId : string relationName : string

1 // Cache;
2 var cacheTimeout = 60 * 1; // 5 minutes cache
3 var objectId = "com.vmware.cse.dt.azure.networkInterfacesFolderFindRelation(" + parentType + "," + parentId + "," + relationName + ")";
4 var objects = System.getModule("com.vmware.coe.dynamicTypes.pluginGeneratorV3.caching").getFromCache(objectId);
5 if (objects != null) return objects;
6
7 // REST Operation variables;
8 var operationName = "folder findRelation";
9 var method = "GET";
10 var subscriptionId = parentId.split("/")[1];
11 var rest = parentId.split("%2F");
12 var resourceGroupName = rest[rest.length - 1];
13 // update URL
14 var urlTemplate = "/subscriptions/" + subscriptionId + "/resourceGroups/" + resourceGroupName + "/providers/Microsoft.Network/networkInterfaces?api-version=2023-09-01";
15 System.debug(urlTemplate);
16 var content = "";
-- ...
-- ...
-- ...

// REST Operation variables;
var operationName = "folder findRelation";
var method = "GET";
var subscriptionId = parentId.split("/")[1];
var rest = parentId.split("%2F");
var resourceGroupName = rest[rest.length - 1];
// update URL
var urlTemplate =
"/subscriptions/" + subscriptionId + "/resourceGroups/" + resourceGroupName + "/providers/Microsoft.Network/networkInterfaces?api-version=2023-09-01";
System.debug(urlTemplate);

```

GET Virtual Machines

Run workflow (3a) to add the REST operation. Note: two input parameters, {subscriptionId} and {resourceGroupName}, are used in URL.

-3a- Add a REST operation

Adds an operation to a REST host.

Properties to create a new operation. The URL must include only the specific operation part (without the host's URL) and can contain placeholders for parameters that are provided at request run time. Examples: /customer/{id}
/customer?orderBy=(orderBy)

Parent host *	https://management.azure.com
Name *	GET Virtual Machines
Template URL *	/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines?api-version=2023-07-01
HTTP method *	GET
Template URL:	/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines?api-version=2023-07-01

Create virtualMachines type from REST Operation

Run workflow (3c) to create a dynamic type for the REST operation. Use the input values below. Update parent type value. Map ID and Name to the corresponding property in REST response.

-3c- Create a type from a REST operation

List objects operation *	GET Virtual Machines	<input type="button" value="X"/>									
Type name *	virtualMachines	<input type="button" value="X"/>									
Icon *	default-16x16.png	<input type="button" value="X"/>									
Parent type *	resourcegroups	<input type="button" value="X"/>									
Parent name and ID	gp1 : 19ebbcfb-ba3c-01d-b27ddd31fa5a/de0a1e01-45f1-4500-80f1-000000000000/gp1	<input type="button" value="X"/>									
Folder label *	VirtualMachines										
hasCustomHeadersAction	<input checked="" type="checkbox"/>										
Action to get custom headers	getAzureCustomHeaders	<input type="button" value="X"/>									
Content preview	<pre>{ "value": [{ "id": "1", "name": "Resource Group 1" }, { "id": "2", "name": "Resource Group 2" }] }</pre>										
Objects path	.value										
Element index	0	<input type="button" value="▼"/>									
Object properties	<input type="button" value="+"/> <table border="1"> <thead> <tr> <th></th> <th>Property</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>identity</td> <td>{"type": "SystemAssigned", "principalId": "32c9e953-07e8-443c-90659a8db", "tenantId": "b39138ca-3cee-412f-9000-000000000000", "clientSecret": "2f0f0000-0000-0000-0000-000000000000"}</td> </tr> <tr> <td><input type="checkbox"/></td> <td>name</td> <td>mak1</td> </tr> </tbody> </table>			Property	Value	<input type="checkbox"/>	identity	{"type": "SystemAssigned", "principalId": "32c9e953-07e8-443c-90659a8db", "tenantId": "b39138ca-3cee-412f-9000-000000000000", "clientSecret": "2f0f0000-0000-0000-0000-000000000000"}	<input type="checkbox"/>	name	mak1
	Property	Value									
<input type="checkbox"/>	identity	{"type": "SystemAssigned", "principalId": "32c9e953-07e8-443c-90659a8db", "tenantId": "b39138ca-3cee-412f-9000-000000000000", "clientSecret": "2f0f0000-0000-0000-0000-000000000000"}									
<input type="checkbox"/>	name	mak1									

Examine VirtualMachines in inventory

Examine the DT object in inventory for error.

The screenshot shows the vRealize Automation interface. On the left, there's a navigation tree with categories like SNMP, PowerShell, vRealize Automation and Cloud Services, AMQP, vAPI, SOAP, vRO Configuration, Dynamic Types, Type Hierarchy, AZURE, and a specific subscription entry for https://management.azure.com. On the right, a detailed configuration panel is open for a 'VirtualMachines' action. The panel includes fields for name (VirtualMachines), namespace (AZURE), urlWithParameters (/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines?api-version=2023-07-01), InParameters (subscriptionId,resourceGroupName), id (19ebbcfb-ba3e-4f75-901d-b27ddd31fa5a/...), and @fullType (DynamicTypes:AZURE.virtualMachinesFolder). A note at the bottom says 'Error - click me for more information'.

Update virtualMachinesFolderFindRelation action

Error 1: MissingApiVersionParameter

Solution: update urlTemplate

Error 2: invalid subscriptionId

An error dialog box is shown with the title 'Error - click me for more information'. Below it is a table titled 'fixAction' with the following rows:

fixAction	Check virtualMachinesFolderFindRelation action
name	Error - click me for more information
namespace	AZURE
DEBUGRequestFullUrl	https://management.azure.com/subscriptions/%2Fsubscriptions%2Fdfe0abc.../resourceGroups/%2FresourceGroups%2Fgp1/providers/Microsoft.Compute/virtualMachines?api-version=2023-07-01
DEBUGObjectsPath	.value
id	error
DEBUGJson	{"error":{"code":"InvalidSubscriptionId","message":"The provided subscription identifier 'subscriptions' is malformed or invalid."}}
@fullType	DynamicTypes:AZURE.virtualMachines
error	{"error":{"code":"InvalidSubscriptionId","message":"The provided subscription identifier 'subscriptions' is malformed or invalid."}}

Solution:

Update urlTemplate in virtualMachinesFolderFindRelation action as below.

```
// REST Operation variables;
var operationName = "folder findRelation";
var method = "GET";
var subscriptionId = parentId.split("/")[1];
var rest = parentId.split("%2F");
var resourceGroupName = rest[rest.length - 1];
// update URL
var urlTemplate =
"/subscriptions/" + subscriptionId + "/resourceGroups/" + resourceGroupName + "/providers/Microsoft.Compute/virtualMachines?api-version=2023-07-01";
System.debug(urlTemplate);
```

HTTPS Operations

Below are the HTTPS Operations after install.

The screenshot shows the 'Policy Runs' interface with a sidebar containing 'Assets', 'Packages', 'Configurations', 'Resources', 'Environments', 'Administration' (with 'Groups' selected), 'Inventory', and 'Audit Logs'. The main area displays a tree view under 'HTTP-REST' with several entries, including 'https://management.azure.com/' and 'sm-vra8.sqa.local'.

Input Form Actions

com.vmware.pve.azure module has pre-defined input form actions. Reader can run them with no changes.

The screenshot shows the configuration of the 'getAzureResourceGroupNames' action. It includes tabs for 'General', 'Script' (selected), 'Version History', and 'Audit'. The 'Script' tab contains a code editor with the following JavaScript:

```
subscriptionId : string
1 var type = "AZURE.resourcegroups";
2
3 var actionResult = System.getModule("com.vmware.cse.dt.azure").resourcegroupsFindAll(type);
4
5 var breakpoint = /%2F/ ;
6
7 var array = [];
8 //System.log(System.getObjectClassName(actionResult));
9 for (i in actionResult) {
10     for (j in actionResult[i]) {
11         System.log(actionResult[i][j].id);
12         string = actionResult[i][j].id;
13         System.log(string);
14         y = string.split("/");
15         if (y[1] == subscriptionId) {
16             z = string.split(breakpoint);
17             array.push(z[z.length-1]);
18         }
19     }
20 }
return array;
```

The right side of the screen shows the 'API Explorer' and 'Properties' sections. The 'API Explorer' shows a tree structure with 'System' and 'Plugins' nodes. The 'Properties' section shows the 'Return type' as 'Array' and an 'Inputs' section with a single input 'subscriptionId' of type 'string'.

getAzureNetworkInterfaceNames action

```

getAzureNetworkInterfaceNames
RUN DEBUG DELETE FIND USAGES FIND DEPENDENCIES
General Script Version History Audit
Runtime Environment
JavaScript
Inputs:
subscriptionId : string
resourceGroupName : string
1 var type = "AZURE.networkInterfaces";
2
3 var actionBar = System.getModule("com.vmware.cse.dt.azure").networkInterfacesFindAll(type);
4
5 var breakpoint = /%2F/ ;
6
7 var array = [];
8 //System.log(System.getObjectClassName(actionBar));
9 for (i in actionBar) {
10     //System.log(actionBar[i]);
11     string = actionBar[i].id;
12     y = string.split("/");
13     z = string.split("%2F");
14     if (y[1] == subscriptionId && z[8] == resourceGroupName ) {
15         z = string.split(breakpoint);
16         array.push(z[z.length-1]);
17     }
18 }
19 return array;

```

API Explorer

- System
 - Objects
 - Enums
 - Types
- Plugins
 - AD

Properties

Return type: string

Inputs:

subscriptionId	string	<input checked="" type="checkbox"/> Array	Description
resourceGroupName	string	<input checked="" type="checkbox"/> Array	Description

getAzureSubscriptionIds action

```

getAzureSubscriptionIds
RUN DEBUG DELETE FIND USAGES FIND DEPENDENCIES
General Script Version History Audit
Runtime Environment
JavaScript
Inputs:
1 var type = "AZURE.subscriptions";
2
3 var actionBar = System.getModule("com.vmware.cse.dt.azure").subscriptionsFindAll(type);
4
5 array = [];
6 for (i in actionBar) {
7     System.log(actionBar[i]);
8     array.push(actionBar[i].id.split("/")[1]);
9 }
10
11 return array;

```

API Explorer

- System
 - Objects
 - Enums
 - Types
- Plugins
 - AD

Properties

Return type: string

Inputs:

subscriptionId	string	<input checked="" type="checkbox"/> Array	Description
-----------------------	--------	---	-------------

Provisioning Workflows

Folder AZURE/Default contains pre-defined workflows. Reader needs create a REST operation and **assign** it to restOperation parameter in workflow before use. In addition, reader may need re-attach input form actions and field inputs to workflow.

Invoke 'Create VM: PUT https://management.azure.com/_sub'.

Use below url to create a REST operation:

```

PUT
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName}?api-version=2023-09-01

```

The screenshot shows a software interface for managing workflows and their schema.

Left Panel (Workflows):

- Workflows
 - ACoE
 - ASB
 - Machine
 - AZURE
 - AZURE CRUD Operations
 - Copy of test DT actions
 - Invoke 'CREATE SQL Managed Instance: PU'
 - Invoke 'Create VM: PUT https://management.azure.com/resources/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName}?api-version=2023-09-01'**
 - Invoke 'DELETE vm2 object'
 - Invoke 'DELETE vm: DELETE https://management.azure.com/resources/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName}'
 - Content Management Tests
 - CSE
 - Dynamic Types Plug-in generator v3
 - Create a plug-in no swagger
 - 3- Repeat for each object type
 - 3a- Add a REST operation
 - 3b- Invoke a REST operation with headers
 - 3c- Create a type from a REST operation
 - 4- Remove a REST operation (Optic)
 - 1- Create new plug-in
 - 2a- Create a REST host
 - 2b- Add REST Host as DT Host
 - Create a plug-in with Swagger
 - Helper

Top Right Panel (Variables):

Name	Type
errorCode	string
restOperation	REST:RESTOperation
method	PUT
dunesId	19ebbcfb- ba3e-4f75-901d- b27ddd31fa5a: 14ef768c-2043-4b14-8f2 d-c06a67131c07
displayName	Create VM: PUT https://management.azure.com/resources/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName}?api-version=2023-09-01
@type	RESTOperation
inParametersCount	3
urlTemplate	https://management.azure.com/resources/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName}?api-version=2023-09-01
name	Create VM
id	19ebbcfb-

Bottom Panel (Workflow Details):

Workflow Diagram:

```

graph TD
    Start(( )) --> DefinePayload[Define Payload]
    DefinePayload --> InvokeAction[InvokeRestOperationWithHeadersFromAction]
    InvokeAction --> Decision{Custom Condition}
    Decision -- True --> End(( ))
    Decision -- False --> Error(( ))
  
```

Workflow Schema (Schema Tab):

Details for the `invokeRestOperationWithHeadersFromAction` action:

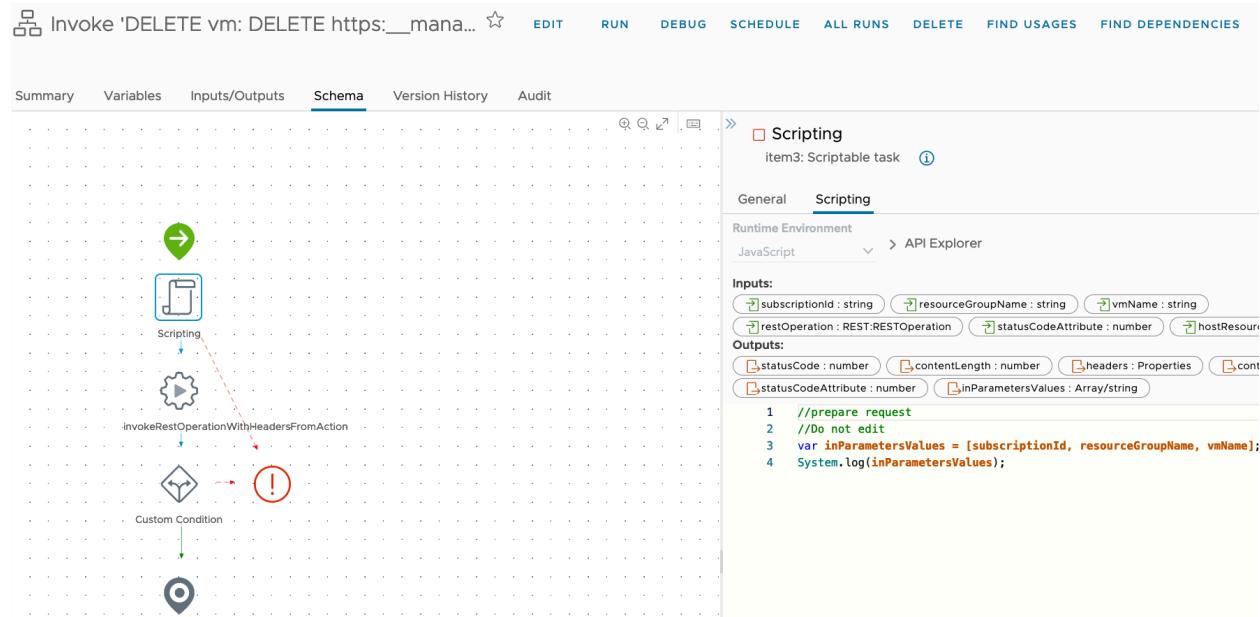
- General:**
 - Name: invokeRestOperationWithHeadersFromAction
 - Next element: item2: Custom Condition
 - Business status:
 - Description: Add a note to the workflow schema.
- Action:** invokeRestOperationWithHeadersFromAction
- Inputs:**
 - restOperation: REST:RESTOperation
 - inParametersValues: Array/string
 - content: string
 - getCustomHeadersAction: Action
- Outputs:** getCustomHeadersAction
 - actionResult: Properties
- Exception handling:**

Invoke 'DELETE vm: DELETE https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName} ?api-version=2023-09-01

Use below url to create a REST operation:

DELETE

https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName} ?api-version=2023-09-01

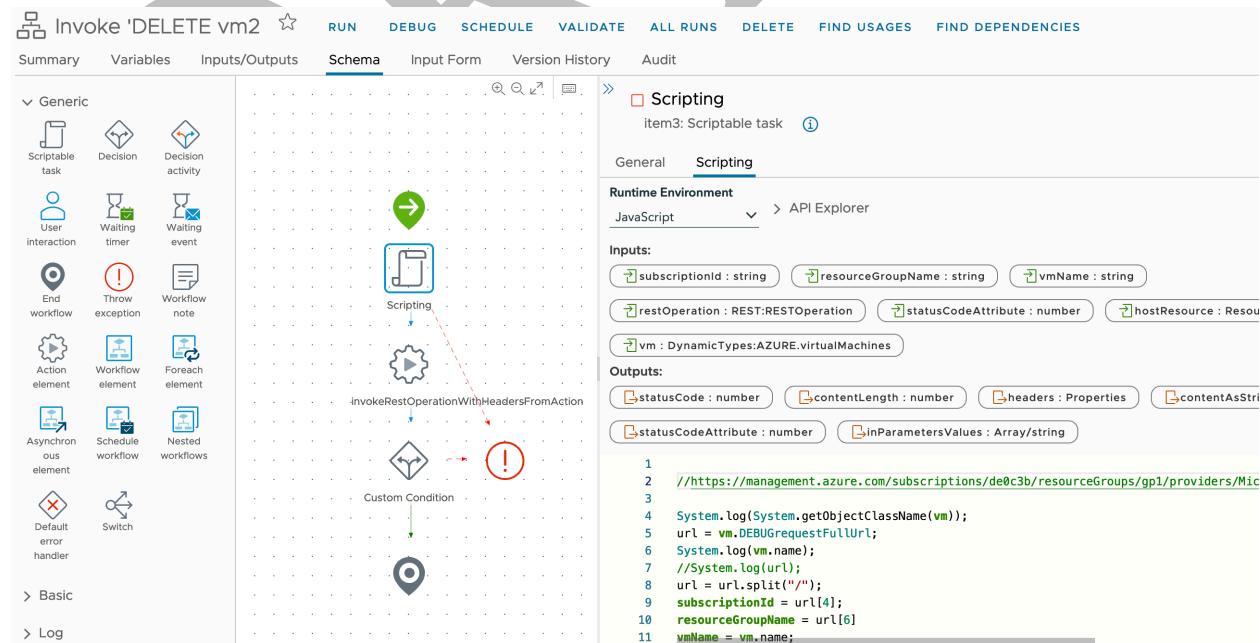


Invoke 'DELETE vm2

Use below url to create a REST operation:

DELETE

https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName} ?api-version=2023-09-01



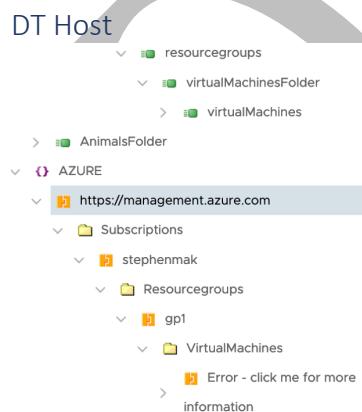
com.vmware.cse.dt.azure module

The tool auto-generates a list of plugin actions. Table below highlights all actions of resourcegroups Dynamic Type.

NEW ACTION		Tags
Name		
virtualMachinesFindAll		com.vmware.cse.dt.azure
virtualMachinesFolderFindAll		com.vmware.cse.dt.azure
virtualMachinesFolderFindById		com.vmware.cse.dt.azure
virtualMachinesFolderHasChildrenInRelation		com.vmware.cse.dt.azure
resourcegroupsFolderFindRelation		com.vmware.cse.dt.azure
resourcegroupsFindById		com.vmware.cse.dt.azure
resourcegroupsFindRelation		com.vmware.cse.dt.azure
resourcegroupsFindAll		com.vmware.cse.dt.azure
resourcegroupsFolderFindAll		com.vmware.cse.dt.azure
resourcegroupsFolderHasChildrenInRelation		com.vmware.cse.dt.azure
subscriptionsFolderFindRelation		com.vmware.cse.dt.azure
subscriptionsFindById		com.vmware.cse.dt.azure
subscriptionsFindRelation		com.vmware.cse.dt.azure

Inventory Objects and Properties

Below are some inventory objects after install.



https://management.azure.com	
name	https://management.azure.com
namespace	AZURE
id	19ebbcfb-ba3e-4f75-901d-b27d
@fullType	DynamicTypes:AZURE.AZUREH
url	https://management.azure.com

Subscriptions

virtualMachinesFolder

virtualMachines

AnimalsFolder

AZURE

- https://management.azure.com
- Subscriptions
- stephenmak**
- Resourcegroups
- gp1
- VirtualMachines
- Error - click me for more information

Zoo

SSH

OvaTransfer

Active Directory

vRO Multi-Node

Auto Deploy

vSphere vCenter Plug-in

HTTP-REST

stephenmak

DEBUGRequestFullUrl	https://management.azure.com/subscriptions/19ebbcfb-ba3e-4f75-901d-b27/resourceGroups/gp1
subscriptionPolicies	{"locationPlacementPolicy": "Standard"}
tags	{"Environment": "Test", "Owner": "Stephen Mak", "Manager": "Stephen Mak", "LastModified": "2018-05-10T14:45:00Z", "LastModifiedBy": "Stephen Mak", "LastModifiedEmail": "smak@contoso.com"}
name	stephenmak
tenantId	b39138ca-3cee-4a3a-8a2d-000000000000
namespace	AZURE
DEBUGObjectsPath	.value
state	Enabled
id	19ebbcfb-ba3e-4f75-901d-b27
managedByTenants	None
DEBUGJson	{"value": [{"id": "/subscriptions/19ebbcfb-ba3e-4f75-901d-b27/resourceGroups/gp1", "tags": {"Environment": "Test", "Owner": "Stephen Mak", "Manager": "Stephen Mak", "LastModified": "2018-05-10T14:45:00Z", "LastModifiedBy": "Stephen Mak", "LastModifiedEmail": "smak@contoso.com"}, "name": "gp1", "type": "Microsoft.Resources/resourceGroups", "location": "eastus", "resourceType": "ResourceGroup", "provisioningState": "Succeeded"}]}
@fullType	DynamicTypes:AZURE.subscription
subscriptionId	de0abcea-0b4e-4a3a-8a2d-000000000000
authorizationSource	RoleBased

Resource groups

Dynamic Types

Type Hierarchy

ExhibitsFolder

AZUREHost

subscriptionsFolder

subscriptions

resourcegroupsFolder

resourcegroups

virtualMachinesFolder

virtualMachines

AnimalsFolder

AZURE

https://management.azure.com

Subscriptions

stephenmak

Resourcegroups

gp1

VirtualMachines

gp1

DEBUGRequestFullUrl	https://management.azure.com/subscriptions/19ebbcfb-ba3e-4f75-901d-b27/resourceGroups/gp1
tags	{}
name	gp1
namespace	AZURE
DEBUGObjectsPath	.value
location	eastus
id	19ebbcfb-ba3e-4f75-901d-b27
DEBUGJson	{"value": [{"id": "/subscriptions/19ebbcfb-ba3e-4f75-901d-b27/resourceGroups/gp1", "tags": {}, "name": "gp1", "type": "Microsoft.Resources/resourceGroups", "location": "eastus", "resourceType": "ResourceGroup", "provisioningState": "Succeeded"}]}
@fullType	DynamicTypes:AZURE.resourceGroup
properties	{"provisioningState": "Succeeded"}

VirtualMachines

name	VirtualMachines
namespace	AZURE
urlWithParameters	/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName}
InParameters	subscriptionId,resourceGroupName,vmName
id	19ebbcfb-ba3e-4f75-901d-3b%2FresourceGroups%2Fmak1/providers/Microsoft.Compute/virtualMachines/mak1
@fullType	DynamicTypes:AZURE.virtualMachine

Virtual Machine Instance

DEBUGRequestFullUrl	https://management.azure.com/subscriptions/3cb-9834-3944663ions/OmsAgentForWindows/providers/Microsoft.Compute/virtualMachines/mak1?api-version=2015-06-15
resources	[{"id": "/subscriptions/3cb-9834-3944663ions/OmsAgentForWindows/providers/Microsoft.Compute/virtualMachines/mak1"}]
identity	{"type": "SystemAssigned", "id": "3cee-4b4a-a4d6-cc33e00000000000"}
name	mak1
namespace	AZURE
DEBUGObjectsPath	.value
location	eastus
id	19ebbcfb-ba3e-4f75-Fde0abcea-0b4e-40b4e-43cb-9834-3alMachines%2Fmak1
DEBUGJson	{ "value": [{ "name": "ups/gp1/providers/this", "location": "eastb510659a8db", "tier": { "vmSize": "StandardD9", "storagePerformance": "Standard", "sku": "20_0e", "os": "Linux", "name": "e", "caching": "ReadDe0abcea-0b4e-431_OsDisk_1_e64ebisks": [], "diskController": "LinuxConfiguration", "user": "ssh/auth", "UBfxYVTEXGf2ynNG8tvKW5d5Vyd+3LPDbNzzBEAnb" } }] }

Add additional REST Operations

Reader can map a REST call to a vRO workflow. Our tool creates a vRO workflow with inputs and Azure authentication. Below are the instructions.

- a) Run below workflow to add a REST Operation. Note: three input parameters are used in URL.

Add a REST operation

Adds an operation to a REST host.

Properties to create a new operation. The URL must include only the specific operation part (without the host's URL) and can contain placeholders for parameters that are provided at request run time. Examples: /customer/{id} ; /customer/{id}/orders?date=(date) ; /customer?orderBy=(orderd

Parent host *	<input type="text" value="https://management.azure.com/"/>
Name *	<input type="text" value="CREATE SQL Managed Instance"/>
Template URL *	<input type="text" value="management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Sql/managedInstances/{managedInstanceName}?api-version=2023-07-01"/>
HTTP method *	<input type="text" value="PUT"/>
Content type	<input type="text" value="application/json"/>

RUN **CANCEL**

Template URL:

`https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Sql/managedInstances/{managedInstanceName}?api-version=2023-07-01`

- b) Run below workflow to generate a workflow from the REST Operation

Generate a new workflow from a REST operation

Generates a new workflow from a REST operation.

Operation **Workflow properties**

Specify an operation to generate a workflow from. If the operation takes input and XSD schemas are added to its host, you can optionally specify the request input type.

Operation *	<input type="text" value="CREATE SQL Managed Instance"/>
Content type	<input type="text" value="application/json"/>

RUN **CANCEL**

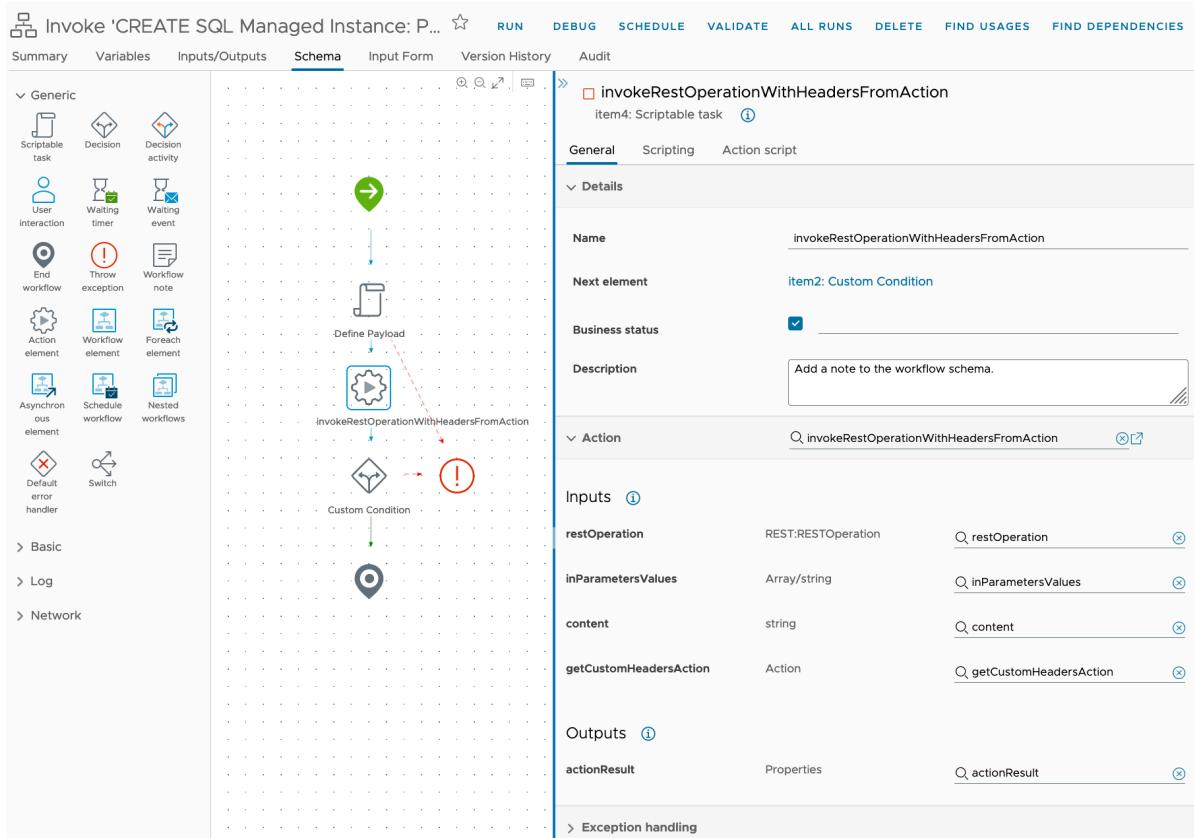
Operation **Workflow properties**

Specify workflow name and category to save to.

Name *

Folder *

- c) Add two new tasks, Define Payload and invokeRestOperationWithHeadersFromAction, to workflow



- d) Edit “Define Payload” task. Paste in scripting the GET response of a resource instance in JSON format (e.g., virtual machine). Update keys and values accordingly. See an example in pre-defined workflow, ‘Create VM: PUT’
https://management.azure.com/_sub/

```
GET
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName} ?api-version=2023-09-01
```

- e) Edit invokeRestOperationWithHeadersFromAction. Assign
`getAzureCustomHeadersAction` to input `getCustomHeadersAction`.

Invoke 'CREATE SQL Managed Instance: PUT https://__manage...'       

Summary **Variables** Inputs/Outputs Schema Input Form Version History Audit

View, edit, and create local variables for your workflow. 

     	Name	Value	Type	Description
 			string	
 		[object REST:RESTOperation]	REST:RESTOperation	
 			number	
 		[object ResourceElement]	ResourceElement	
 		Not set	Array/string	
 		[object Action]	Action	
	createdAt	1704993072145		
	editable	true		
	canEdit	true		
	name	getAzureCustomHeaders		
	description	Get authorization header for Azures Stephen Mak, Product Value Engineering, Broadcom		
	@returnType	Properties		
	id	d3fafe39-59fb-4b6d-8340-74198b 2e9525		
	type	ScriptModule		
	globalTags	com.vmware.coe.dynamicTypes.pl uginGeneratorV3.getCustomHead er...:__SYSTEM_TAG__		
	categoryName	com.vmware.coe.dynamicTypes.pl uginGeneratorV3.getCustomHead ersActions		
	version	0.0.0		
	updatedAt	1706833070915		
 			string	content
 			string	Response content as string
 		Not set	Properties	Response headers
 			number	Response content length
 			Properties	

f) Apply input form actions (e.g., getAzureSubscriptionIds)

Invoke 'CREATE SQL Managed Instance: P...' ⚡

RUN DEBUG SCHEDULE VALIDATE ALL RUNS DELETE FIND USAGES FIND DEPENDENCIES

Summary Variables Inputs/Outputs Schema Input Form Version History Audit

Request Validations

Workflow Inputs

- subscriptionId Added
- resourceGroupName Added
- managedInstanceName Added

Url parameters

subscriptionId

resourceGroupName

managedInstanceName

Properties

subscriptionId

Field ID: subscriptionId

Appearance	Values	Constraints
Label and type		
Label	subscriptionId	
Placeholder	Enter placeholder	
Data type	String	
Display type	DropDown	
> Visibility	Yes	
> Read-only	No	
> Custom help		

Invoke 'CREATE SQL Managed Instance: P...' ⚡

RUN DEBUG SCHEDULE VALIDATE ALL RUNS DELETE FIND USAGES FIND DEPENDENCIES

Summary Variables Inputs/Outputs Schema Input Form Version History Audit

Request Validations

Workflow Inputs

- subscriptionId Added
- resourceGroupName Added
- managedInstanceName Added

Url parameters

subscriptionId

resourceGroupName

managedInstanceName

Properties

subscriptionId

Field ID: subscriptionId

Appearance	Values	Constraints
Default value Constant		
Value source	Constant	
Value	Enter value	
Value options External source		
Value source	External source	
Select action	com.vmware.cse.dt.azure/getAzur ...	

g) Test workflow

Invoke 'CREATE SQL Managed Instance: PUT https://__manage...'

Autogenerated workflow.

subscriptionId *	d...
resourceGroupName *	gp1
managedInstanceName *	SQL_Test_1

RUN **CANCEL**

References

Authentication Failure

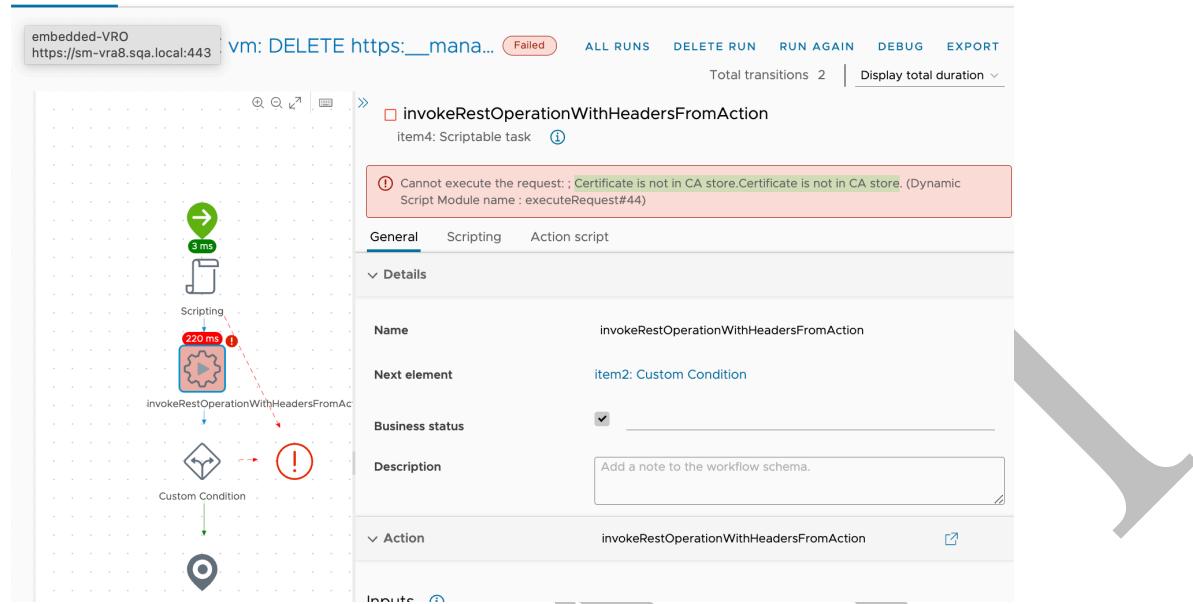
```
{"error": {"code": "AuthenticationFailed", "message": "Authentication failed. The 'Authorization' header is missing."}}
```

Troubleshooting:

- 1) Test getAzureCustomHeaders action

Solution: run update rest operation workflow

Cert Error

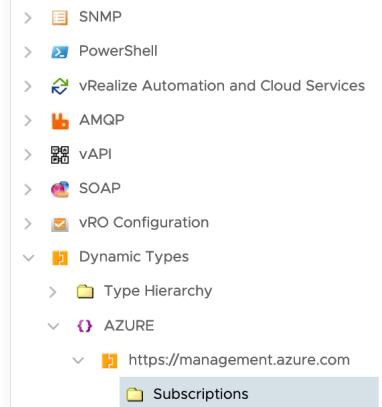


Solution: examine DT inventory objects. If nothing was displayed, below cert is the issue.
Manually reimport the certificate in vRO control center.

The screenshot shows the 'Certificates' section of the vRO Control Center. The 'TRUSTED CERTIFICATES' tab is selected. Below it, there is a sub-section titled 'IMPORT FROM URL' with a 'URL' input field containing 'https://management.azure.com'. There are also 'Proxy URL', 'Port', and 'Use Proxy' fields, along with 'CANCEL' and 'IMPORT' buttons.

Inventory Issue

Inventory

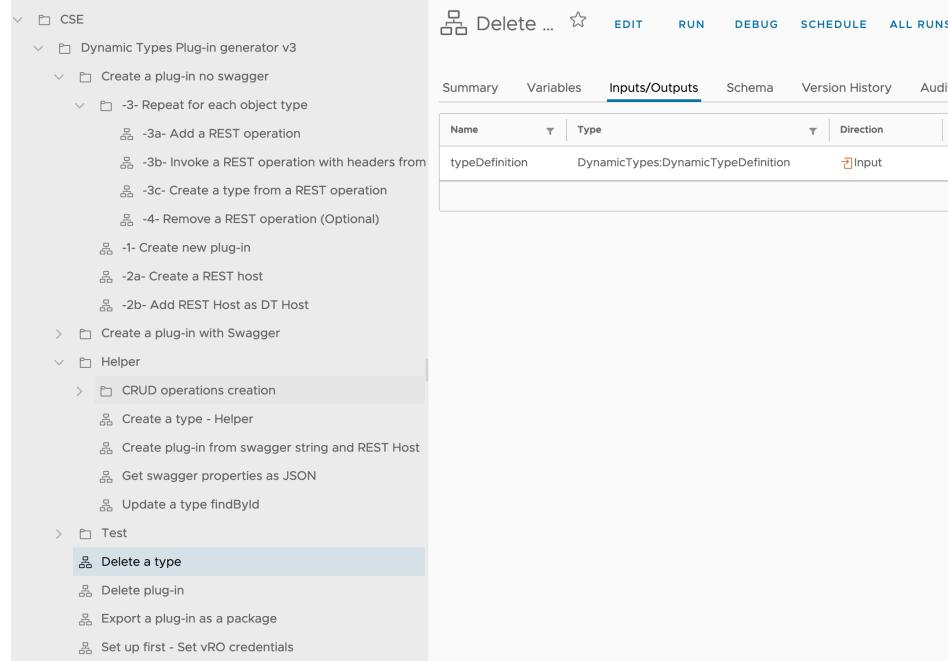


name	Subscriptions
namespace	AZURE
urlWithParameters	/subscriptions?api-version=2023-07-01
InParameters	
id	19ebbcfb-ba3e-4f75-901d-b27ddd31fa5a
@fullType	DynamicTypes:AZURE.subscriptionsFolder

Solution: examine inventory objects. If nothing was displayed, cert is the issue. Manually reimport the management.azure.com certificate.

Delete a DT type

- 1) Run 'Delete a type' workflow two times. First, select a child type in Type Hierarchy as input (e.g., virtualMachines). Then, select its parent (e.g., virtualMachineFolder).



- 2) Delete all actions of both types from com.vvware.cse.dt.azure module (e.g., subscriptions, subscriptionsFolder). See the list below.

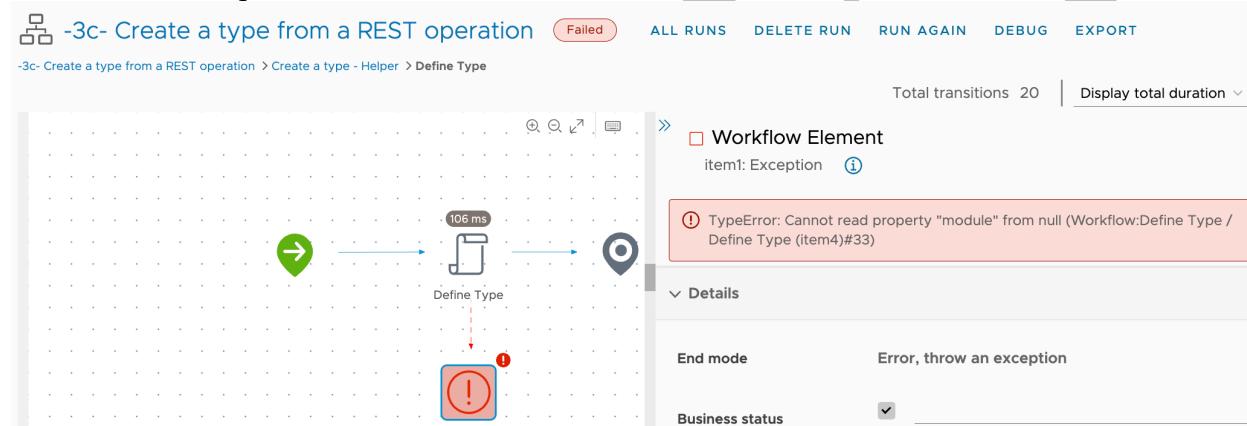
com.vmware.cse.dt.azure

EDIT **DELETE**

+ NEW ACTION **MOVE** **DELETE**

Name	Type
subscriptionsFindAll	Action
subscriptionsFindById	Action
subscriptionsFindRelation	Action
subscriptionsFolderFindAll	Action
subscriptionsFolderFindById	Action
subscriptionsFolderFindRelation	Action
subscriptionsFolderHasChildrenInRelation	Action

Create a Type from REST Operation issue
Below is an example.



Solution: delete the type and its actions. Then recreate the type. If necessary, rollback and retry.

Rollback and Retry

If install didn't produce the expected result, it is necessary to delete the namespace and its dynamic types. To do so, reader deletes DT types in the reversed order and runs the 'remove namespace' workflow to delete namespace at the end. In addition, reader deletes all actions in com.vmware.cse.dt.azure module.

Azure Managed instances

Below are links about Azure Managed Instances. Each can be created via a REST call.

<https://learn.microsoft.com/en-us/rest/api/sql/managed-instances?view=rest-sql-2021-11-01>

<https://learn.microsoft.com/en-us/rest/api/sql/managed-instances/update?view=rest-sql-2021-11-01&tabs=HTTP>

<https://learn.microsoft.com/en-us/azure/azure-sql/managed-instance/quickstart-content-reference-guide?view=azuresql>

<https://learn.microsoft.com/en-us/rest/api/sql/managed-instances/create-or-update?view=rest-sql-2021-11-01&tabs=HTTP>

PUT

https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Sql/managedInstances/{managedInstanceName}?api-version=2023-07-01

Azure REST Reference Page

<https://learn.microsoft.com/en-us/rest/api/azure/>

Azure search service Screenshot

Validation Success

Basics

Subscription	stephenmak
Resource Group	gp1
Location	West US
Service name	(new) maks
Pricing tier	free (50 MB, max 1 replicas, max 1 partitions, max 1 search units)
Estimated cost per month	\$0.00

Scale

Replicas	1
Partitions	1

Create Previous Next Download a template for automation

Create Application Secret

To access resources of a subscription, you create a secret for the application. If you don't have an application, see "Service principal" section to create one.

Below are screenshots to create a secret. <https://learn.microsoft.com/en-us/rest/api/servicebus/get-azure-active-directory-token#use-postman-to-get-the-azure-ad-token> is another example.

Service principal

Instead of having applications sign in as a fully privileged user, Azure offers service principals. An Azure service principal is an identity created for use with applications, hosted services, and automated tools. This identity is used to access resources.

See link for instructions to create a Microsoft Entra application and service principal with scope and role. <https://learn.microsoft.com/en-us/entra/identity-platform/howto-create-service-principal-portal>.

Azure offers four scope levels. A management group sees all subscriptions of the tenant. <https://learn.microsoft.com/en-us/azure/role-based-access-control/scope-overview> has the detail.

Below is an example of assigning a role to a service principal.

Home > Subscriptions > Subscriptions > stephenmak | Access control (IAM) > Add role assignment ...

Role Members Conditions Review + assign

Selected role Reader

Assign access to User, group, or service principal Managed identity

Members + Select members

Name	Object ID	Type
smak-app1	99c1f24f-ab9f-4c9e-aca9-f77c52da616f	App

Description Optional

Dynamic Type Generator Acknowledgement

Dynamic Type Generator v3 was created by Chris Decanini of VMWare/Broadcom. Version 3.1 has below updates over version 3 and has been regression tested with vRA8.14.1, vRA8.11.1 and cloud extensibility proxy.

<input checked="" type="checkbox"/> Invoke 'DELETE vm: DELETE https://management.azure.com...'		Higher Version	Workflow	0.0.1
<input checked="" type="checkbox"/> networkInterfacesFindAll		Higher Version	Action	0.0.1
<input checked="" type="checkbox"/> Invoke 'CREATE SQL Managed Instance: PUT https://management...'		Higher Version	Workflow	0.0.1
<input checked="" type="checkbox"/> getAzureResourceGroupNames		Higher Version	Action	0.0.0
<input checked="" type="checkbox"/> getAzureNetworkInterfaceNames		Higher Version	Action	0.0.0
<input checked="" type="checkbox"/> getAzureSubscriptionIds		Higher Version	Action	0.0.1
<input checked="" type="checkbox"/> subscriptionsFindAll		Higher Version	Action	0.0.1
<input checked="" type="checkbox"/> Invoke 'Create VM: PUT https://management.azure.com/_sub...'		Higher Version	Workflow	0.0.1
<input checked="" type="checkbox"/> resourcegroupsFindAll		Higher Version	Action	0.0.1
<input checked="" type="checkbox"/> Test DT actions		Higher Version	Workflow	0.0.1
<input checked="" type="checkbox"/> executeOrchestratorRequestWithConfigurationElement		Higher Version	Action	0.0.0
<input checked="" type="checkbox"/> getAzureCustomHeaders		Higher Version	Action	0.0.0
<input checked="" type="checkbox"/> executeOrchestratorRequest		Higher Version	Action	0.0.1
<input checked="" type="checkbox"/> resourcegroupsFindByid		Higher Version	Action	0.0.1
<input checked="" type="checkbox"/> Invoke 'DELETE vm2 object'		Higher Version	Workflow	0.0.1
<input checked="" type="checkbox"/> microsoftonline		Higher Version	Configuration Element	0.0.0
<input checked="" type="checkbox"/> 16				
Import configuration attribute values				
Import configuration SecureString attribute values				
Tag import mode		Import tags but preserve existing values.		

