

vRealize Operations Management

Super Metrics and XML Editing

Prepared by: Jeremy Wheeler
VMware Professional Services
wheeler@vmware.com

March 2015

Table of Contents

1. Document Overview	3
2. Lab Configuration	4
2.1 Lab Setup: Custom Group	4
2.2 Lab Setup: Super Metric	5
2.3 Lab Setup: New Policy	7
3. Creating a custom XML file	8
4. Retrieving your object's identifier ID (API Call #1)	9
5. Retrieving your metric description (API Call #2)	10
6. Modify your XML file	11
7. Update your widget	12
8. Validate your XML changes	13

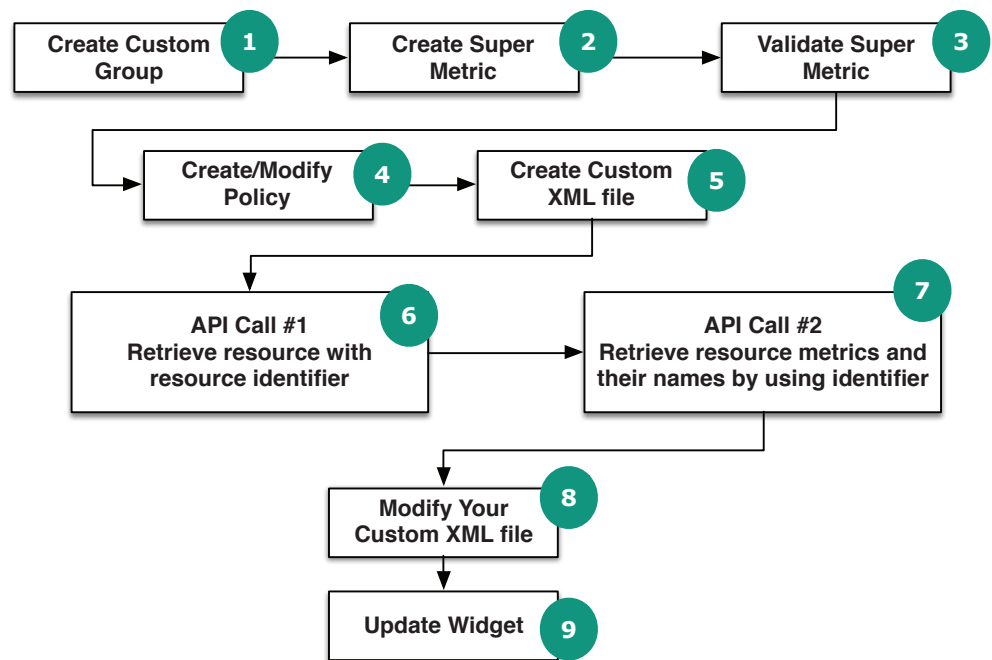
vRealize Operations Management SuperMetrics and XML Editing

1. Document Overview

How to find the metric description you can use for metric configuration in XML files. As with vROps 6 the way to retrieve the metric names from v5 doesn't work anymore. This document explains how to utilize the REST API (Suite API) to achieve what some are starting to struggle with.

The word 'Lab:' in this document references what steps I performed in my lab.

The following flow-chart explains the high-level steps involved to present your Super Metric onto a Dashboard.

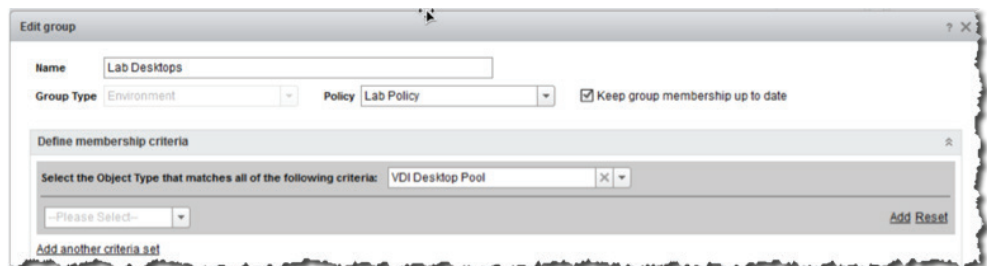


2. Lab Configuration

2.1 Lab Setup: Custom Group

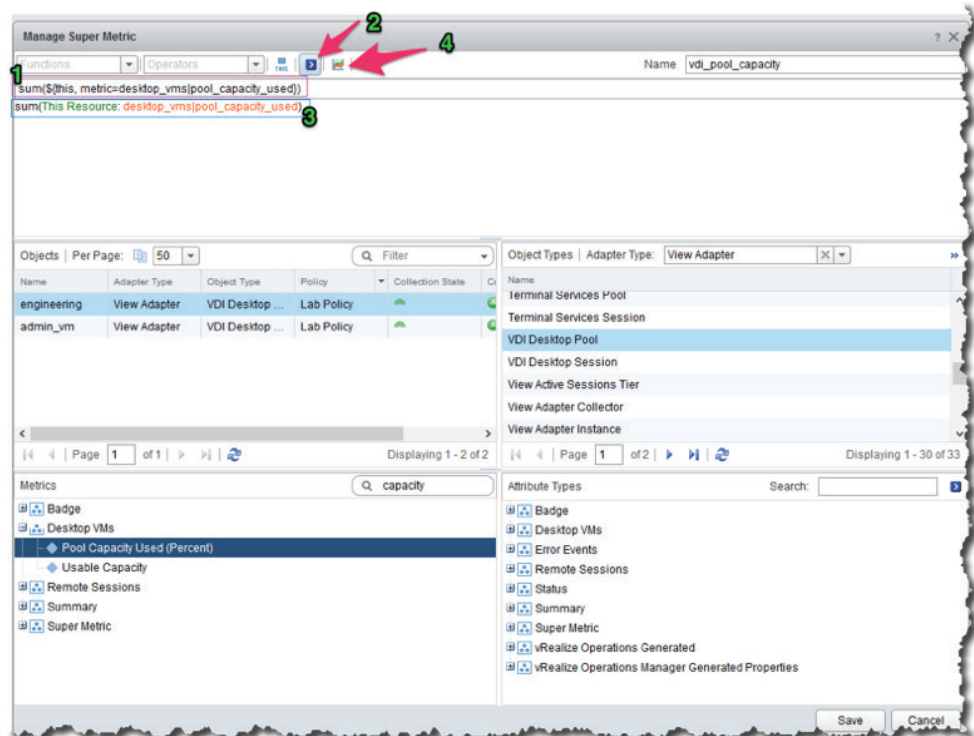
In my lab I created the following:

1. Custom Group
 - Name: Lab Desktops
 - Group Type: Environment
 - Policy: Default Policy
 - Membership criteria:
 - Criteria: VDI Desktop Pool



2.2 Lab Setup: Super Metric

Objective: For my testing, I am determining the capacity used for a View Pool.

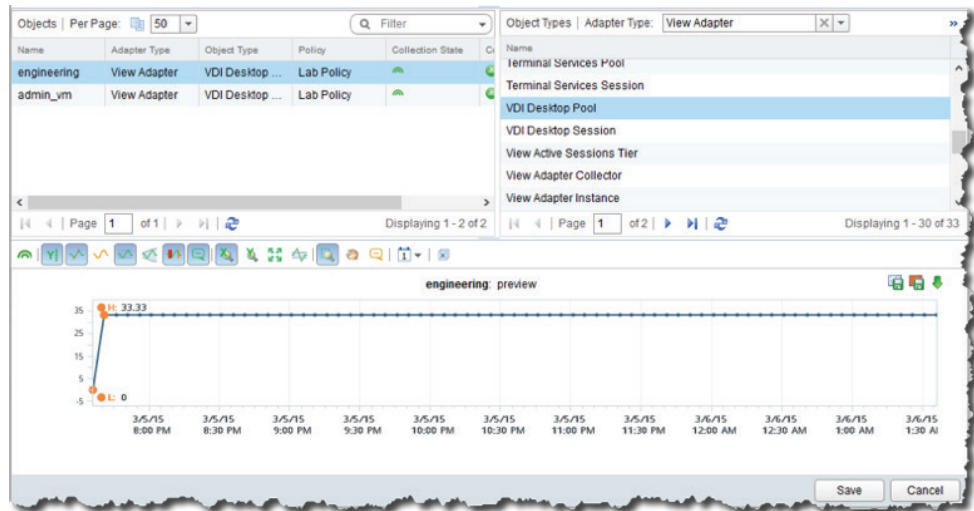


Super Metric creation and testing steps:

- In section (1) of this picture you will enter your Super Metric formula or build your Super Metric based on the Adapter Type, Objects, Object Types, Metrics, Attribute types.
- Once you complete your formula select (2) (blue play icon) to test your formula.
- After testing your formula's syntax window (3) will populate with the short-form formula.
- Select (4) (the graph icon) to test the Super Metric. Note: You may need to select a timeframe to see results (see picture on next page for an example.)

My formula: `sum($this.metric:desktop_vms|pool_capacity_used)`

Example of executing a Super Metric through the graph icon:



2.3 Lab Setup: New Policy

My next step is to create new or modify a policy to make sure the Super Metric is active for the objects you want it to be assigned for.

Note: After assigning a Super Metric to a policy it should take 2 cycles (10 minutes) maximum till the Super Metric is visible and accessible via the “All Metrics” tab below troubleshooting. Assuming the related metrics are collected in 5min interval either.

- Name: Lab Policy
- Base Policy: vSphere Solution’s Default Policy
- Override Attributes
 - Super Metric|vdi_pool_capacity
 - Type: Super Metric
 - Adapter Type: All Adapter Types
 - Object Type: All Object Types
 - State: Local
 - KPI: Inherited
- Super Metric|vdi_pool_capacity
 - Type: Super Metric
 - Adapter Type: View Adapter
 - Object Type: VDI Desktop Pool
 - State: Local
 - KPI: Inherited
- Apply policy to Group
 - Lab Desktops

3. Creating a custom XML file

Create a copy of the appropriate XML file; do NOT edit canned XML files.

- Copy XML process
 - Select the canned XML file
 - Select the pencil to edit
 - Select all contents (Ctrl-A) and copy into clipboard (Ctrl-C)
 - Click Cancel to stop editing the canned XML
 - Open your web browser to the following URL
 - <http://codebeautify.org/xmlviewer>
 - Paste the XML contents into the XML input field and select the middle button 'Beautify'
 - Select the two layered notebooks in the upper right or copy the contents in the Result dialog to clipboard
 - Select the green '+' to add a new XML file
 - Enter the name of your new XML

Lab: *In my lab we will call this file lab_desktop.xml*

- Past the newly formatted contents into the XML file

Note: The next steps outline the process to inject a Super Metric into an XML. One requirement that should be identified is the internal name for the metric (Super metric) is required. There is a 2-way process.

1. Retrieve resource with resource identifier.
2. Use the resource identifier to retrieve resource metrics and their names.

4. Retrieving your object's identifier ID (API Call #1)

Navigate to the following location:

- From Home select Environment
- Select
 - Inventory Tree that pertains to your Super Metric, or
 - All Objects and drill into the needed object or
 - Your custom group
- Select your object under 'Related Objects'
- Click the 'Troubleshooting' tab
- Select 'All Metrics'
- Find your needed object

Lab: "Lab Desktops" icon, i.e. our object

- Open a new tab/browser window and navigate to the following URL:
 - https://<your vrops server>/suite-api/api/resources?regex=.*<Your object>.*

Lab: https://172.16.0.19/suite-api/api/resources?regex=.*Lab%20Desktops.*

Note: You can use the complete name or just a fraction. With ".*" at the beginning/end we make sure that we have not missed something.

You should see a similar output dump to the following screenshot:



```

- <ops:resources>
- <ops:pageInfo totalCount="1">
  <ops:page>0</ops:page>
  <ops:pageSize>100</ops:pageSize>
</ops:pageInfo>
- <ops:links>
  <ops:link href="/suite-api/api/resources?regex=.*Lab%20Desktops.*&page=0&pageSize=100" rel="SELF" name="current"/>
  <ops:link href="/suite-api/api/resources?regex=.*Lab%20Desktops.*&page=0&pageSize=100" rel="RELATED" name="first"/>
  <ops:link href="/suite-api/api/resources?regex=.*Lab%20Desktops.*&page=0&pageSize=100" rel="RELATED" name="last"/>
</ops:links>
- <ops:resource creationTime="1425625787374" identifier="31522d45-63d4-4d8c-b97a-8bae419a9fd5">
  <ops:description>
  <ops:resourceKey>
    <ops:name>Lab Desktops</ops:name>
    <ops:adapterKindKey>
    <ops:resourceKindKey>Environment</ops:resourceKindKey>
  </ops:resourceIdentifiers>
  <ops:resourceKey>
  </ops:resourceStatusStates>
  - <ops:resourceStatusState>
    <ops:adapterInstanceId>ed601a44-293d-424a-a40d-0598dc9dc53f</ops:adapterInstanceId>
    <ops:resourceStatus>DATA_RECEIVING</ops:resourceStatus>
    <ops:resourceState>STARTED</ops:resourceState>
    <ops:statusMessage>
    </ops:resourceStatusState>
  </ops:resourceStatusStates>
  <ops:resourceHealth>ORANGE</ops:resourceHealth>
  <ops:resourceHealthValue>50.0</ops:resourceHealthValue>
  <ops:dtEnabled>true</ops:dtEnabled>
</ops:badges>

```

- Verify the name of the object is correct

Lab: 'Lab Desktops'.

- Copy the identifier ID, you will use this for the next step

5. Retrieving your metric description (API Call #2)

- Open a new tab/browser window and navigate to the following URL:
 - <https://<your vrops server>/suite-api/api/resources/<your object identifier ID>/statkeys>

Lab: <https://172.16.0.19/suite-api/api/resources/31522d45-63d4-4d8c-b97a-8bae419a9fd5/statkeys>

You should see a similar output dump to the following screenshot:

```

- <ops:stat-key>
  <ops:key>Compliance|Critical|totalCount</ops:key>
</ops:stat-key>
- <ops:stat-key>
  <ops:key>Health|Warning|totalCount</ops:key>
</ops:stat-key>
- <ops:stat-key>
  <ops:key>Anomaly|Immediate|totalCount</ops:key>
</ops:stat-key>
- <ops:stat-key>
  <ops:key>CapacityRemaining|Immediate|totalCount</ops:key>
</ops:stat-key>
- <ops:stat-key>
  <ops:key>TimeRemainingCapIQ|Immediate|totalCount</ops:key>
</ops:stat-key>
- <ops:stat-key>
  <ops:key>Workload|Normal|totalCount</ops:key>
</ops:stat-key>
- <ops:stat-key>
  <ops:key>Workload|Critical|totalCount</ops:key>
</ops:stat-key>
- <ops:stat-key>
  <ops:key>Stress|Normal|totalCount</ops:key>
</ops:stat-key>
- <ops:stat-key>
  <ops:key>Super Metric|sm_b896a92d-657f-4cb2-b550-8ed4ff1526b6
</ops:stat-key>
</ops:stat-key>
- <ops:stat-key>
  <ops:key>Health|populationAverage</ops:key>
</ops:stat-key>
</ops:stat-key>

```

Scroll until you find your Super Metric identifier ID. This is the name of the metric we are going to use for the interactive widget and the xml definitions.

Navigate to:

- Administration
- Manage Metric Config (two down from Super Metrics in the left panel)
- Expand the tree ReskndMetric

6. Modify your XML file

Modify your new XML file to reflect your Super Metric:

- Find the section that has the ResourceKind that pertains to your Super Metric.

Lab: *In our example we need 'ViewPool'*

- Copy an existing line and past it, example shown below:

```

<ResourceKind resourceKindKey="ViewPool">
  <Metric red="10" orange="40" yellow="75" unit="%" label="Pool Health" attrkey="badgehealth" />
  <Metric red="100" orange="90" yellow="80" unit="%" label="Pool Workload" attrkey="badge/workload" />
  <Metric red="95" orange="90" yellow="80" unit="%" label="Pool Capacity Used" attrkey="desktop_vms|pool_capacity_used" />
  <Metric red="10002" orange="10001" yellow="10000" unit="sessions" label="Pool Session Count" attrkey="remote_sessions|session_count" />
  <Metric red="10002" orange="10001" yellow="10000" unit="desktops" label="Pool Maximum Desktops" attrkey="desktop_vms|max_desktops" />
  <Metric red="300" orange="120" yellow="30" unit="sec" label="Pool Avg Session Logon Time" attrkey="remote_sessions|avg_logon_time" />
  <Metric red="300" orange="120" yellow="30" unit="sec" label="Pool Avg Session Reconnect Time" attrkey="remote_sessions|avg_reconnection_time" />
  <Metric red="600" orange="400" yellow="250" unit="ms" label="Pool Average PCoIP Latency" attrkey="remote_sessions|avg_pcoip_latency" />

```

Modify your new string to match your requirements for the super metric.

Lab:

- <Metric red="90"
- orange="85"
- yellow="75"
- unit="%"
- label="vdi_latency" (common name of your Super Metric goes here)
- attrkey="Super Metric|sm_b896a92d-657f-4cb2-b550-8ed4ff1526b6" /> (past your Super Metric identifier ID in this section.)

Select 'Save'

Verify your policy has the Super Metric setup in the overrides section.

7. Update your widget

Navigate to:

- Home
- Dashboard List drop-down
- Select your custom dashboard

Edit Object Metric Charts widget

- In the 'Metric Configuration' section
 - Use the drop-down and select your custom XML file from the list

Lab: 'lab_desktop'

- Select Save

8. Validate your XML changes

Navigate to:

- Home
- Dashboard List drop-down
- Select your custom dashboard

Lab: 'Lab Dashboard'

- In your View Sessions widget, select a VDI session

Lab: my session is vmbucket\usernameID

- In your Session Related Objects widget, select a desktop pool

Lab: admin_vm pool

Your Object Metric Charts widget should update with the new Super Metric. You may need to wait a few cycles before the data is updated.