Creating and Applying Super Metrics in a vSphere Environment

VCenter Operations Enterprise

TECHNICAL WHITE PAPER

BY GREGG PARSONS, SR. SYSTEMS ENGINEER, VMWARE

DECEMBER-30-2011

V. 1.0

Table of Contents

Background

When implementing vCenter Operations Enterprise for performance analytics in a vSphere environment, sometimes in addition to the entity-level metrics that are pulled from vCenter or automatically calculated by vCenter Operations Enterprise we might also be interested in some aggregated performance data. For example, we might want to know the average CPU usage of a group of virtual machines and understand the normal range of values for that group's average. Within vCenter Operations Enterprise we can create a super metric to calculate the average of the 'CPU Usage|Usage (%)' metric then attach that super metric to some container that represents the group of interest. Once the super metric is created and applied to the group, vCenter Operations Enterprise will then begin to calculate that average value whenever new CPU usage metrics are received from vCenter and will also treat that super metric just like any other metric within the analytics engine. This means that we will start to get dynamic thresholds calculated for the super metric values – something that can be very powerful in understanding the health of our environment.

Please read the section titled 'Creating and Editing Super Metrics' in the *VMware vCenter Operations Enterprise Installation and Administration Guide* before trying to implement super metrics yourself. That guide does a thorough job of explaining how to define and assign super metrics but lacks concrete examples of how they might be used in a vSphere environment. This document is intended to fill that hole. Included here are step by step instructions for some use cases that might be found in a typical vSphere environment. The use cases were chosen to provide examples of using various super metric functions and applying them to different levels within the vSphere hierarchy of virtual machines, hosts, clusters, resource pools, vCenter folders, etc.

Even though the use cases contained in this document are specific to metrics pulled into vCenter Operations Enterprise via the vCenter adapter, the concepts can be applied to any environment where vCenter Operations Enterprise is deployed and to metrics coming from any adapter.

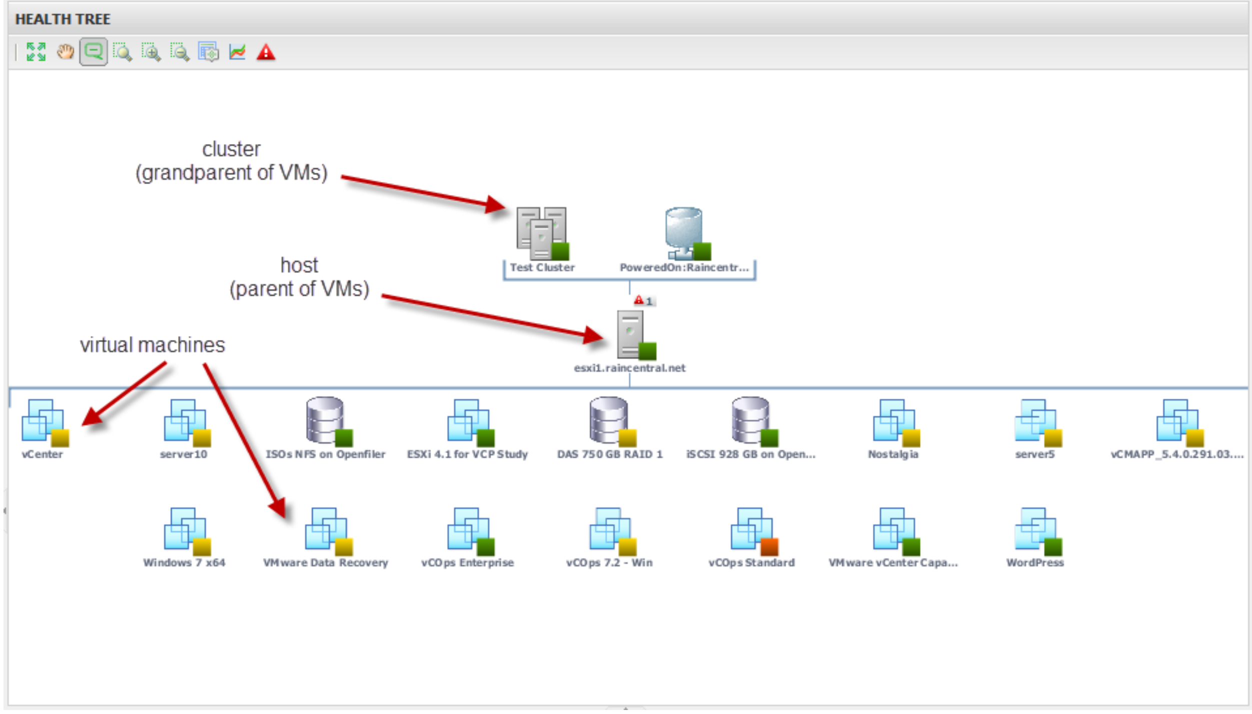
Environment

The use cases defined here were executed in an environment running vCenter Operations Enterprise version 1.0.1 with the vCenter adapter. For the reader to follow along with the exercises in this document, it is assumed that they have an environment where vCenter Operations Enterprise is collecting resource inventory and metrics from at least one vCenter instance.

Use Cases / Examples

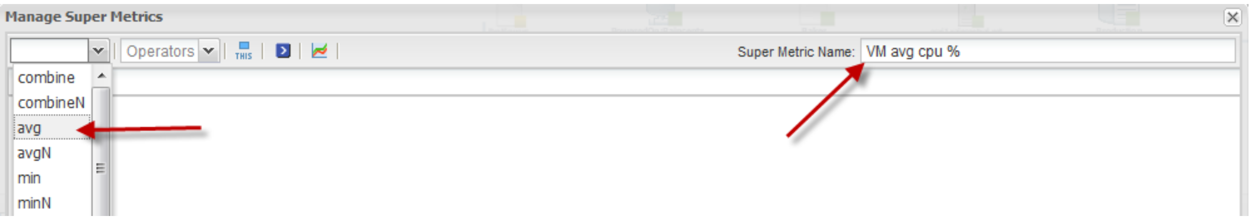
1. Average CPU usage of all VMs running on a host

Suppose we want to know the average CPU usage of all powered on virtual machines on a host. The first step is to understand the relationship between the resource whose metric we want to average (the VM) and the resource where we want to apply the super metric (the host). For this example, it is clear that the VM is a direct child of the host but what if we were instead interested in finding the average CPU usage of all VMs in a cluster? Then there would be a grandparent/grandchild relationship between the two resources. We will address this scenario in the next use case but for super metric looping functions (like average) we need to know how many levels apart the two resources are. The best way to find the relationship between two resources is by using the Health Tree in the Resource Detail screen. We may have to navigate up and down the tree and keep track of how many levels separate the resources of interest.



Once we know that we will be applying the super metric one level above (in this case) the resource with the metric of interest, navigate to the super metric editor to create the super metric: ENVIRONMENT > ADVANCED > SUPER METRICS > SUPER METRIC EDITOR… and click the icon to add a new super metric.

In the Manage Super Metrics window, give the super metric a descriptive name – something like “VM avg cpu %” then from the Functions drop down select ‘avg’ to start building our super metric function. avg is the right function to use in this case (versus avgN) since we know that we are applying the super metric one level above the resource with the metric of interest.



Manage Super Metrics

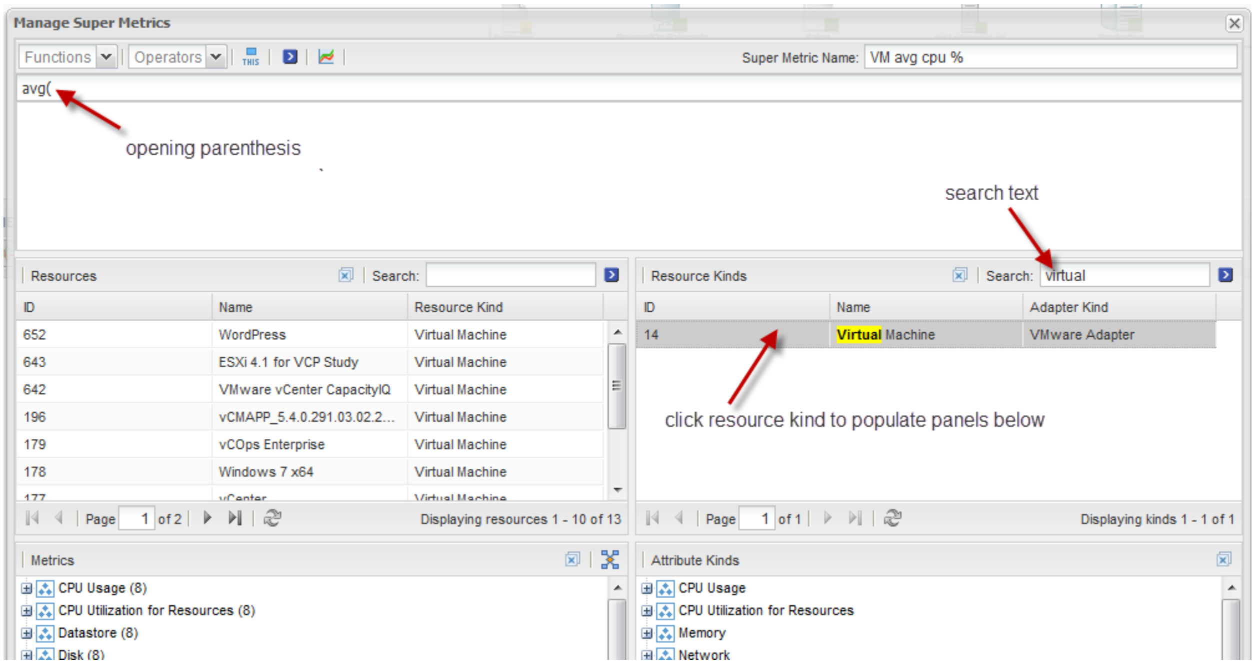
Operators

Super Metric Name: VM avg cpu %

- combine
- combineN
- avg
- avgN
- min
- minN

Type an opening parenthesis on the formula line after ‘avg’.

The section below the formula is divided into a Resources panel (left) and a Resource Kind panel (right). Only use the Resources panel if you want to use a specific resource in your formula. In this case we want to average the CPU usage percentage across all VMs in the host so we will find the Virtual Machine entry in the Resource Kinds panel and click the line to select it and populate the panes below. Note that there are likely several pages of resource kinds so it’s easiest to enter a partial or full text string in the Search box to find the resource kind of you are looking for.



Manage Super Metrics

Functions | Operators

Super Metric Name: VM avg cpu %

avg(

opening parenthesis

search text

click resource kind to populate panels below

Resources — Search:

ID	Name	Resource Kind
652	WordPress	Virtual Machine
643	ESXi 4.1 for VCP Study	Virtual Machine
642	VMware vCenter CapacityIQ	Virtual Machine
196	vCMAPP_5.4.0.291.03.02.2...	Virtual Machine
179	vCOps Enterprise	Virtual Machine
178	Windows 7 x64	Virtual Machine
177	vCenter	Virtual Machine

Page 1 of 2 — Displaying resources 1 - 10 of 13

Resource Kinds — Search: virtual

ID	Name	Adapter Kind
14	Virtual Machine	VMware Adapter

Page 1 of 1 — Displaying kinds 1 - 1 of 1

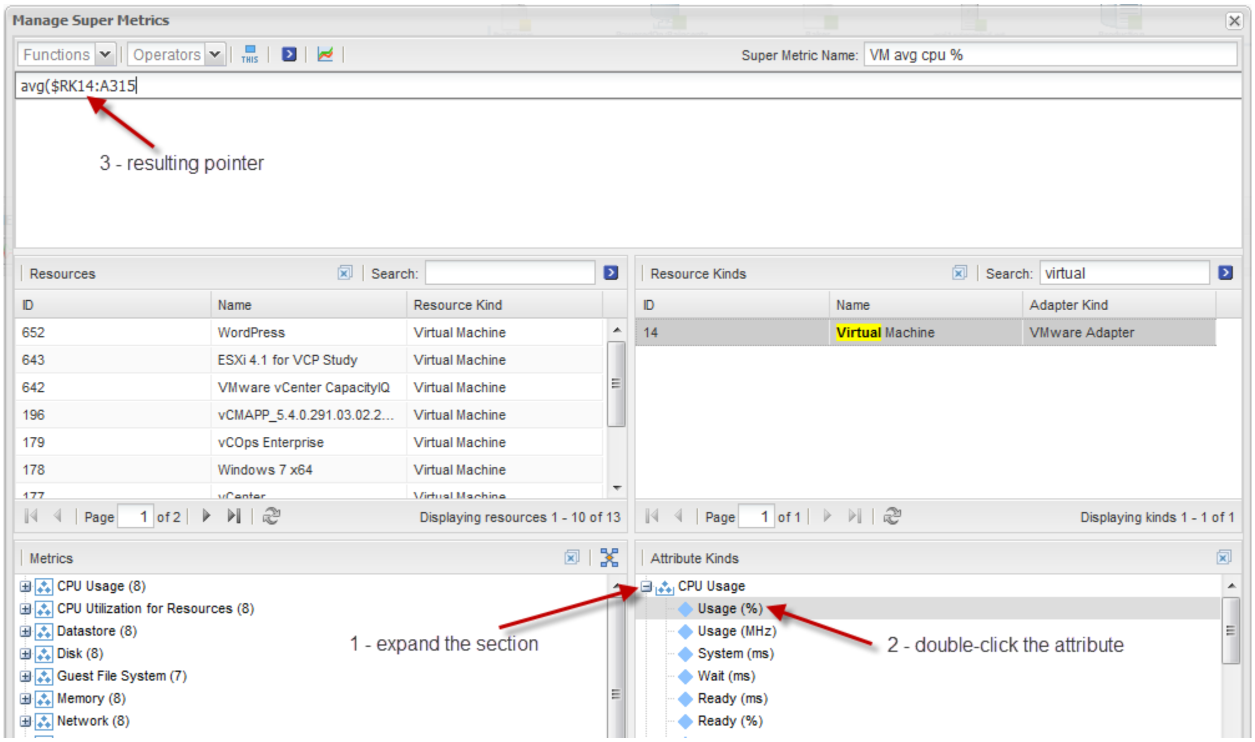
Metrics

- CPU Usage (8)
- CPU Utilization for Resources (8)
- Datastore (8)
- Disk (8)

Attribute Kinds

- CPU Usage
- CPU Utilization for Resources
- Memory
- Network

The bottom section of this window is divided into a Metrics panel (left) and an Attributes Kind panel (right). Since we are using a looping function as opposed to a single function (see the installation and administration guide for definitions), we will select the metric of interest from the Attributes Kind pane. Expand the CPU Usage section and double click on Usage (%) to add it to the formula. You will notice that in the formula line the selected resource kind/attribute combination is represented by tags from the vCenter Operations Enterprise database (note that your tags will likely be different than those shown here).



Super Metric Name: VM avg cpu %

avg($RK14:A315

Resources — Search:

ID	Name	Resource Kind
652	WordPress	Virtual Machine
643	ESXi 4.1 for VCP Study	Virtual Machine
642	VMware vCenter CapacityIQ	Virtual Machine
196	vCMAPP_5.4.0.291.03.02.2...	Virtual Machine
179	vCOps Enterprise	Virtual Machine
178	Windows 7 x64	Virtual Machine
177	vCenter	Virtual Machine

Displaying resources 1 - 10 of 13

Resource Kinds — Search: virtual

ID	Name	Adapter Kind
14	Virtual Machine	VMware Adapter

Displaying kinds 1 - 1 of 1

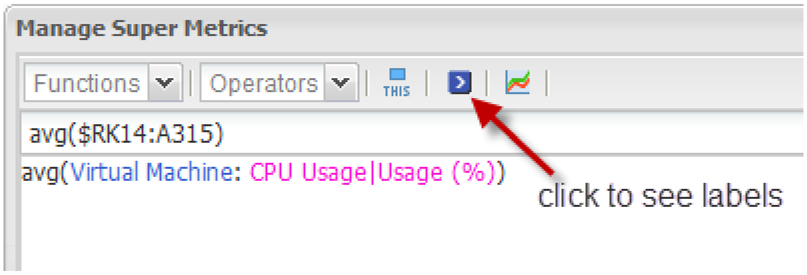
Metrics

- CPU Usage (8)
- CPU Utilization for Resources (8)
- Datastore (8)
- Disk (8)
- Guest File System (7)
- Memory (8)
- Network (8)

Attribute Kinds

- CPU Usage
  - Usage (%)
  - Usage (MHz)
  - System (ms)
  - Wait (ms)
  - Ready (ms)
  - Ready (%)

Type a close parenthesis to finish the formula then click the blue icon above the formula line (mouse over to see the tool tip “Show Formula Description”). This will show the resource kind and attribute labels.

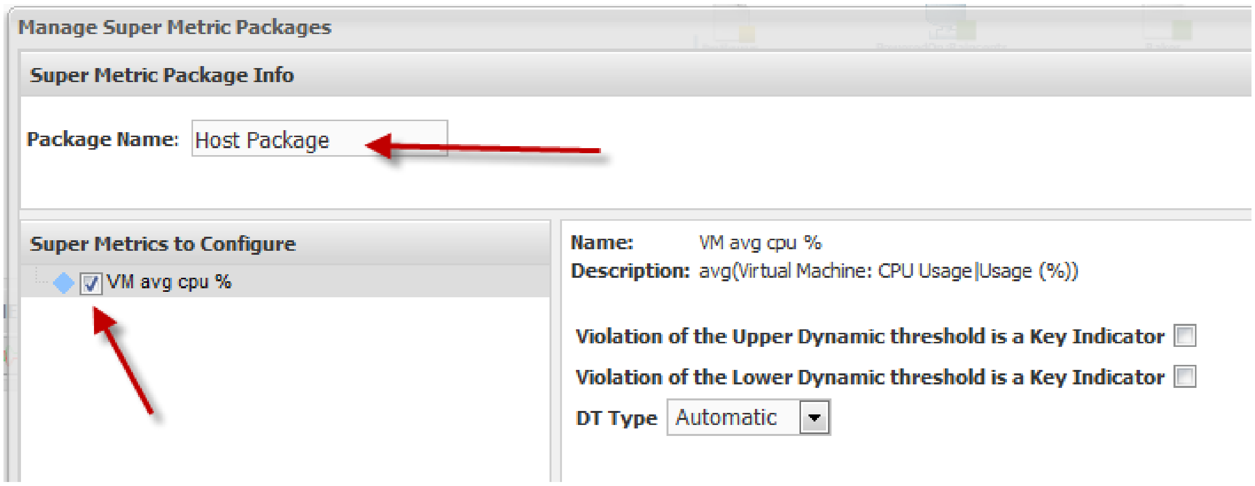


avg($RK14:A315)

avg(Virtual Machine: CPU Usage|Usage (%))

Verify that the labels are correct then click the OK button to save the super metric. Click the OK button to close the Manager Super Metrics window.

The next step is to create a super metric package that will be attached to the host or hosts for the calculation of VM CPU usage averages: ENVIRONMENT > ADVANCED > SUPER METRICS > SUPER METRIC PACKAGES… and click the icon to add a new package. Since we will be attaching this super metric package to hosts, use a descriptive but generic package name like “Host package”. If we create other super metrics later that we want to also apply at the host level, we can just add them to this package. Check the box next to the new super metric we just created to add it to the package. Optionally, we can check the boxes in the right panel to create a KPI for each super metric that we add to a package and can even set classic thresholds just like any other metric in vCenter Operations Enterprise.



Manage Super Metric Packages

Super Metric Package Info

Package Name: Host Package

Super Metrics to Configure

☑ VM avg cpu %

Name: VM avg cpu %

Description: avg(Virtual Machine: CPU Usage|Usage (%))

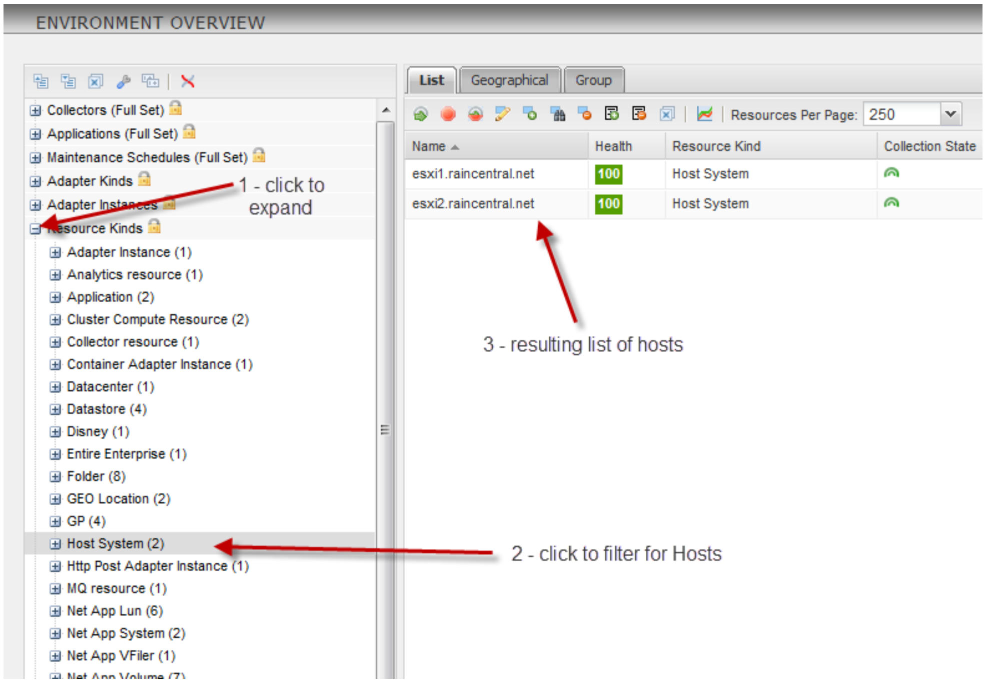
Violation of the Upper Dynamic threshold is a Key Indicator □

Violation of the Lower Dynamic threshold is a Key Indicator □

DT Type: Automatic

Click the OK button to save the package. Click the OK button to close the Manager Super Metrics Packages window.

Next, we need to attach the newly created super metric package to one or more hosts. In ENVIRONMENT > ENVIRONMENT OVERVIEW, expand the Resource Kinds tag in the left panel and click on the Host System tag in that section. Assuming that we don't have any other tags selected and that our search box is empty, we will see a list of all of the host resources in the right pane.



ENVIRONMENT OVERVIEW

- Collectors (Full Set)
- Applications (Full Set)
- Maintenance Schedules (Full Set)
- Adapter Kinds
- Adapter Instances
- Resource Kinds
  - Adapter Instance (1)
  - Analytics resource (1)
  - Application (2)
  - Cluster Compute Resource (2)
  - Collector resource (1)
  - Container Adapter Instance (1)
  - Datacenter (1)
  - Datastore (4)
  - Disney (1)
  - Entire Enterprise (1)
  - Folder (8)
  - GEO Location (2)
  - GP (4)
  - Host System (2)
  - Http Post Adapter Instance (1)
  - MQ resource (1)
  - Net App Lun (6)
  - Net App System (2)
  - Net App VFiler (1)
  - Net App Volume (7)

Name	Health	Resource Kind	Collection State
esxi1.raincentral.net	100	Host System	
esxi2.raincentral.net	100	Host System	

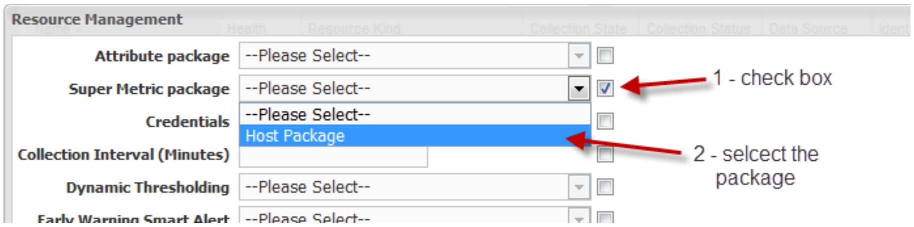
1 - click to expand

2 - click to filter for Hosts

3 - resulting list of hosts

In the right pane, select all of the hosts that we want to attach the super metric package to (hosts where we want to calculate the average CPU usage for the VMs) then click the Edit Resource icon (pencil) above the list of hosts.

In the Resource Management window that opens, check the box on the Super Metric Package line then select the package that we just created from the drop-down list then click the OK button to save wer selection.



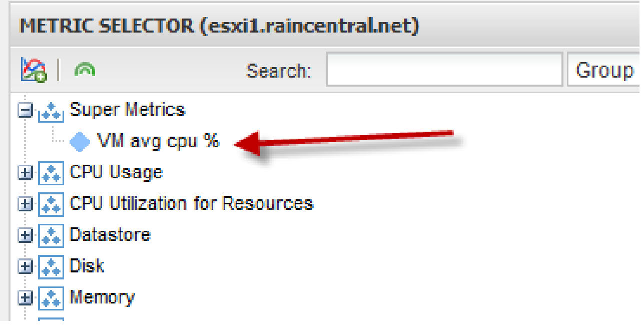
Resource Management

Attribute package	--Please Select--	□
Super Metric package	--Please Select-- --Please Select-- Host Package	☑
Credentials		□
Collection Interval (Minutes)		□
Dynamic Thresholding	--Please Select--	□
Early Warning Smart Alert	--Please Select--	□

1 - check box

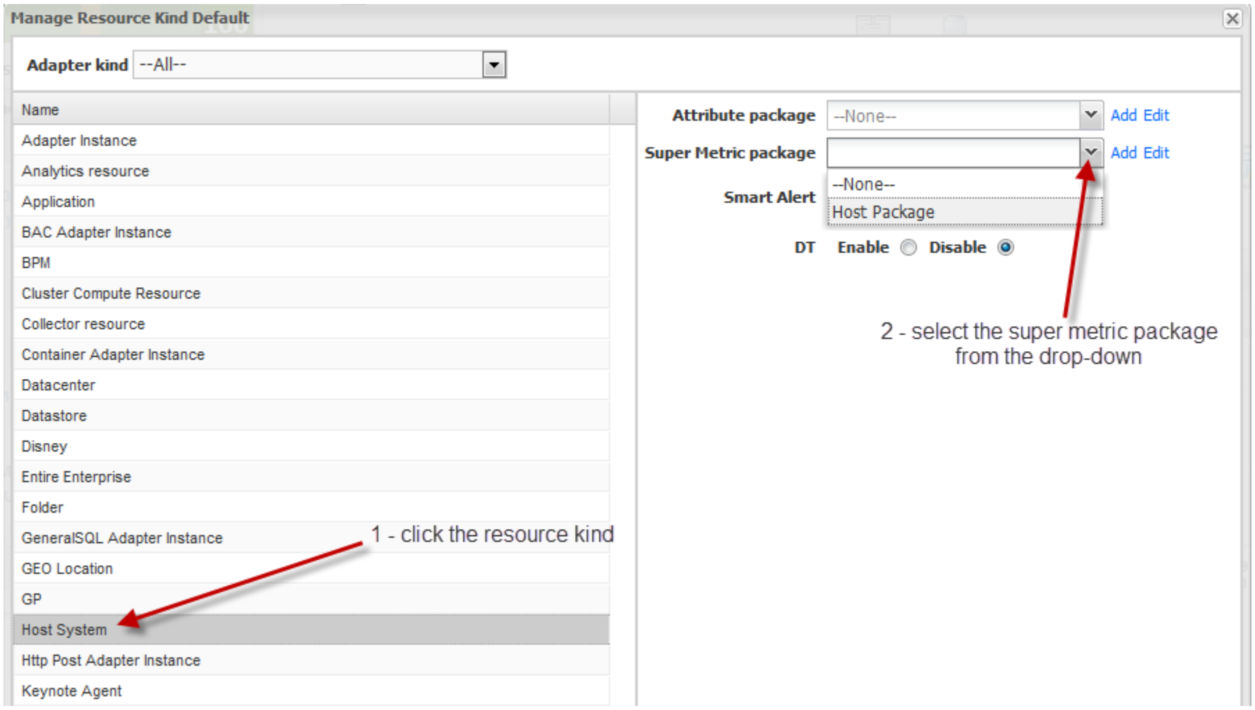
2 - selcect the package

We have completed the process. The hosts that we just modified (by adding the super metric package) will change to an unknown health status (question mark in blue box) while the changes are applied. Once a new poll cycle completes, we can browse to the All Metrics Resource Detail view for any of the hosts and look at the metric selector pane. We should see a section at the top of the metrics titled Super Metrics. Expand the section and we should see our super metric.



METRIC SELECTOR (esxi1.raincentral.net)

We can also have any new hosts that are discovered from vCenter in the future via the adapter automatically have the super metric package attached to them. To do so, navigate to ENVIRONMENT > CONFIGURATION > RESOURCE KIND DEFAULTS… In the left panel select the resource kind (Host System) then in the right panel select the package from the Super Metric package drop-down. Click OK to save the selection.



Manage Resource Kind Default

Adapter kind: --All--

Name
Adapter Instance
Analytics resource
Application
BAC Adapter Instance
BPM
Cluster Compute Resource
Collector resource
Container Adapter Instance
Datacenter
Datastore
Disney
Entire Enterprise
Folder
GeneralSQL Adapter Instance
GEO Location
GP
Host System
Http Post Adapter Instance
Keynote Agent

Attribute package: --None--

Super Metric package: (--None-- / Host Package)

Smart Alert

DT: Enable ○ Disable ●

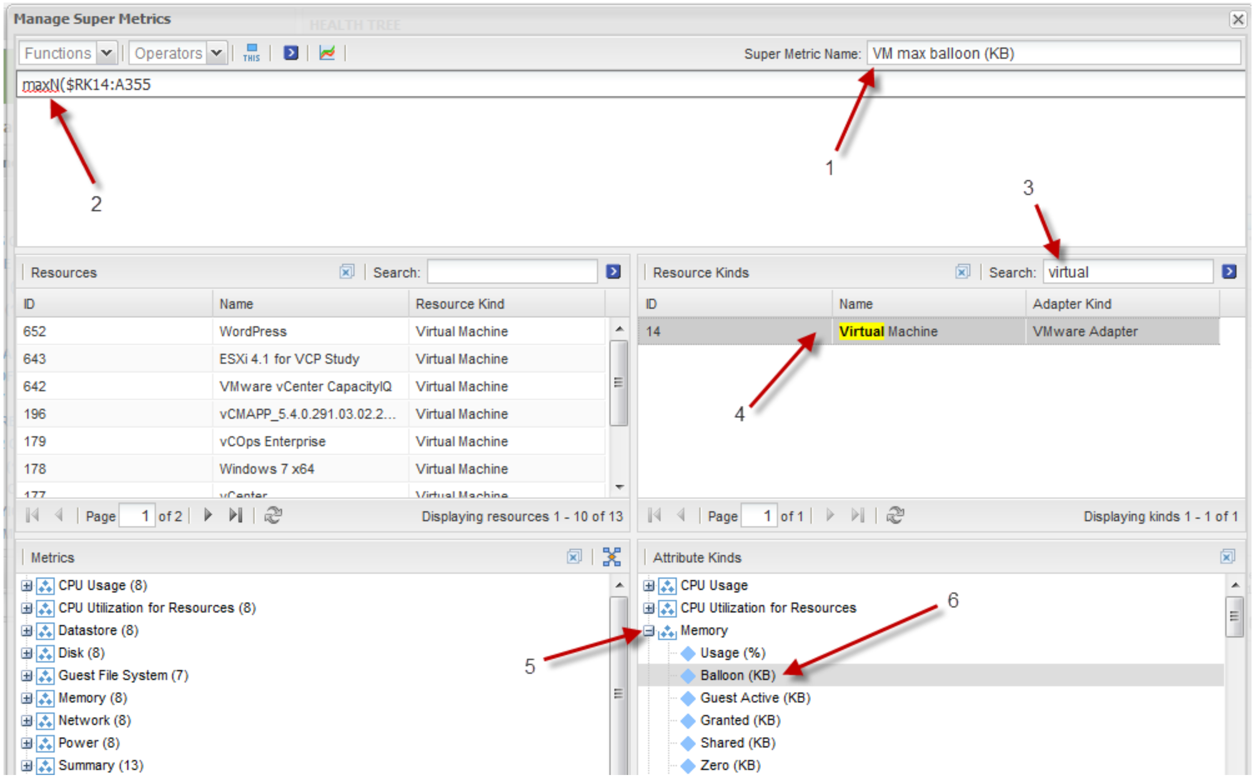
1 - click the resource kind

2 - select the super metric package from the drop-down

2. Maximum amount of memory ballooning across the VMs in a cluster

Suppose for a cluster we want to know the largest amount of balloon memory that any VM in that cluster is contributing. As with the previous example, we need to understand the relationship between the resource whose metric we want to find the maximum of (the VM) and the resource where we want to apply the super metric (the cluster). In this case, the relationship is grandparent/grandchild or two levels apart.

With that knowledge, create a new super metric as in the previous example but this time, call it ‘VM max balloon (KB)’ and select the maxN function. Again, from the installation and configuration guide, we will find that for the xxxN functions, instead of working on just the immediate children it looks down (or up) the number of levels specified in the formula. Again, select the Virtual Machine resource kind but this time double click on Memory->Balloon (KB) in the Attribute Kinds panel.



Super Metric Name: VM max balloon (KB)

```
maxN($RK14:A355
```

Resources

ID	Name	Resource Kind
652	WordPress	Virtual Machine
643	ESXi 4.1 for VCP Study	Virtual Machine
642	VMware vCenter CapacityIQ	Virtual Machine
196	vCMAPP_5.4.0.291.03.02.2...	Virtual Machine
179	vCOps Enterprise	Virtual Machine
178	Windows 7 x64	Virtual Machine
177	vCenter	Virtual Machine

Displaying resources 1 - 10 of 13

Resource Kinds — Search: virtual

ID	Name	Adapter Kind
14	Virtual Machine	VMware Adapter

Displaying kinds 1 - 1 of 1

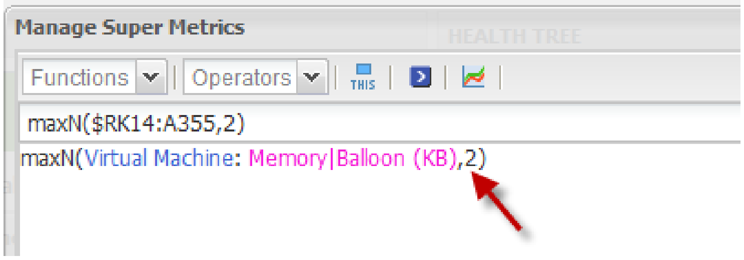
Metrics

- CPU Usage (8)
- CPU Utilization for Resources (8)
- Datastore (8)
- Disk (8)
- Guest File System (7)
- Memory (8)
- Network (8)
- Power (8)
- Summary (13)

Attribute Kinds

- CPU Usage
- CPU Utilization for Resources
- Memory
  - Usage (%)
  - Balloon (KB)
  - Guest Active (KB)
  - Granted (KB)
  - Shared (KB)
  - Zero (KB)

The maxN function needs to be told how many levels up (or down) to look in the relationship tree from where the super metric package is attached to where the metric of interest exists. In this case, type a comma followed by the number 2 and a close parenthesis to finish the function. This ‘2’ tells the function to look down one and two levels for the metric. Click the Show Formula Description button to see the resulting function definition with labels.



maxN($RK14:A355,2)

maxN(Virtual Machine: Memory|Balloon (KB),2)

Save this super metric. Create a new super metric package called ‘Cluster Package’ and add it to your Cluster Compute Resources. Also add the package to the Cluster Compute Resource in the resource kind defaults so any new clusters will inherit the super metric package.

3. I/O read rate as a percentage of I/O write rate for all virtual machines

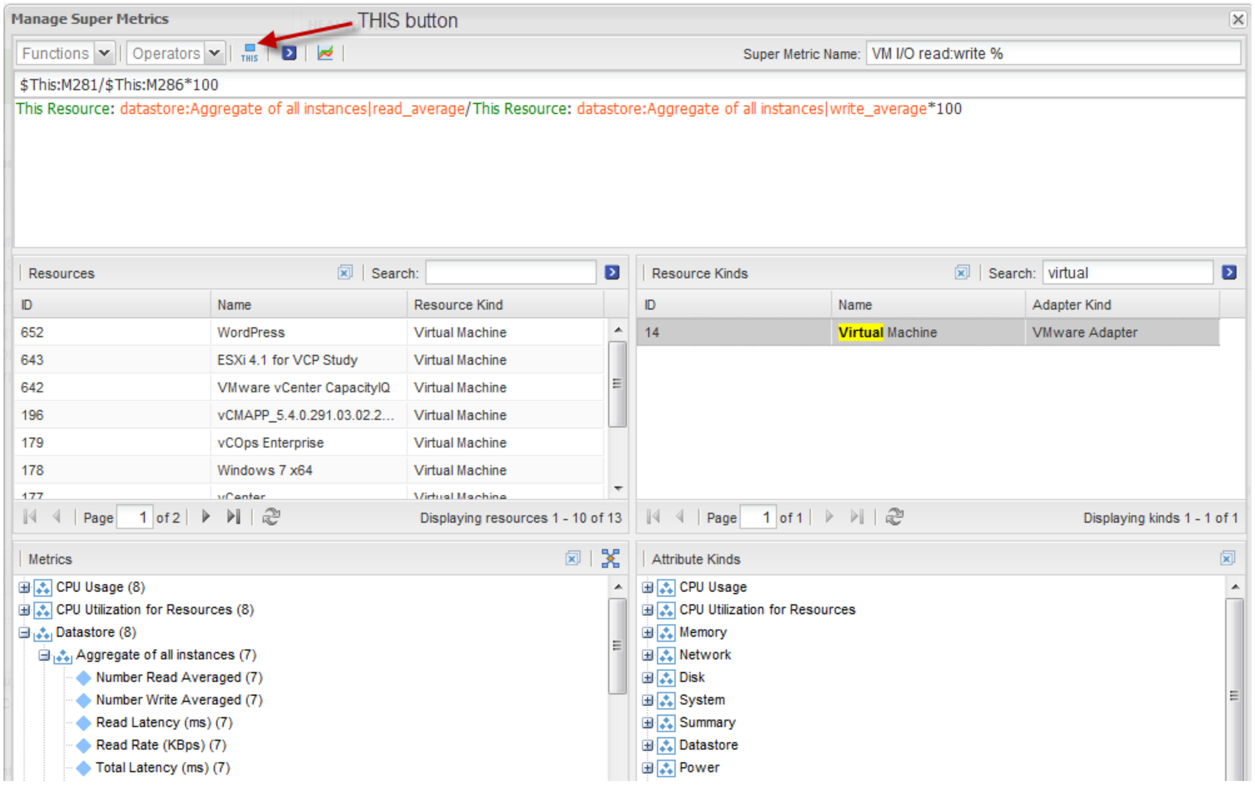
Suppose we want to know the aggregated datastore read rate as a percentage of write rate for all VMs. In this case, we will be applying the super metric to the actual resource where we are deriving the metrics from.

Create a new super metric as in the previous example but this time, call it ‘VM I/O read:write %’. Instead of a function, we will be using operators in the formula. Also, since we want to calculate a value for each VM based on metrics for that VM, we will use the ‘$This’ operator.

Again, select the Virtual Machine resource kind but this time since we are not using a looping function we will select from the Metrics panel below instead of the Attribute Kinds panel. Note that if you don’t see the VM metrics in the Metrics (left) panel, click the icon in the top-right corner of that panel (Show Common Metrics) to toggle the view. Expand Datastore then expand ‘Aggregate of all instances’. Before we double click ‘Read Rate (KBps), click once on the ‘THIS’ icon above the formula bar – this will prepend ‘$This:’ to the metric tag to evaluate the selected metric for each VM that the super metric is attached to later.

Either select ‘/’ from the Operators drop-down or just type the character on the formula line. Next , click again on the ‘THIS’ icon then double-click ‘Write Rate (KBps) in the Metrics panel.Finally, in the formula line type ‘*100’ to convert the decimal value to a percentage.

Check your formula with labels by clicking the ‘Show Formula Description’ button.



Super Metric Name: VM I/O read:write %

$This:M281/$This:M286*100

This Resource: datastore:Aggregate of all instances|read_average/This Resource: datastore:Aggregate of all instances|write_average*100

Resources — Search:

ID	Name	Resource Kind
652	WordPress	Virtual Machine
643	ESXi 4.1 for VCP Study	Virtual Machine
642	VMware vCenter CapacityIQ	Virtual Machine
196	vCMAPP_5.4.0.291.03.02.2...	Virtual Machine
179	vCOps Enterprise	Virtual Machine
178	Windows 7 x64	Virtual Machine
177	vCenter	Virtual Machine

Page 1 of 2 — Displaying resources 1 - 10 of 13

Resource Kinds — Search: virtual

ID	Name	Adapter Kind
14	Virtual Machine	VMware Adapter

Page 1 of 1 — Displaying kinds 1 - 1 of 1

Metrics

- CPU Usage (8)
- CPU Utilization for Resources (8)
- Datastore (8)
  - Aggregate of all instances (7)
    - Number Read Averaged (7)
    - Number Write Averaged (7)
    - Read Latency (ms) (7)
    - Read Rate (KBps) (7)
    - Total Latency (ms) (7)

Attribute Kinds

- CPU Usage
- CPU Utilization for Resources
- Memory
- Network
- Disk
- System
- Summary
- Datastore
- Power

To finish the exercise, create a new super metric package with this super metric and attach it to virtual machines.

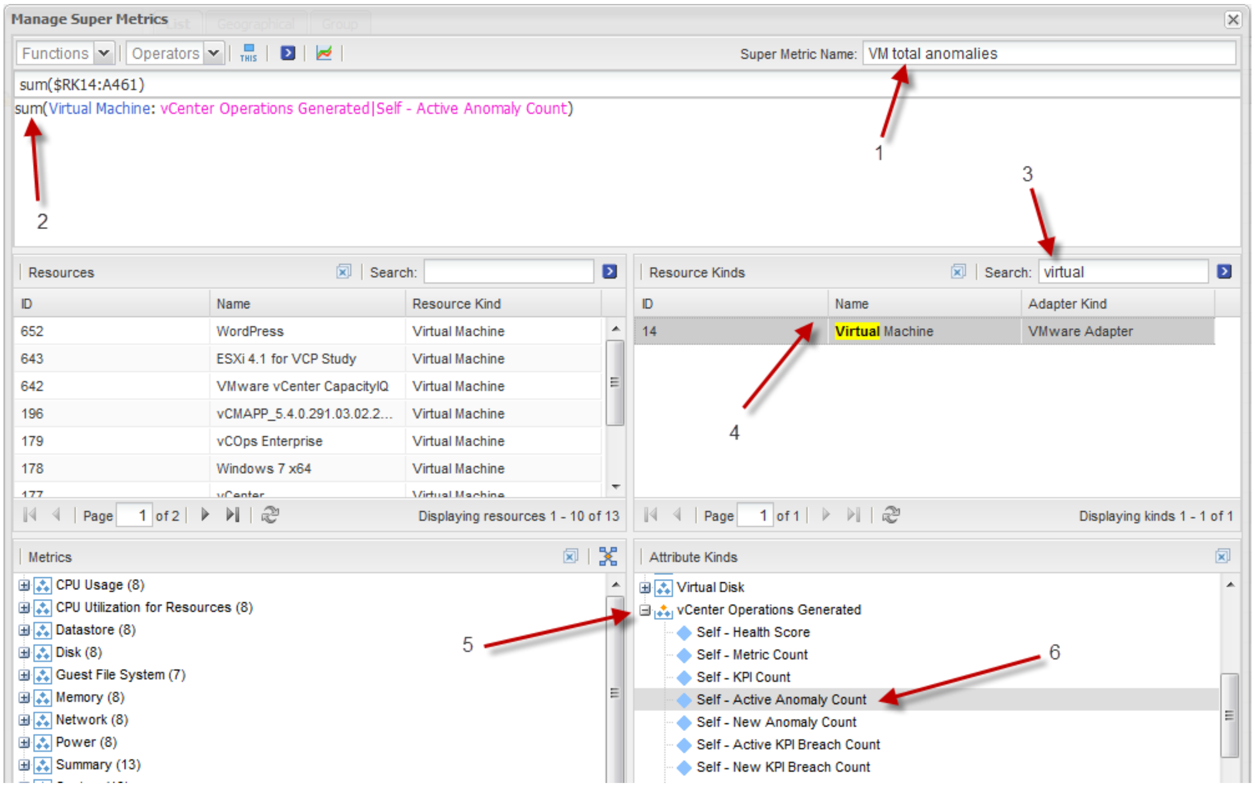
4. Application of super metrics to any arbitrary group of resources

The previous examples can be extended to use the various functions available in the super metric editor. They can also be applied at any grouping construct that has been created in vCenter Operations Enterprise automatically by the adapter. For example, in addition to the host, cluster and VM resources that we have already looked at, we could also apply super metric packages to resource pool groupings and even vCenter folder groupings.

Of course, one of the most powerful places to use super metrics within vCenter Operations Enterprise is in Applications and Application Tiers. If you have defined applications then you can start to get an understanding of health from a service or business-level perspective. If, for example, you have an n-tier application definition in place and one of the tiers contains all of the web servers for that particular application then you can apply super metrics to that tier that measure something specific you want to know about the health of the web tier for the application. This becomes much more powerful if you bring in metrics from other monitoring tools using other vCenter Operations Enterprise adapters and add those to the application definitions.

What if we want to calculate some aggregate value across some number of virtual machines (or other resources) that are not already grouped by host, cluster, folder, application tier, etc. using super metrics? vCenter Operations Enterprise includes the very powerful ability to create our own custom tags, add resources to those tags and do many things with the tagged group – including applying super metrics.

Suppose we want to know the total number of active anomalies (metric values outside of their dynamic range) for a specific group of VMs. In this case, create the super metric as in the previous example but this time call it ‘VM total anomalies’. Use the ‘sum’ function, select a Resource Kind of Virtual Machine and an Attribute Kind of ‘vCenter Operations Generated|Self - Active Anomaly Count ‘.



Super Metric Name: VM total anomalies

```
sum($RK14:A461)
sum(Virtual Machine: vCenter Operations Generated|Self - Active Anomaly Count)
```

Resources — Search:

ID	Name	Resource Kind
652	WordPress	Virtual Machine
643	ESXi 4.1 for VCP Study	Virtual Machine
642	VMware vCenter CapacityIQ	Virtual Machine
196	vCMAPP_5.4.0.291.03.02.2...	Virtual Machine
179	vCOps Enterprise	Virtual Machine
178	Windows 7 x64	Virtual Machine
177	vCenter	Virtual Machine

Page 1 of 2 — Displaying resources 1 - 10 of 13

Resource Kinds — Search: virtual

ID	Name	Adapter Kind
14	Virtual Machine	VMware Adapter

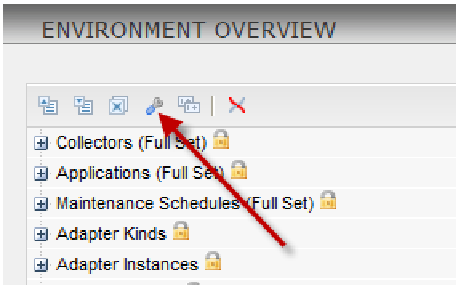
Page 1 of 1 — Displaying kinds 1 - 1 of 1

Metrics
- CPU Usage (8)
- CPU Utilization for Resources (8)
- Datastore (8)
- Disk (8)
- Guest File System (7)
- Memory (8)
- Network (8)
- Power (8)
- Summary (13)

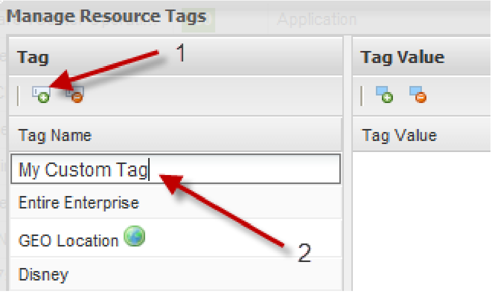
Attribute Kinds
- Virtual Disk
- vCenter Operations Generated
  - Self - Health Score
  - Self - Metric Count
  - Self - KPI Count
  - Self - Active Anomaly Count
  - Self - New Anomaly Count
  - Self - Active KPI Breach Count
  - Self - New KPI Breach Count

Save the super metric and create a super metric package called ‘Super Duper Custom Package’.

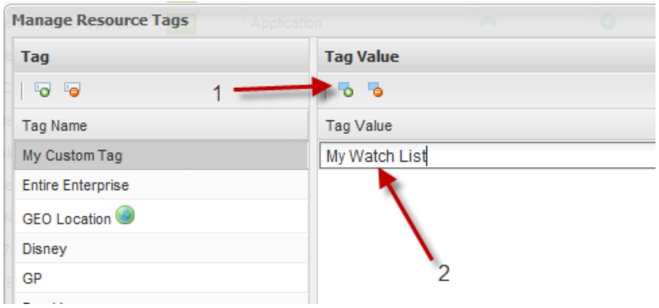
Now we are ready to create the custom group of virtual machines. From ENVIRONMENT > ENVIRONMENT OVERVIEW in the left pane, click the wrench icon (Manage Tags).



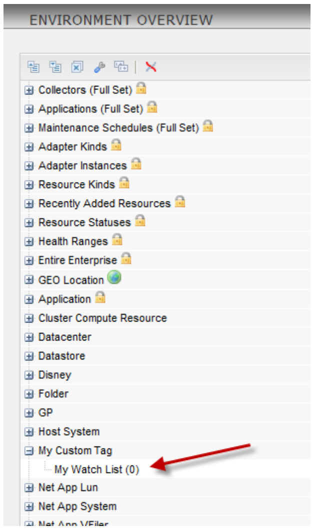
This will open the Manage Resource Tags window. Click the icon in the left panel to ‘Add Tag’, type ‘My Custom Tag’ in the box and press the Enter key. (Note that this is a finicky window within the application and you may have to exit the window and start over if you have problems.)



Highlight your new tag and add at least one new Tag Value in the right panel – call it ‘My Watch List’.

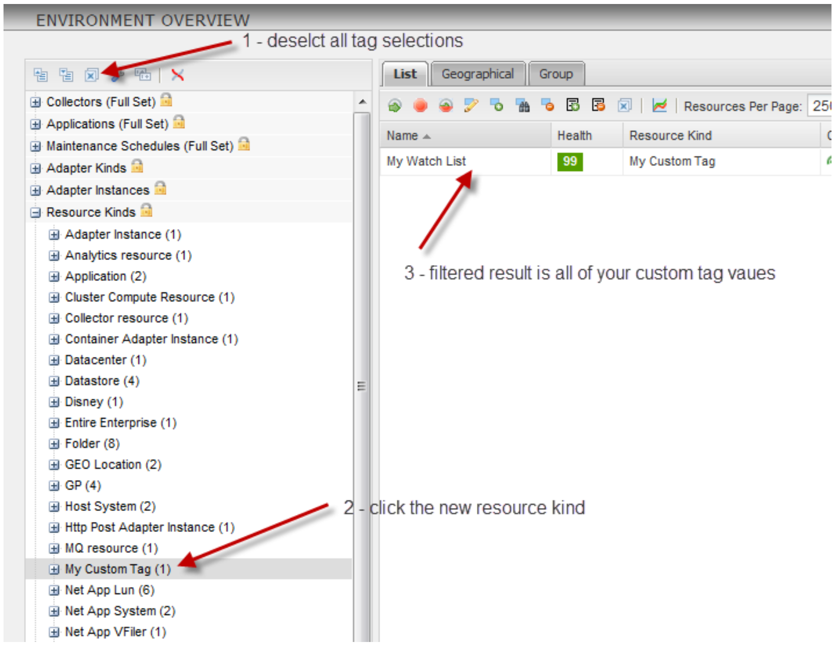


Click OK to save your changes. Now in the left panel of the Environment Overview you should see your new tag. If you expand it you will see your new Tag Value.



To add resources (in this case VMs), select them in the right panel (one at a time or multi-select) then drag the resource(s) over and drop them on the tag value ‘My Watch List’. You should see the counter for that tag value increment to the number of resources that you dropped on it.

Now if you click Deselect All, expand the Resource Kinds group on the left then click on the ‘My Custom Tag’ resource kind, in the right panel you will see all of the tag values you created.



Name	Health	Resource Kind
My Watch List	99	My Custom Tag

1 - deselct all tag selections

2 - click the new resource kind

3 - filtered result is all of your custom tag vaues

You can select the resource ‘My Watch List’, edit it and attach the super metric package ‘Super Duper Custom Package’ that you created earlier in this example to get the active anomaly count from all of the severs in your watch list.

Summary

Super metrics are a very powerful construct within vCenter Operations Enterprise but can sometimes be challenging to implement in order to get the desired results. Hopefully the use case examples shown in this document have helped the reader gain a better understanding of how to create and apply super metrics.