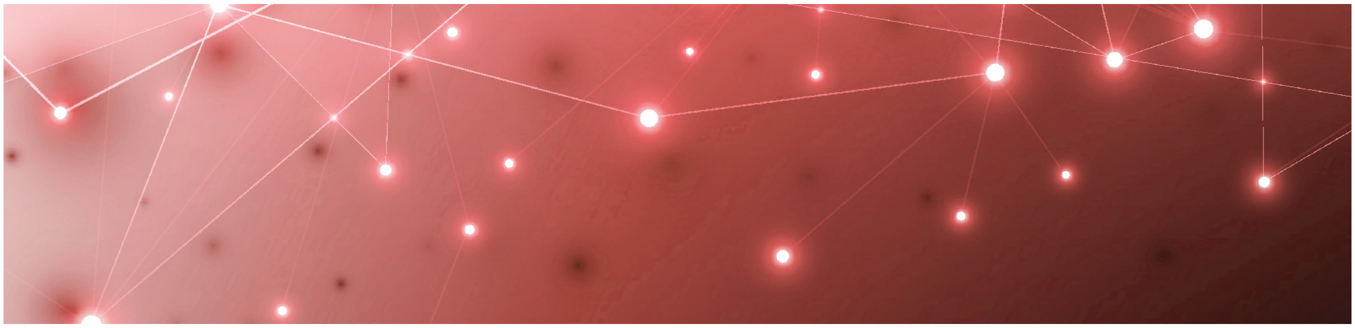


Savision is a subsidiary of Martello Technologies

Savision Live Maps

User Guide

Getting Started with Summary Dashboards

RELEASE 10.6

DOCUMENT DATE: AUGUST 19, 2020

NOTICE

The information contained in this document is believed to be accurate in all respects but is not warranted by Martello Technologies Corporation. The information is subject to change without notice and should not be construed in any way as a commitment by Martello Technologies or any of its affiliates or subsidiaries. Martello Technologies and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes.

Trademarks

MarWatch™, Savision, Martello Technologies, and the Martello Technologies logo are trademarks of Martello Technologies Corporation.

Windows and Microsoft are trademarks of Microsoft Corporation.

Other product names mentioned in this document may be trademarks of their respective companies and are hereby acknowledged.

User Guide
Release 10.6 - August 19, 2020

Contents

CHAPTER 1

CHAPTER 2

CHAPTER 3

CHAPTER 4

CHAPTER 5

CHAPTER 6

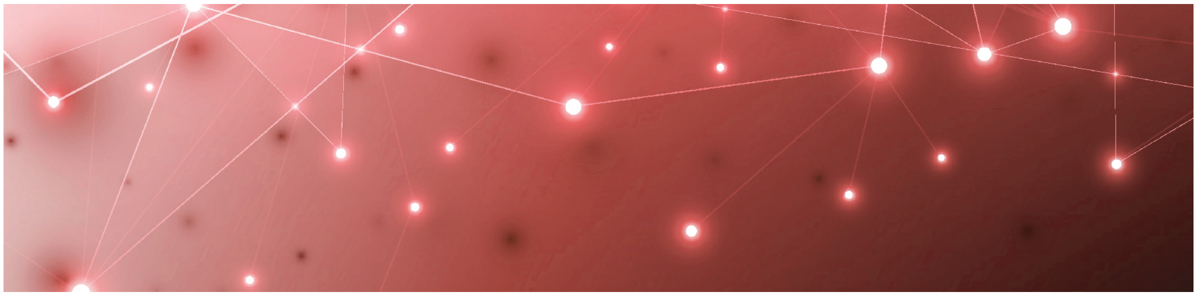
CHAPTER 7

CHAPTER 8

CHAPTER 9

CHAPTER 10

CHAPTER 11



CHAPTER 1

Introduction

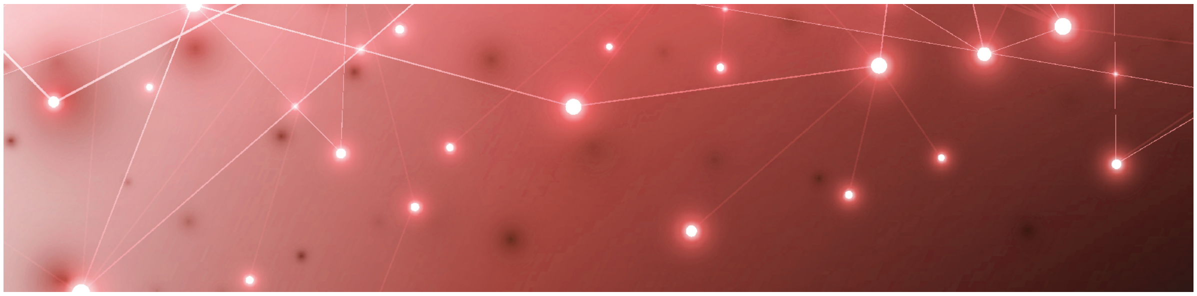
Purpose and Intended Audience

This guide provides information about how to get started with Summary Dashboards in Savision Live Maps. It provides an overview of all the available widgets, as well as examples of how to configure some of the most commonly used widgets. It also provides information about the dashboard templates and how to install and customize them.

This guide is intended for use by administrators and operators.

Revision History

Document Date	Description
August 19, 2020	Savision Live Maps Release 10.6



CHAPTER 2

About Live Maps

Savision Live Maps helps you implement business service management for your Microsoft System Center Operations Manager (SCOM) environment.

Savision Live Maps structures and visualizes IT components that are monitored in SCOM 2012 R2, 2016, 180X, and 2019. Its visualization features allows stakeholders in your organization to understand the impact and root cause of service outages. It offers out-of-the-box dashboards for a range of users in your organization, including:

- C-level management
- Service owners
- IT management
- Helpdesk operators
- Application engineers
- Infrastructure engineers

CHAPTER 3

About Summary Dashboards

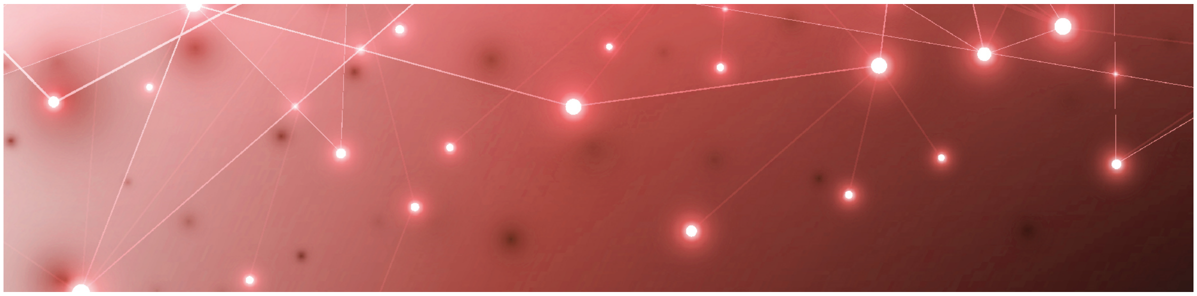
Summary Dashboards provide an efficient way to get started with Savision Live Maps. You can build a Summary Dashboard from the collection of widgets provided in the Savision Live Maps Portal. The widgets allow you to visualize components and their associated data collected from your Microsoft System Center Operations Manager (SCOM) environment.

You can create Summary Dashboards to support the various user roles in your organization. These Summary Dashboards allow users to view and interact with the SCOM data that is most meaningful to their role. Because the SCOM data is permission-based, a given user can only see content they are permitted to see in SCOM.

You can also add non-SCOM related content to a Summary Dashboard using the HTML, Webpage, PowerShell, and SQL widgets. For example, you can use these widgets to view status information from Systems Center Configuration Manager (SCCM) or Systems Center Service Manager.

To help you get started quickly, Savision Live Maps provides templates for Summary Dashboards. These dashboards are pre-configured with scripts that you can modify for your environment.

This guide provides an overview of all the available widgets, as well as examples of how to configure some of the most commonly used widgets. It also provides information about the dashboard templates and how to install and customize them.



CHAPTER 4

Widgets

Summary Dashboards are a way of visualizing your data. They give you the flexibility to choose the data that you want to see, and customize how it displays. Savision Live Maps provides widgets that you can use to build your Summary Dashboards. You can add a maximum of 12 widgets to a dashboard. The following table lists the available widgets and provides links to examples of how to configure the most commonly used widgets.

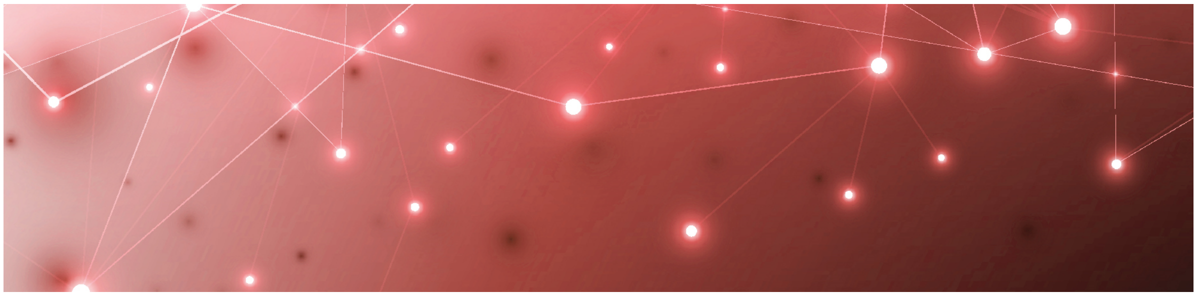
Table 1: Summary Dashboard Widgets

Widget	Description
Affected Services	Displays any of the Live Maps Services that contain the selected object. You can view the services in a list or a heat map, and you can sort alphabetically or by the health of the service.
Alert Listing	Displays a list of individual alerts. You can view the alerts based on the class of objects, the group, the alert state, and the alert resolution. For more information and examples, see "Alert Listing Widget" on page 11.
Alert Summary	Displays a list of objects and the total number of alerts for each object. You can view the alerts based on the class of objects, the group, the alert state, and the alert resolution.
Custom Dashboard	Displays any dashboard you create with the Live Maps Authoring Console in a Summary dashboard.

Widget	Description
Event Summary	Displays a list of objects with the total number of events for the object. You can view the events based on the class of objects or the group. For more information and examples, see "Tutorial" on page 18.
Health State	Display a list or heat map that show the current health state of objects. For more information and examples, see "Health State Widget" on page 24.
HTML	Embeds HTML markup that you can use to display custom information.
Object Properties	Displays the property values for a single selected object.
Performance Graph	Display performance counters in a variety of visualizations. For more information and examples, see "Performance Graph Widget" on page 30.
PowerShell	Display the results of a PowerShell script. There are several options for how you can view the results of a script. The PowerShell widget has the flexibility to show data that is not available in SCOM. For more information and examples, see "PowerShell Widget" on page 32.
Services	Displays the current health state of your Services as a list or as a heat map.
Service Alerts	Displays a listing of individual alerts within a service. You can view the alerts based on the class of objects, the group, the alert state, and the alert resolution.
Service Perspective	Displays the current status of the selected service perspective.
SLA Status	Displays the current SLA status of the selected Service or Perspective.

Widget	Description
SQL	Displays the results of a SQL query. There are several options for how you can view the results of a script. The SQL widget has the flexibility to show data that is not available in SCOM. For more information and examples, see "SQL Widget" on page 39.
Supplier Services	Displays any of the Live Maps Services that supply the selected Service.
Unit Monitors	Displays a list or heat map that shows the current health state of selected Unit Monitors. You can view the objects by group.
Webpage	Embeds the contents of an internal or external website.

Note: The available widgets are dependent on licensing, so you may not see all of the widgets listed above in your Savision Live Maps environment.



CHAPTER 5

Alert Listing Widget

When you add the Alert Listing widget to a dashboard, you can specify the class of objects, the group, the alert state, and the alert resolution to display. For example, you can use this widget to display:

- All alerts for a group of servers.
- Only critical alerts or warnings for all servers.
- Only new alerts created in the last week.

For an example of how to create an Alert Listing widget, see "Tutorial" on page 11.

Tutorial

When you configure the Alert Listing widget, you have the option to use the Advanced field to customize the information that the widget displays. The Advanced field passes a query string directly to SCOM and displays the results in Savision Live Maps. For example, you can use the Advanced field to display alerts that impact a specific department in your organization, or that are assigned to a specific team for resolution.

This example shows how to use the Advanced field to display alerts that are assigned to a specific user for resolution.

1. From the main menu, select **Summary Dashboards**.
2. Click the lock in the top right corner of the dashboard to enter edit mode.
3. Click the **Add** button at the bottom of the screen.
4. Select the **Alert Listing** widget and click **Next**.
5. On the **Data** tab, enter the following information:

Field	Description
Select a Class	Use this field when you want to retrieve information from a specific class only. This field is typically required, but when you use the Advanced field, you do not need to specify a class. For the purposes of this example, leave it blank.

Field	Description
Select a Group	This field is not required. Use this field when you want to retrieve information from a specific group only. For the purposes of this example, leave it blank.
Select an Alert State	Select one or more alert states. The options are Error, Warning, and Info. For this example, select all of them.
Alert Resolution	Select one or more resolution states. For this example, select New.
Time Span	Select a time span for the report. For this example, select One Day.
Top N to display	Enter `10` to show the top 10 alerts that are assigned to the team.
Columns	Click the drop-down menu and select the columns of information that you want to display. For the purposes of this example, select Severity, Source, Name, Last Modified, State, and Created.
Advanced	In this example, you will use the Advanced field to display the name of the user who owns the alert. To retrieve this information, you need to specify the name of the user as it appears in the Owner field on the alert. For example, enter `AND Owner='Tim'`. The `AND` operator is needed in this example because the widget is retrieving information about the owner in addition to the information that you have indicated in the other fields, such as the resolution state. You can use the Advanced field to target multiple properties. For example, if you have used CustomField3 on the alert to indicate that the alert is assigned to the shipping department, enter `AND Owner='Tim' OR CustomField3 like '%shipping%'` to retrieve alerts owned by Tim or assigned to the shipping department.

Tip: You can use the Advanced field to search a wide range of properties. For a full list, click the Information icon on the interface, or use the link in the Related Topics section below.

6. Click the **Views** tab and select the type of visualization you want. For the purposes of this example, select Grid.
7. Click the **General** tab and enter the following information:
    - **Title**—Optional. For this example, enter `Alerts Assigned to Support Team 1`.

- **Refresh Interval**—How often you want the data to refresh. Use the default of 120 seconds.
- **Make Last Updated Time Visible on Widget**—Select the checkbox to enable this option.
- **No Result Message**—Use the default text: `No data available`. This message displays if the query does not return any results.

Related Topics

For more information about the types of searches you can configure in the Advanced field, see "Using the Advanced Field" on page 13.

Using the Advanced Field

The Advanced field is available in the following widgets:

- Health State
- Alert Listing
- Event Summary

You can use the Advanced field to perform a SQL query that searches for object properties. The properties that you can search for depend on the type of widget you are using. The following table lists the properties that the Advanced field supports for each widget.

Table 2: Supported Properties

Property	Health State Widget	Event Summary Widget	Alert Listing Widget
Id	•	•	•
Name	•	•	•
Path	•	•	
FullName	•	•	
DisplayName	•	•	
IsManaged	•	•	
LastModified	•	•	•
HealthState	•	•	
StateLastModified	•	•	
IsAvailable	•	•	
AvailabilityLastModified	•	•	
InMaintenanceMode	•	•	

Property	Health State Widget	Event Summary Widget	Alert Listing Widget
MaintenanceModeLastModified	•	•	•
Description			•
MonitoringObjectId			•
MonitoringClassId			•
MonitoringObjectName			•
MonitoringObjectDisplayName			•
MonitoringObjectPath			•
MonitoringObjectFullName			•
IsMonitorAlert			•
ProblemId			•
MonitoringRuleId			•
ResolutionState			•
Priority			•
Severity			•
Category			•
Owner			•
ResolvedBy			•
TimeRaised			•
TimeAdded			•
LastModifiedBy			•
TimeResolved			•
TimeResolutionStateLastModified			•
CustomField1 to CustomField10			•
TicketId			•
Context			•
ConnectorId			•

Property	Health State Widget	Event Summary Widget	Alert Listing Widget
LastModifiedByNonConnector			•
MonitoringObjectInMaintenanceMode			•
MonitoringObjectHealthState			•
ConnectorStatus			•
NetbiosComputerName			•
NetbiosDomainName			•
PrincipalName			•
AlertParams			•
SiteName			•
StateLastModified			•

You can refine the query by using comparison operators in conjunction with the properties. The query may need to start with AND or OR, to determine how it relates to the other fields. The following table lists the comparison operators that are supported in the Advanced field.

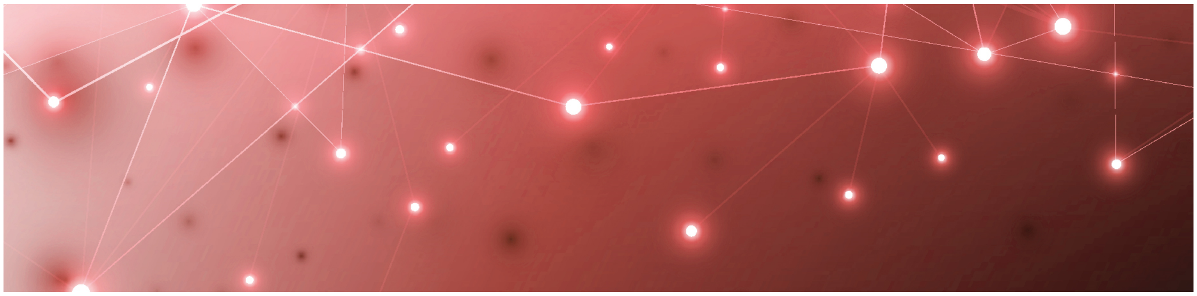
Table 3: Comparison Operators

Operator	Description	Example
=, ==	Returns results if the left and right operand are equal.	Name = 'mymachine.mydomain.com'
!=, <>	Returns results if the left and right operand are unequal.	Name != 'mymachine.mydomain.com'
>	Returns results if the left operand is greater than the right operand.	Severity > 0
<	Returns results if the left operand is less than the right operand.	Severity < 2
>=	Returns results if the left operand is greater than or equal to the right operand.	Severity >= 1

Operator	Description	Example
<=	Returns results if the left operand is less than or equal to the right operand.	Severity <= 3
LIKE	Returns results if the left operand matches the pattern that is defined by the right operand. Use the characters in the wildcard table later in this topic to define the pattern.	Name 'LIKE SQL%' Evaluates to true if the Name value is "SQLEngine." Name LIKE '%SQL%' Evaluates to true if the Name value is "MySQLEngine."
MATCHES	Returns results if the left operand matches the regular expression defined by the right operand. For information about and examples of regular expression syntax, see the MSDN topic, .NET Framework Regular Expressions on the Microsoft website.	Name MATCHES 'SQL*05' Evaluates to true if the Name value is "SQL2005."
IS NULL	Returns results if the value of the left operand is null.	ConnectorId IS NULL Evaluates to true if the ConnectorId property does not contain a value.
IS NOT NULL	Returns results if the value of the left operand is not null.	ConnectorId IS NOT NULL Evaluates to true if the ConnectorId property contains a value.
IN	Returns results if the value of the left operand is in the list of values defined by the right operand. Note: The IN operator is valid for use only with properties of type Guid.	Id IN ('080F192C-52D2-423D-8953-B3EC8C3CD001', '080F192C-53B2-403D-8753-B3EC8C3CD002') Evaluates to true if the value of the Id property is one of the two globally unique identifiers provided in the expression.

Operator	Description	Example
AND	Returns results if the left and right operands are both true.	Name = 'SQL%' AND Description LIKE 'MyData%'
OR	Returns results if either the left or right operand is true.	Name = 'SQL%' OR Description LIKE 'MyData%'
NOT	Returns results if the right operand is not true.	NOT (Name = 'IIS' OR Name = 'SQL')

Note: If you construct an invalid query, the resulting error message provides information that can help you determine why the construction was invalid.



CHAPTER 6

Event Summary Widget

When you add the Event Summary widget to a dashboard, you can display a list of objects with the total number of events for the object. You can view the events based on the class of objects or the group. For example, you can use this widget to display a list of the servers that have the highest number of events.

For an example of how to create an Event Summary widget, see "Tutorial" on page 18.

Tutorial

When you configure the Event Summary widget, you have the option to use the Advanced field to customize the information that the widget displays. The Advanced field passes a query string directly to SCOM and displays the results in Savision Live Maps.

This example shows how to use the Advanced field to display events for objects that are available but that are in an error state or a warning state.

1. From the main menu, select **Summary Dashboards**.
2. Click the lock in the top right corner of the dashboard to enter edit mode.
3. Click the **Add** button at the bottom of the screen.
4. Select the **Event Summary** widget and click **Next**.
5. On the **Data** tab, enter the following information:
    - **Select a Class**—Use this field when you want to retrieve information from a specific class only. This field is typically required, but when you use the Advanced field, you do not need to specify a class. For the purposes of this example, leave it blank.
    - **Select a Group**—This field is not required. Use this field when you want to retrieve information from a specific group only. For the purposes of this example, leave it blank.
    - **Top N to display**—Enter the number of objects you want to display. For this example, enter 100.
    - **Advanced**—For this example, enter `(HealthState = 2 OR HealthState = 3 ) AND IsAvailable = 1`

Tip: You can use the Advanced field to search a wide range of properties, including custom properties. For a full list, click the Information icon on the interface, or use the link in the Related Topics section below.

6. Click the **General** tab and enter the following information:
   - **Title**—Enter a title for the widget. For this example, enter `Events for Available Objects in an Error or Warning State`.
   - **Refresh Interval**—How often you want the data to refresh. Use the default of 120 seconds.
   - **Make Last Updated Time Visible on Widget**—Select the checkbox to enable this option.
   - **No Result Message**—Use the default text: `No data available`. This message displays if the query does not return any results.

Related Topics

For more information about the types of searches you can configure in the Advanced field, see "Using the Advanced Field" on page 19.

Using the Advanced Field

The Advanced field is available in the following widgets:

- Health State
- Alert Listing
- Event Summary

You can use the Advanced field to perform a SQL query that searches for object properties. The properties that you can search for depend on the type of widget you are using. The following table lists the properties that the Advanced field supports for each widget.

Table 4: Supported Properties

Property	Health State Widget	Event Summary Widget	Alert Listing Widget
Id	•	•	•
Name	•	•	•
Path	•	•	
FullName	•	•	

Property	Health State Widget	Event Summary Widget	Alert Listing Widget
DisplayName	●	●	
IsManaged	●	●	
LastModified	●	●	●
HealthState	●	●	
StateLastModified	●	●	
IsAvailable	●	●	
AvailabilityLastModified	●	●	
InMaintenanceMode	●	●	
MaintenanceModeLastModified	●	●	●
Description			●
MonitoringObjectId			●
MonitoringClassId			●
MonitoringObjectName			●
MonitoringObjectDisplayName			●
MonitoringObjectPath			●
MonitoringObjectFullName			●
IsMonitorAlert			●
ProblemId			●
MonitoringRuleId			●
ResolutionState			●
Priority			●
Severity			●
Category			●
Owner			●
ResolvedBy			●
TimeRaised			●

Property	Health State Widget	Event Summary Widget	Alert Listing Widget
TimeAdded			•
LastModifiedBy			•
TimeResolved			•
TimeResolutionStateLastModified			•
CustomField1 to CustomField10			•
TicketId			•
Context			•
ConnectorId			•
LastModifiedByNonConnector			•
MonitoringObjectInMaintenanceMode			•
MonitoringObjectHealthState			•
ConnectorStatus			•
NetbiosComputerName			•
NetbiosDomainName			•
PrincipalName			•
AlertParams			•
SiteName			•
StateLastModified			•

You can refine the query by using comparison operators in conjunction with the properties. The query may need to start with AND or OR, to determine how it relates to the other fields. The following table lists the comparison operators that are supported in the Advanced field.

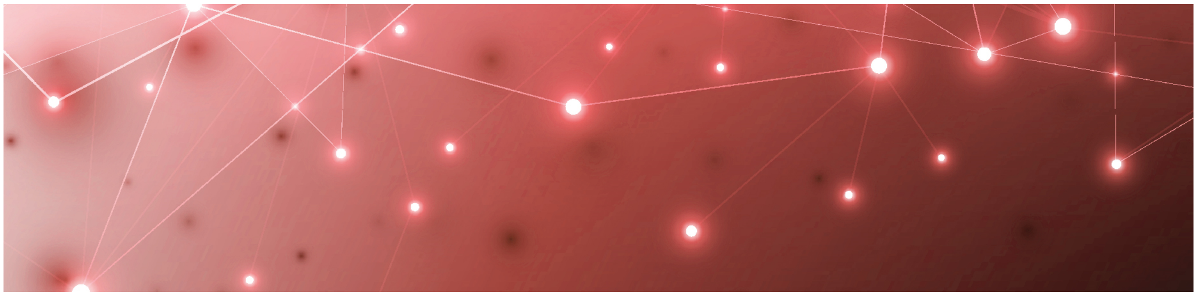
Table 5: Comparison Operators

Operator	Description	Example
=, ==	Returns results if the left and right operand are equal.	Name = 'mymachine.mydomain.com'

Operator	Description	Example
!=, <>	Returns results if the left and right operand are unequal.	Name != 'mymachine.mydomain.com'
>	Returns results if the left operand is greater than the right operand.	Severity > 0
<	Returns results if the left operand is less than the right operand.	Severity < 2
>=	Returns results if the left operand is greater than or equal to the right operand.	Severity >= 1
<=	Returns results if the left operand is less than or equal to the right operand.	Severity <= 3
LIKE	Returns results if the left operand matches the pattern that is defined by the right operand. Use the characters in the wildcard table later in this topic to define the pattern.	Name 'LIKE SQL%' Evaluates to true if the Name value is "SQLEngine." Name LIKE '%SQL%' Evaluates to true if the Name value is "MySQLEngine."
MATCHES	Returns results if the left operand matches the regular expression defined by the right operand. For information about and examples of regular expression syntax, see the MSDN topic, .NET Framework Regular Expressions on the Microsoft website.	Name MATCHES 'SQL*05' Evaluates to true if the Name value is "SQL2005."
IS NULL	Returns results if the value of the left operand is null.	ConnectorId IS NULL Evaluates to true if the ConnectorId property does not contain a value.

Operator	Description	Example
IS NOT NULL	Returns results if the value of the left operand is not null.	ConnectorId IS NOT NULL Evaluates to true if the ConnectorId property contains a value.
IN	Returns results if the value of the left operand is in the list of values defined by the right operand. Note: The IN operator is valid for use only with properties of type Guid.	Id IN ('080F192C-52D2-423D-8953-B3EC8C3CD001', '080F192C-53B2-403D-8753-B3EC8C3CD002') Evaluates to true if the value of the Id property is one of the two globally unique identifiers provided in the expression.
AND	Returns results if the left and right operands are both true.	Name = 'SQL%' AND Description LIKE 'MyData%'
OR	Returns results if either the left or right operand is true.	Name = 'SQL%' OR Description LIKE 'MyData%'
NOT	Returns results if the right operand is not true.	NOT (Name = 'IIS' OR Name = 'SQL')

Note: If you construct an invalid query, the resulting error message provides information that can help you determine why the construction was invalid.



CHAPTER 7

Health State Widget

The Health State widget displays the current health of objects based on their class or group, or on a specified health state. For example, you can use this widget to:

- Show all members of a group and their current health state.
- Show Only critical or warning states for members of a class.

You can view the information in a list or in a heat map.

For an example of how to create a Health State widget, see "Tutorial" on page 24.

Tutorial

This example shows how to use the Advanced field to display objects that are in maintenance mode. You can display objects that are in maintenance mode within a specified period of time, or that changed state within a specified period of time.

1. From the main menu, select **Summary Dashboards**.
2. Click the lock in the top right corner of the dashboard to enter edit mode.
3. Click the **Add** button at the bottom of the screen.
4. Select the **Health State** widget and click **Next**.
5. On the **Data** tab, enter the following information:
   - **Select a Class**—Use this field when you want to retrieve information from a specific class only. This field is typically required, but when you use the Advanced field, you do not need to specify a class. For the purposes of this example, leave it blank.
   - **Select a Group**—This field is not required. Use this field when you want to retrieve information from a specific group only. For the purposes of this example, leave it blank.
   - **Include Sub-Members**—If you scoped the search to a group, you can select the checkbox if you want to include sub-members.
   - **Health States**—You can use this field to select any of the following health states: No data available, Not monitored, Healthy, Warning, or Error.
   - **Top N to display**—Enter 100 to show the top 100 objects that are in maintenance mode.

- **Sort By**—Select Health State from the drop-down menu.
- **Advanced**—Enter the following string: `InMaintenanceMode = 1`.

> **Tip:** If you selected options in the Health State field, you need to add the `AND` operator to the query in the Advanced field. The `AND` operator is needed in this case because the widget is retrieving information specified in the Advanced field in addition to the health state information. For example, enter
> `AND InMaintenanceMode = 1`

6. Click the **Views** tab and select the type of visualization you want. For the purposes of this example, select the Available option for the Pie Chart and Heat Map visualizations, and make the Heat Map the default view.
7. Click **Options** and select the display options for the widget:
    - **Entity Label Options**—Select Show Object Name and Show Class Name.
    - **Select Path**—From the drop-down list, select Don't Show Path.
    - **Select Columns**—From the drop-down list, select 2.
8. Click the **General** tab and enter the following information:
    - **Title**—optional. For this example, enter `Objects in Maintenance Mode`.
    - **Refresh Interval**—How often you want the data to refresh.
    - **Make Last Updated Time Visible on Widget**—Select the checkbox to enable this option.
    - **No Result Message**—Use the default text: `No data available`. This message displays if the query does not return any results.

Related Topics

For more information about the types of searches you can configure in the Advanced field, see "Using the Advanced Field" on page 25.

Using the Advanced Field

The Advanced field is available in the following widgets:

- Health State
- Alert Listing
- Event Summary

You can use the Advanced field to perform a SQL query that searches for object properties. The properties that you can search for depend on the type of widget you are using. The following table lists the properties that the Advanced field supports for each widget.

Table 6: Supported Properties

Property	Health State Widget	Event Summary Widget	Alert Listing Widget
Id	•	•	•
Name	•	•	•
Path	•	•	
FullName	•	•	
DisplayName	•	•	
IsManaged	•	•	
LastModified	•	•	•
HealthState	•	•	
StateLastModified	•	•	
IsAvailable	•	•	
AvailabilityLastModified	•	•	
InMaintenanceMode	•	•	
MaintenanceModeLastModified	•	•	•
Description			•
MonitoringObjectId			•
MonitoringClassId			•
MonitoringObjectName			•
MonitoringObjectDisplayName			•
MonitoringObjectPath			•
MonitoringObjectFullName			•
IsMonitorAlert			•
ProblemId			•
MonitoringRuleId			•
ResolutionState			•
Priority			•
Severity			•

Property	Health State Widget	Event Summary Widget	Alert Listing Widget
Category			•
Owner			•
ResolvedBy			•
TimeRaised			•
TimeAdded			•
LastModifiedBy			•
TimeResolved			•
TimeResolutionStateLastModified			•
CustomField1 to CustomField10			•
TicketId			•
Context			•
ConnectorId			•
LastModifiedByNonConnector			•
MonitoringObjectInMaintenanceMode			•
MonitoringObjectHealthState			•
ConnectorStatus			•
NetbiosComputerName			•
NetbiosDomainName			•
PrincipalName			•
AlertParams			•
SiteName			•
StateLastModified			•

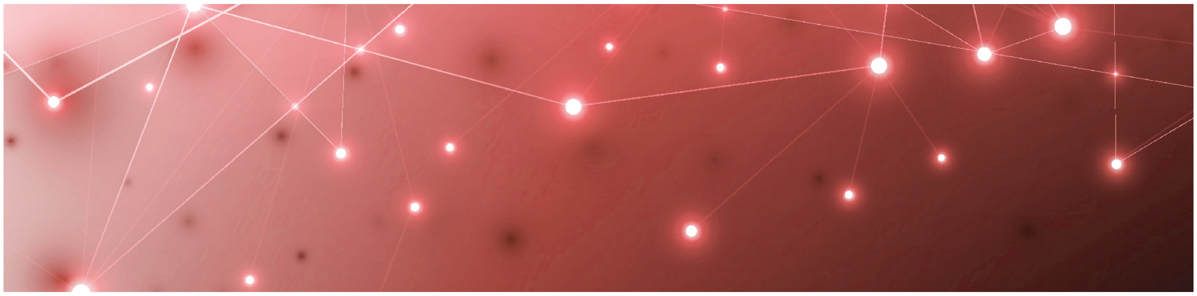
You can refine the query by using comparison operators in conjunction with the properties. The query may need to start with AND or OR, to determine how it relates to the other fields. The following table lists the comparison operators that are supported in the Advanced field.

Table 7: Comparison Operators

Operator	Description	Example
=, ==	Returns results if the left and right operand are equal.	Name = 'mymachine.mydomain.com'
!=, <>	Returns results if the left and right operand are unequal.	Name != 'mymachine.mydomain.com'
>	Returns results if the left operand is greater than the right operand.	Severity > 0
<	Returns results if the left operand is less than the right operand.	Severity < 2
>=	Returns results if the left operand is greater than or equal to the right operand.	Severity >= 1
<=	Returns results if the left operand is less than or equal to the right operand.	Severity <= 3
LIKE	Returns results if the left operand matches the pattern that is defined by the right operand. Use the characters in the wildcard table later in this topic to define the pattern.	Name 'LIKE SQL%' Evaluates to true if the Name value is "SQLEngine." Name LIKE '%SQL%' Evaluates to true if the Name value is "MySQLEngine."
MATCHES	Returns results if the left operand matches the regular expression defined by the right operand. For information about and examples of regular expression syntax, see the MSDN topic, .NET Framework Regular Expressions on the Microsoft website.	Name MATCHES 'SQL*05' Evaluates to true if the Name value is "SQL2005."

Operator	Description	Example
IS NULL	Returns results if the value of the left operand is null.	ConnectorId IS NULL Evaluates to true if the ConnectorId property does not contain a value.
IS NOT NULL	Returns results if the value of the left operand is not null.	ConnectorId IS NOT NULL Evaluates to true if the ConnectorId property contains a value.
IN	Returns results if the value of the left operand is in the list of values defined by the right operand. Note: The IN operator is valid for use only with properties of type Guid.	Id IN ('080F192C-52D2-423D-8953-B3EC8C3CD001', '080F192C-53B2-403D-8753-B3EC8C3CD002') Evaluates to true if the value of the Id property is one of the two globally unique identifiers provided in the expression.
AND	Returns results if the left and right operands are both true.	Name = 'SQL%' AND Description LIKE 'MyData%'
OR	Returns results if either the left or right operand is true.	Name = 'SQL%' OR Description LIKE 'MyData%'
NOT	Returns results if the right operand is not true.	NOT (Name = 'IIS' OR Name = 'SQL')

Note: If you construct an invalid query, the resulting error message provides information that can help you determine why the construction was invalid.



CHAPTER 8

Performance Graph Widget

When you use the Performance Graph widget to create a dashboard, you can visualize performance data in several formats. The following are a few examples of the formats available:

- **Sparkline**—Allows you to see the trend for each counter and its minimum, maximum, and current value.
- **Entity state list**
- **Grid**—Shows the data retrieved for the performance graphs in table format, and exports all of the data to Excel.
- **Heat map**—Allows you to see the current health state of the objects.

Other choices include line and bar charts. All of the formats allow you to sort data by largest or smallest values, and over a specified period of time.

For an example of how to create a Performance Graph widget, see "Tutorial" on page 30.

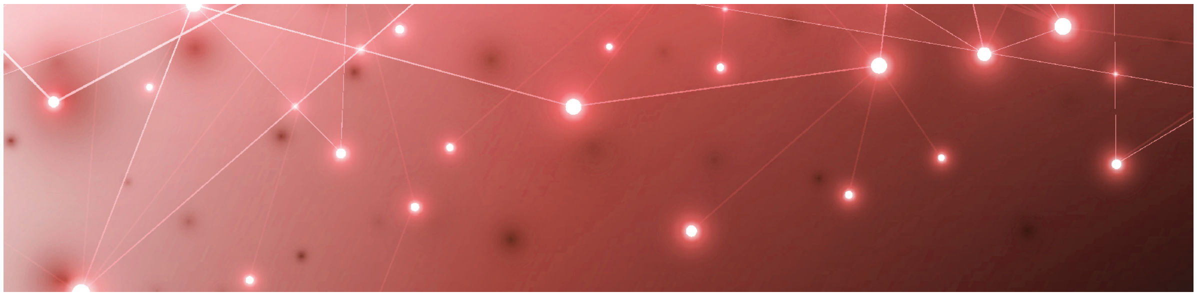
Tutorial

This example shows you how to configure a Performance Graph widget to display information about computer memory. In this example, the widget is configured to show the 10 computers with the least available memory. The data is displayed in an Entity State List.

1. From the main menu, select **Summary Dashboards**.
2. Click the lock in the top right corner of the dashboard to enter edit mode.
3. Click the **Add** button at the bottom of the screen.
4. Select the **Performance Graph** widget and click **Next**.
5. Enter the following information:

Field	Description
Select Performance Rule	Enter a search term or select Show All to see all performance rules that are configured in SCOM. For example, select `% Logical Free Disk Space Windows Server 2012`.
Select a Group	Optional. You can scope the search to a group to refine the information. For example, select a group to refine the search to a specific service, such as an IIS webserver group.
Select a unit of measure	For this example, select GB.
Time span	To view trends over time, such as whether the usage of the web server is increasing, select a longer period of time. To see the current status, select a shorter period of time.
Data type	Select the type of data you want to view: minimum, average, or maximum.
Sort by	Select lowest or highest. To view computers with the least available memory, select lowest.
Data at least this recent	Use the default setting of 60 minutes.
Floor and Include Counters without Values in Timeframe	By default, the graph is displayed in the context of its minimum and maximum values. For example, if the minimum free disk space was 10 GB and the maximum was 15 GB, the graph displays this range only. To view the graph from 0 to the maximum value, select the checkbox.
Top N to display	Use the default value of 10.

6. Click the **Views** tab and select **Entity State List**.



CHAPTER 9

PowerShell Widget

The PowerShell widget allows you to use a PowerShell script to retrieve information and display it on a Savision Live Maps Summary dashboard. You can use the widget to display SCOM data, as well as data that is not available in SCOM.

There are several ways that you can visualize the data that you retrieve:

- Grid—This view displays data in a table.
- Single metric—This view displays a single value.
- Bullet graph—This view displays three values: actual, target, and maximum.
- Pie chart
- Entity state list
- Heat map
- Spark line—Allows you to see the trend for each counter and its most recent value.
- Line chart
- Console
- Horizontal bar
- Horizontal bar compact

You can choose the view that best suits the data. For example, you can use the grid view for objects in a database, or you can use a pie chart to represent the number of computers in specified health states.

Each type of visualization has its own requirements for the formatted properties of returned objects. The following table lists the requirements for each view.

Table 8: Requirements for PowerShell Data Views

View	Requirement
Grid	Binds to a dynamic list of fields.

View	Requirement
Single metric	Binds to the following fields: • Title—a string. • Value—can be any return type but it is treated as a string for rendering.
Bullet graph	Binds to the following fields: • Title—string (optional) • Target—number • Actual—number • Maximum—number
Pie chart	Binds to the following fields: • Title—string • Target—number
Entity state list	Binds to the following fields: • Id - Guid—if this is a valid monitoring object id in SCOM, you can click on the object to be redirected to the Object Explorer in Savision Live Maps. • DisplayName—a string • Class—a string • IsAvailable—boolean • InMaintenanceMode—boolean • HealthState—a string with one of the following values: • NoDataAvailable • Uninitialized • Success • Warning • Error

View	Requirement
Heat map	Binds to the following fields: • Id - Guid—if this is a valid monitoring object id in SCOM, you can click on the object to be redirected to the Object Explorer in Savision Live Maps. • DisplayName—a string • Class—a string (optional/empty string) • IsAvailable—boolean • InMaintenanceMode—boolean • HealthState—a string with one of the following values: • NoDataAvailable • Uninitialized • Success • Warning • Error
Sparkline	Allows you to see the trend for each counter and its most recent value.
Console	No requirements. This view displays the results of the script in a format like the PowerShell console.
Line chart Horizontal Bar Horizontal Bar Compact	Binds to the following fields: • key—a string, which must be unique for each line or bar series • x—datetime in javascript (epoch) format • y—number

For an example of how to create a PowerShell widget, see "Tutorial" on page 34.

Tutorial

This section provides an example of how you can implement a PowerShell widget that reads a CSV file generated by Microsoft System Center Configuration Manager (SCCM). In this example, the CSV file contains compliance data for a software update group. The PowerShell script reads the CSV file and displays the information in Savision Live Maps as a pie chart.

1. From the main menu, select **Summary Dashboards**.
2. Click the lock in the top right corner of the dashboard to enter edit mode.
3. Click the **Add** button at the bottom of the screen.
4. Select the **PowerShell** widget and click **Next**.
   The Data tab displays by default.
5. Click the **General** tab and enter the following information:

- **Title**—Optional. Enter a title for the widget.
- **Refresh interval**—Indicate how often you want the data to refresh.
- **Make Last Updated Time Visible on Widget**—Select the checkbox to enable this option.
- **No Result Message**—Add a customizable HTML message to display if the query does not return a result.

6. Click the **Views** tab and select the type of Visualization you want. This example chooses the **Pie Chart** view as the default view, and does not make any other option available.
   The default visualization displays every time the dashboard is viewed. When you select the Available checkbox for one or more visualizations, users can choose any of the available views to change the way they see the data. When you make additional views available, ensure that your script meets the requirements for that type of view. For an example of a script that displays the information in this example in the Single Metric format, see "View CSV Data as a Single Metric" on page 35.
7. Click the **Data** tab and enter the PowerShell script. For example:

```
$csvPath = 'C:\DemoData\Compliance 1 - Overall compliance.csv'


$csvData = Get-Content -Path $csvPath | Select-Object -Skip 6 | Out-
String | ConvertFrom-Csv | select @{Name= "Title"; Expression =
{$_.Details_Table0_Status}},
    @{Name= "Value"; Expression = {$_.Details_Table0_PComputers}}



$csvData | Sort-Object -property Title -des
```

8. Click **Run**.
   The right side of the page displays a preview of the query results. Review the results and adjust the script if needed.
9. Click **Add**.

Additional Views

This section provides a sample script that you can use to show the same query results in the following view:

- "View CSV Data as a Single Metric" on page 35

Each type of visualization has its own requirements for the formatted properties of returned objects. For a list of the requirements, see "PowerShell Widget" on page 32.

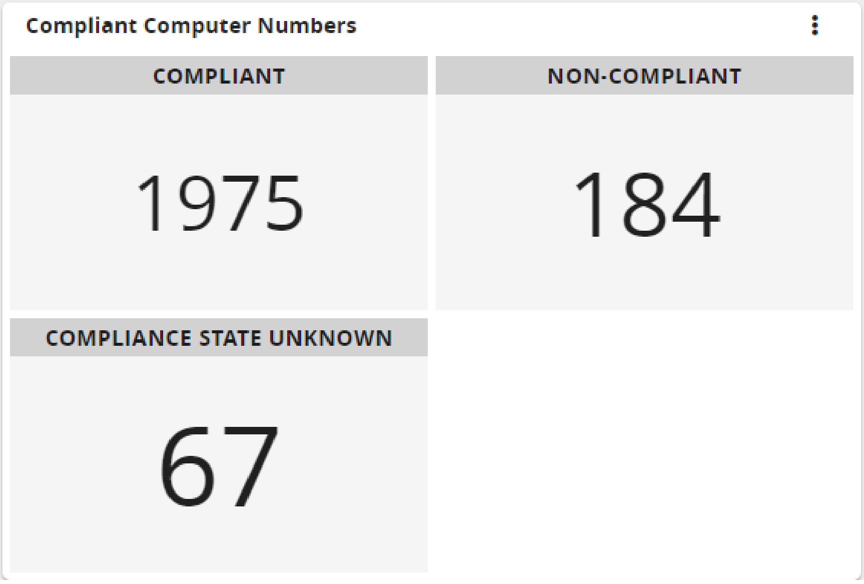
View CSV Data as a Single Metric

The following script reads a CSV file generated by Microsoft System Center Configuration Manager (SCCM) and displays the information in Savision Live Maps in Single Metric format.

```
DescriptionLabelTextbox,DescriptionTextbox


Description,This report returns the overall compliance data for a
software update group.


Header_Table1_CollectionName,Header_Table1_NumberInCollection,Details_
Table1_CollectionName,Details_Table1_NumberInCollection

Collection Name,Clients in Collection,Prod_Windows10,2226


Header_Table0_Status,Header_Table0_NumberOfComputers,Header_Table0_
PComputers,Details_Table0_Status,Details_Table0_
NumberOfComputers,Details_Table0_PComputers

State,Count of Computers,% of Total,Compliant,1975,88.72

State,Count of Computers,% of Total,Non-compliant,184,8.27

State,Count of Computers,% of Total,Compliance state unknown,67,3.01
```

The following image shows how the results of this script display on the Summary dashboard.

Tip: When you configure the widget to display data in the Single Metric format, you can click the **Options** tab and specify the number of columns to show.



Compliant Computer Numbers

COMPLIANT	NON-COMPLIANT
1975	184
COMPLIANCE STATE UNKNOWN	
67	

Sample Scripts

This section provides sample scripts that show you how you can use the PowerShell widget to perform the following tasks:

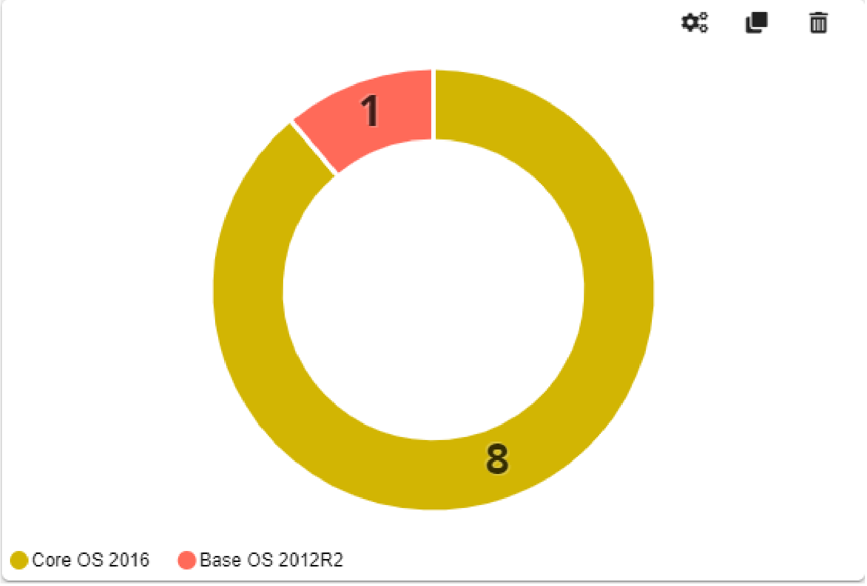
- "View Windows Operating Systems as a Pie Chart" on page 37

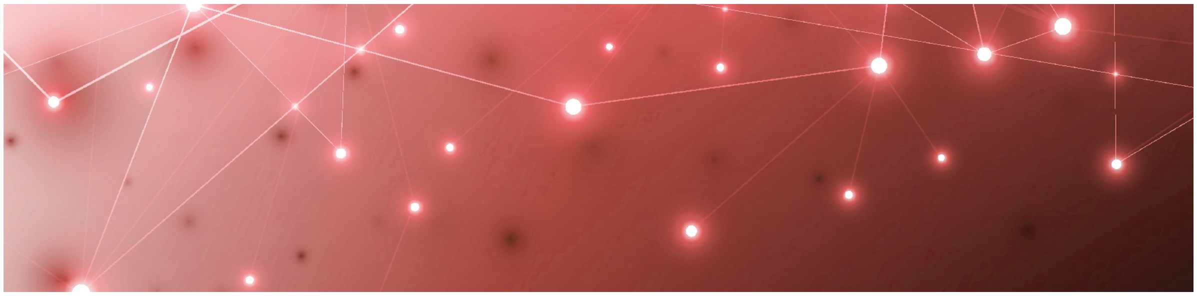
View Windows Operating Systems as a Pie Chart

The following script generates a count of Windows computers according to operating system and displays the information in Savision Live Maps in pie chart format.

```
Import-module OperationsManager
New-SCOMManagementGroupConnection -ComputerName <SCOM MGMT Server>


$Result = @()

$Target = 10 #Set target for Rate Gauges

$Max = 0

$Versions = @()

$Versions = Get-SCOMClass -Name Microsoft.Windows.Computer | Get-
SCOMClassInstance | Select-Object

@{LABEL='Version';EXPRESSION='`
[LW.Base.Microsoft.Windows.Server.BaseOS.Category].CategoryName'} |
foreach {$_.'Version'.value} | Select-Object -Unique

$UniqueVersions = $Versions.Count


$Computers = Get-SCOMClass -Name Microsoft.Windows.Computer | Get-
SCOMClassInstance | Select-Object
Id,HealthState,InMaintenanceMode,IsAvailable,@
{LABEL='Server';EXPRESSION='`
[Microsoft.Windows.Computer].PrincipalName'},@
{LABEL='Version';EXPRESSION='`
[LW.Base.Microsoft.Windows.Server.BaseOS.Category].CategoryName'}

for ($i=0; $i -lt $UniqueVersions; $i++) {

    $versionCount = 0

    Foreach ($computer in $Computers) {

        If ($computer.'Version'.value -eq $Versions[$i]) {
        $versionCount++
        }
        }

        $TempValue = New-Object System.Object
        $TempValue | Add-Member -type NoteProperty -name Title -Value
        $Versions[$i]
        $TempValue | Add-Member -type NoteProperty -name Value -Value
        $versionCount

        $Result += $TempValue
    }
    $Result
```

The following image shows how the results of this script display on the Summary dashboard.





CHAPTER 10

SQL Widget

The SQL widget allows you to use a SQL script to retrieve information and display it on a Savision Live Maps Summary dashboard. You can use the widget to display SCOM data, as well as data that is not available in SCOM.

There are several ways that you can visualize the data that you retrieve:

- Grid—This view displays data in a table
- Single metric—This view displays a single value.
- Bullet graph—This view displays three values: actual, target, and maximum.
- Pie chart
- Entity state list
- Heat map
- Spark line—Allows you to see the trend for each counter and its most recent value.
- Line chart
- Horizontal bar
- Horizontal bar compact

You can choose the view that best suits the data. For example, you can use the grid view for objects in a database, or you can use a pie chart to represent the number of computers in specified health states.

Each type of visualization has its own requirements for the formatted properties of returned objects. The following table lists the requirements for each view.

Table 9: Requirements for SQL Data Views

View	Requirement
Grid	Binds to a dynamic list of fields.

View	Requirement
Single metric	Binds to the following fields: • Title—a string. • Value—can be any return type but it is treated as a string for rendering.
Bullet graph	Binds to the following fields: • Title—string (optional) • Target—number • Actual—number • Maximum—number
Pie chart	Binds to the following fields: • Title—string • Target—number
Entity state list	Binds to the following fields: • Id - Guid—if this is a valid monitoring object id in SCOM, you can click on the object to be redirected to the Object Explorer in Savision Live Maps. • DisplayName—a string • Class—a string • IsAvailable—boolean • InMaintenanceMode—boolean • HealthState—a string with one of the following values: • NoDataAvailable • Uninitialized • Success • Warning • Error

View	Requirement
Heat map	Binds to the following fields: • Id - Guid—if this is a valid monitoring object id in SCOM, you can click on the object to be redirected to the Object Explorer in Savision Live Maps. • DisplayName—a string • Class—a string (optional/empty string) • IsAvailable—boolean • InMaintenanceMode—boolean • HealthState—a string with one of the following values: • NoDataAvailable • Uninitialized • Success • Warning • Error
Spark line	Allows you to see the trend for each counter and its most recent value.
Line chart Horizontal Bar Horizontal Bar Compact	Binds to the following fields: • key—a string, which must be unique for each line or bar series • x—datetime in javascript (epoch) format • y—number

For an example of how to create a SQL widget, see "Tutorial" on page 41.

Tutorial

This section provides an example of how you can implement a SQL widget that retrieves a list of Windows computers in a specified group and displays them in a grid format that you can sort.

1. From the main menu, select **Summary Dashboards**.
2. Click the lock in the top right corner of the dashboard to enter edit mode.
3. Click the **Add** button at the bottom of the screen.
4. Select the **SQL** widget and click **Next**.
5. Click the **General** tab and enter the following information:
   - **Title**—optional
   - **Refresh interval**—How often you want the data to refresh.
   - **No Result Message**—A customizable HTML message to display if the query does not return a result.

6. Click the **Views** tab and select the type of Visualization you want. This example uses the Grid view as the default view, and does not make any other visualizations available.
   The default visualization displays every time the dashboard is viewed. When you select the **Available** checkbox for one or more visualizations, users can choose any of the available views to change the way they see the data. When you make additional views available, ensure that your script meets the requirements for that type of view, as shown in the "Additional Views" on page 42.
7. Click **Options** and select the number of columns for the grid display.
8. Click the **Data** tab and select a database:
   - **DataWarehouse**— the DW database
   - **Operations**—the Operations Manager database
   - **SQLServer**—any other SQL instance
   - **Oracle**—Oracle database

   This example uses the Operations database.
9. Click the **Query** tab and enter the script. For example:

```
Declare @ComputerGroup varchar(255) = 'All Windows Computers'


SELECT Distinct megv.DisplayName as DisplayName,
     megv.BaseManagedEntityId as Id,
     megv.HealthState,
     megv.IsAvailable,
     megv.InMaintenanceMode,
     'Windows Computer' as Class
FROM ManagedEntityGenericView megv
INNER JOIN RelationshipGenericView rgv WITH(NOLOCK) ON
megv.basemanagedentityid = rgv.TargetObjectId
WHERE megv.IsDeleted = '0'
AND rgv.SourceObjectDisplayName = @ComputerGroup
ORDER BY megv.displayname
```

10. Click **Run**.
    The right side of the page displays a preview of the query results. Review the results and adjust the script if needed.
11. Click **Add**.

Additional Views

This section provides sample scripts that you can use to show the same query results in the following views:

- "View Results as a Heat Map" on page 43
- "View Results as a Pie Chart" on page 44

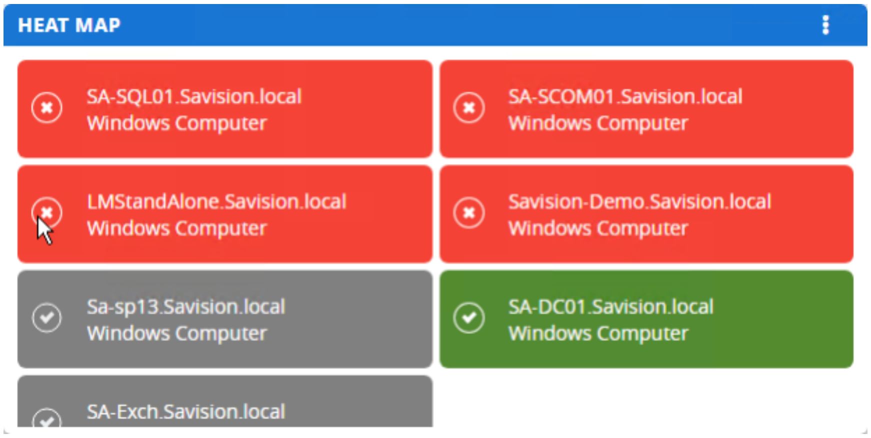
Each type of visualization has its own requirements for the formatted properties of returned objects. For a list of the requirements, see " " on page 39

View Results as a Heat Map

The following script searches for Windows computers and displays the results in a heat map that is organized by health state.

```
Declare @ComputerGroup varchar(255) = 'All Windows Computers'


SELECT Distinct megv.DisplayName as DisplayName,
     megv.BaseManagedEntityId as Id,
     Case When megv.HealthState = 3 Then 'Error'
     When megv.HealthState = 2 Then 'Warning'
     When megv.HealthState = 1 Then 'Success'
     Else 'Uninitialized' End As HealthState,
     megv.IsAvailable as IsAvailable,
     megv.InMaintenanceMode as InMaintenanceMode,
     'Windows Computer' as Class


FROM ManagedEntityGenericView megv
INNER JOIN RelationshipGenericView rgv WITH(NOLOCK) ON
megv.basemanagedentityid = rgv.TargetObjectId
WHERE megv.IsDeleted = '0'
AND rgv.SourceObjectDisplayName = @ComputerGroup
ORDER BY Healthstate ASC
```

The following image shows how the results of this script display on the Summary dashboard.



SA-SQL01.Savision.local
Windows Computer

SA-SCOM01.Savision.local
Windows Computer

LMStandAlone.Savision.local
Windows Computer

Savision-Demo.Savision.local
Windows Computer

Sa-sp13.Savision.local
Windows Computer

SA-DC01.Savision.local
Windows Computer

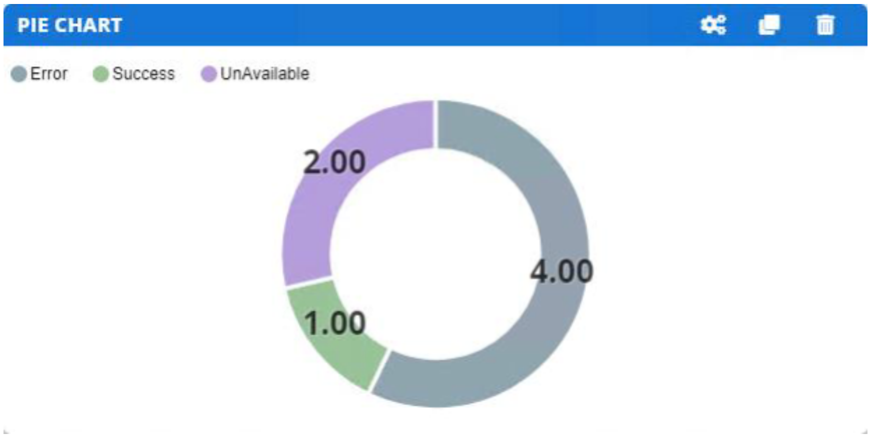
SA-Exch.Savision.local

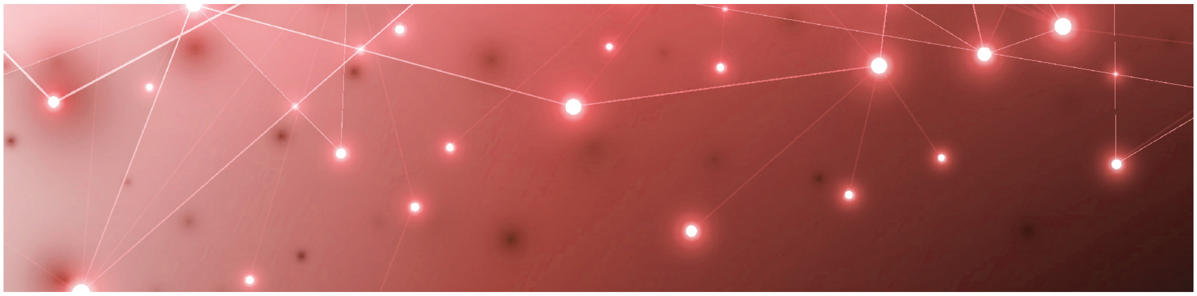
View Results as a Pie Chart

The following script searches for Windows computers and displays the results in a pie chart.

```
Declare @ComputerGroup varchar(255) = 'All Windows Computers'


SELECT Count(t.Value) as Value,

     T.Value as Title


FROM (SELECT Distinct megv.DisplayName as DisplayName,

     megv.BaseManagedEntityId as Id,
     Case When megv.IsAvailable = 0 Then 'UnAvailable'
     When megv.HealthState = 3 Then 'Error'
     When megv.HealthState = 2 Then 'Warning'
     When megv.HealthState = 1 Then 'Success'
     Else 'Uninitialized' End As Value,
     megv.IsAvailable as IsAvailable,
     megv.InMaintenanceMode as InMaintenanceMode,
     'Windows Computer' as Class


FROM ManagedEntityGenericView megv

INNER JOIN RelationshipGenericView rgv WITH(NOLOCK) ON

megv.basemanagedentityid = rgv.TargetObjectId

WHERE megv.IsDeleted = '0'

AND rgv.SourceObjectDisplayName = @ComputerGroup) t

Group by t.Value
```

The following image shows how the results of this script display on the Summary dashboard.





CHAPTER 11

Pre-Configured Dashboards

This release includes pre-configured dashboards that you can import and customize. The pre-configured dashboards are built using Summary Dashboard widgets.

Each dashboard includes an About this Dashboard section at the top, which describes the purpose of the dashboard and shows an example of what the dashboard will look like after you configure it. You can customize this section with your own information and branding, or you can delete it from the dashboard when you have completed the configuration.

The following sections describe the types of pre-configured dashboards that are available, and provide instructions for importing them into your Savision Live Maps environment.

Item	Description
Pre-Configured Dashboards	
"Client Computer Health" on page 47	This dashboard displays health information for Windows client computers.
"Configuration Manager Clients" on page 48	This dashboard displays the status of Configuration Manager clients that have a client agent installed.
"Configuration Manager Software" on page 49	This dashboard displays the status of the software on Configuration Manager clients, such as information about software updates and operating systems.
"FTP" on page 50	This dashboard shows useful information about the state of the FTP servers within your Live Maps environment.
"Hyper-V" on page 51	This dashboard displays information about the health of your Hyper-V devices.

Item	Description
"SCOM Data Warehouse SQL Database Health" on page 52	This dashboard displays information about the health of your SQL databases.
"SCOM Environment Health" on page 54	This dashboard displays health information for your SCOM environment and hardware.
"SCOM Noise" on page 55	This dashboard displays information about the parts of your SCOM environment that generally cause performance issues.
"Server Health" on page 56	This dashboard displays health information for computers operating on Windows servers.
"Service Manager Health" on page 56	Shows the health of your Systems Center Service Manager environment.
"Service Manager Work Items" on page 57	This dashboard displays information about work items that impact the Service Manager environment.
Procedures	
"Import a Dashboard" on page 58	Import a pre-configured dashboard into your environment.
"Customize a Dashboard" on page 59	Edit the pre-configured dashboard to customize it for your Savision Live Maps environment.

Client Computer Health

The Client Computer Health dashboard displays health information for Windows client computers. It includes a heat map of health states on the left and an alert listing widget on the right. The health state widget provides a quick look to briefly see if computers are either healthy, unhealthy, or unresponsive. The alert listing widget provides more specific notifications for each computer to aid in the troubleshooting process.

The following table lists the types of information retrieved and the visualization format used for each.

Table 10: Health States of Windows Client Computers

Information	Visualization Type
Client Computer Health States	Uses the Health States Widget to show a heat map. The heat map is the default view; users have the option to view the information as an Entity State List or a pie chart.
Client Computer Alerts	Uses the Alert Listing widget to show data in a pie chart. The pie chart is the default view; users have the option to view the information in a grid.

To import this pre-configured dashboard into Savision Live Maps and customize it for your environment, complete the following tasks:

- "Import a Dashboard" on page 58
- "Customize a Dashboard" on page 59

Configuration Manager Clients

This dashboard displays the status of Configuration Manager clients that have a client agent installed. The following table lists the types of information retrieved and the visualization format used for each.

Table 11: Configuration Manager Client Information

Information	Visualization Type
Client Agent Installation	Displays the number of devices that have a client agent installed. It is configured to display as a pie chart by default, but users have the option to view it in a single metric format.
Status of CM Client Computers	Displays the status of devices that have a client agent installed. It is configured to display as a heat map by default, but users have the option to view it in grid format.
Logical Disk Free Space (GB)	Displays the eight devices with the least amount of free space on the logical disk. It is configured to display as single metric by default, but users have the option to view it as a pie chart.
Type of Machine and Number of Processors	Displays the type of device, its name and manufacturer, as well athe number of processors. It is configured to display in grid format only.

To import this pre-configured dashboard into Savision Live Maps and customize it for your environment, complete the following tasks:

- "Import a Dashboard" on page 58
- "Customize a Dashboard" on page 59

Configuration Manager Software

The Configuration Manager Software dashboard displays information about the status of the software on Configuration Manager clients, such as information about software updates and operating systems. The following table lists the types of information retrieved and the visualization format used for each.

Table 12: Configuration Manager Software

Information	Visualization Type
"Windows 8 test" Update Deployment by Type	Shows security updates and Windows patches. It is configured to display as a pie chart by default, but users have the option to view it as a single metric. The Windows 8 test group is an example only. You can customize this widget by specifying which deployment group you want to target for this query. Deployment groups are defined in Configuration Manager.
Operating Systems	Shows the number of computers that are running each operating system. It is configured to display as a pie chart by default, but users have the option to view it as a single metric or grid.
Updates per Device Collection	Device collections are defined in Configuration Manager. Customize this widget by selecting which device collection to target. It is configured to display as a pie chart by default, but users have the option to view it as a single metric.
Chrome Installation on Clients	Shows the clients where Chrome is installed and the health state of each client. You can modify the script to search for different software. The default view is a heat map.
Computers Requiring Updates	Displays a list of computers that require software updates. The default view is a grid.

Information	Visualization Type
Update Percent Compliancy	Shows the status of updates, with a timestamp indicating when the update was published. For updates that are in progress, this widget displays the percentage of the available updates that are installed. The default view is a horizontal bar, but users have the option to view it as a compact horizontal bar.
Number of devices per device collection	Lists the number of devices in each device collection, as defined by the SCCM administrator. The default view is a grid.
PowerShell Script Status	Lists the status of PowerShell scripts so that you can see which scripts have been executed successfully and which ones have failed. The default view is a grid.

To import this pre-configured dashboard into Savision Live Maps and customize it for your environment, complete the following tasks:

- "Import a Dashboard" on page 58
- "Customize a Dashboard" on page 59

FTP

This dashboard shows useful information about the state of the FTP servers within your Live Maps environment. The following table lists the types of information retrieved and the visualization format used for each.

Table 13: FTP

Information	Visualization Type
FTP Server Health	Shows the health states of the top 100 FTP servers. It is configured to display as a heat map.
FTP Servers by Total Traffic	Shows the total traffic in bytes for the top 10 FTP servers on an hourly basis over a period of 24 hours. It configured to display as a line chart by default, but users have the option to view it as a compact spark line chart.

Information	Visualization Type
Top FTP Servers by Current Connection	Shows the maximum number of current connections for the top 10 FTP servers on an hourly basis over a period of 24 hours. It configured to display as a line chart by default, but users have the option to view it as a compact spark line chart.
Top FTP Servers by Events	Shows events for the top 100 FTP Servers. It is configured to display as an entity state list.
FTP Server Alerts	Shows warnings and errors on the top 100 FTP servers over a period of one week. It is configured to display in a grid, but users have the option to view it as a pie chart.
FTP Site Alerts	Shows warnings and errors on the top 100 FTP sites over a period of one week. It is configured to display in a grid, but users have the option to view it as a pie chart.

To import this pre-configured dashboard into Savision Live Maps and customize it for your environment, complete the following tasks:

- "Import a Dashboard" on page 58
- "Customize a Dashboard" on page 59

Hyper-V

This dashboard displays information about the health of your Hyper-V devices. It shows information about hosts and virtual machines separately. The following table lists the types of information retrieved and the visualization format used for each.

Table 14: Hyper-V

Information	Visualization Type
Hyper-V Hosts	Shows the health states of the top 100 servers that host Hyper-V. It is configured to display as a heat map.
Virtual Machines	Shows the health states of the top 100 virtual machines that host Hyper-V. It is configured to display as a heat map.
Used Hard Drive Space	Shows the percentage of used hard drive space for virtual machines. It is configured to display as a line chart.

Information	Visualization Type
Virtual Hard Disk	Shows the health state of virtual hard disks. It is configured to display as an entity state list by default, but users have the option to view it as a heat map.
Hyper-V Virtual Machine Alerts	Shows warnings and errors on Hyper-V virtual machines. It is configured to display as a heat map.
Hyper-V Host Alerts	Shows warnings and errors on servers that host Hyper-V. It is configured to display in a grid.

To import this pre-configured dashboard into Savision Live Maps and customize it for your environment, complete the following tasks:

- "Import a Dashboard" on page 58
- "Customize a Dashboard" on page 59

SCOM Data Warehouse SQL Database Health

The SCOM Data Warehouse SQL Health dashboard displays information about the health of your SQL databases. The information is retrieved from SCOM using SQL widgets, which query the SQL databases in the data warehouse. The SQL queries used in the widgets are designed so that you can visualize the information in multiple formats. The following table lists the types of information retrieved and the visualization format used for each.

Table 15: SQL Database Health in the SCOM Data Warehouse

Information	Visualization Type
SQL DB States	Shows the health state of SQL databases in the data warehouse. It is configured to display as an entity state list by default, but users have the option to view it as a heat map.
Top 10 Largest Tables (MB)	Shows the top 10 largest database tables. It is configured to display as a pie chart by default, but users have the option to view it as a grid.
Buffer Cache Hit Ratio	Shows information about buffer cache hits and misses. It is configured to display as a bullet graph by default, but users have the option to view it as a single metric, an entity state list, or a heat map.

Information	Visualization Type
Errors	Shows errors based on type, such as syntax, out of resources, locking errors, db integrity errors, and more. It is configured to display as a pie chart by default, but users have the option to view it as a grid or a single metric.
Drive Space	Shows the amount of free space on specified drives in a grid format.
Requests Per Second	Shows batch requests per second as a single metric but is also available to users in a grid format.
Memory Usage	Shows how much physical memory is in use and how much is available. It is configured to display as a pie chart by default, but users have the option to view it as a grid.
Deadlocks Per Second	Shows a specific error: the number of deadlocks per second. It is configured to display as a single metric by default, but users have the option to view it as a grid.
SQL Server DB Usage	Shows details about database usage on the SQL server. It is configured to display as a grid. You can export the information in this widget to a .CSV file using the menu in the top right corner of the grid.
SQL Server Connections	Shows details about connections on the SQL server. It is configured to display as a grid. You can export the information in this widget to a .CSV file using the menu in the top right corner of the grid.
Entities Modified Last 24 Hrs	Shows details about entities in the data warehouse that have been modified in the past 24 hours. It is configured to display as a grid. You can export the information in this widget to a .CSV file using the menu in the top right corner of the grid.

To import this pre-configured dashboard into Savision Live Maps and customize it for your environment, complete the following tasks:

- "Import a Dashboard" on page 58
- "Customize a Dashboard" on page 59

SCOM Environment Health

The SCOM Environment Health Dashboard displays health information for your SCOM environment and hardware.

The following table lists the types of information retrieved and the visualization format used for each.

Table 16: Health State of SCOM Environment

Information	Visualization Type
Live Maps Server Capacity	Shows the amount of free memory in comparison to the total system memory. It is configured to display as a single metric.
Management Servers	Shows the health state of management servers. It is configured to display as a heat map.
Data Access Services	Shows the health state of data access services. It is configured to display as a heat map.
Management Configuration	Shows the health state of the management configuration service group. It is configured to display as a heat map.
SQL Operations Capacity	Shows the amount of free memory in comparison to the total system memory, as well as the amount of free disk space on C: and D: drives. Each value is configured to display as a single metric.
SQL Data Warehouse Capacity	Shows the amount of free memory in comparison to the total system memory, as well as the amount of free disk space on C: and D: drives. Each value is configured to display as a single metric.
Management Server Resource Pools	Shows the health state of all management server resource pools. It is configured to display as a heat map.
All Operations Manager Objects	Shows the health state of all Operations Manager objects. It is configured to display as a heat map by default, but users have the option to view it as an entity state list.

To import this pre-configured dashboard into Savision Live Maps and customize it for your environment, complete the following tasks:

- "Import a Dashboard" on page 58
- "Customize a Dashboard" on page 59

SCOM Noise

This dashboard identifies the parts of your SCOM environment that generally cause performance issues. You can use this information to configure your SCOM environment and reduce the load on both the management server and the database server. The following table lists the types of information retrieved and the visualization format used for each.

Table 17: SCOM Noise

Information	Visualization Type
Monitor Noise Per Object	Shows the 10 noisiest monitors in the database over the past 7 days, organized by object or computer. It is configured to display as a pie chart by default, but users have the option to view it as a bar chart or in a grid.
Alerts by Repeat Count	Shows alerts that have been repeated, organized the number of times they have been raised. It is configured to display as a pie chart by default, but users have the option to view it in a grid.
Configuration Churn Per Day	Uses a SQL script to shows configuration objects in the data warehouse that have been changed within the past 24 hours. It is configured to display as a heat map by default, but users have the option to view it as an entity state list or in a grid.
Modules And Workflows	Targets the counters for Health Service/Workflow and Health Service/Module. It is configured to display as a compact horizontal bar chart by default, but users have the option to view it as a line chart, an entity state list, or in a grid.
Most Active Performance Counters	Shows the top performance counters, organized by the performance object and counter name.

To import this pre-configured dashboard into Savision Live Maps and customize it for your environment, complete the following tasks:

- "Import a Dashboard" on page 58
- "Customize a Dashboard" on page 59

Server Health

This dashboard provides health information about any computers operating on Windows servers. The following table lists the types of information retrieved and the visualization format used for each.

Table 18: Server Health

Information	Visualization Type
Server Health States	Shows the health states of Windows servers. It is configured to display as a pie chart by default, but users have the option to view it as an entity state list or a heat map.
Percent of Memory Used (Top 5)	Shows the top five Windows servers with the highest percentage of memory used. It is configured to display as a line chart.
Processor Time (Top 5)	Shows the top five Windows servers with the highest percentage of processor time. It is configured to display as a line chart.
Logical Disk Free Space (Top 5)	Shows the five Windows servers with the least amount of free logical disk space available. It is configured to display as a line chart.

To import this pre-configured dashboard into Savision Live Maps and customize it for your environment, complete the following tasks:

- "Import a Dashboard" on page 58
- "Customize a Dashboard" on page 59

Service Manager Health

This dashboard provides health information your Systems Center Service Manager environment. The following table lists the types of information retrieved and the visualization format used for each.

Table 19: Service Manager Health

Information	Visualization Type
Service Manager Health State	Displays the health state of the Service Manager in a heat map.

Information	Visualization Type
Service Manager Database Server Capacity	Uses the SQL widget to retrieve information about the capacity of the Service Manager database. It displays the amount of free memory compared to total memory, as well as the amount of free disk space on C:\. It is configured to display as a single metric by default, but users have the option to view it as a grid.
Alerts	Displays new warnings and errors in the Service Manager environment. It is configured to display as a grid by default.
Tasks By Category	Uses the SQL widget to retrieve information about the number of Service Manager tasks in each category. It is configured to display as a pie chart by default, but users have the option to view it as a grid.
Rules By Category	Uses the SQL widget to retrieve information about the Service Manager rules in each category. It is configured to display as a pie chart by default, but users have the option to view it as a grid.
Installed Management Packs	Uses the SQL widget to retrieve information about the management packs that are installed on the server. It displays as a grid.

To import this pre-configured dashboard into Savision Live Maps and customize it for your environment, complete the following tasks:

- "Import a Dashboard" on page 58
- "Customize a Dashboard" on page 59

Service Manager Work Items

The Service Manager Work Items dashboard displays information about work items that impact the Service Manager environment. The information is retrieved from Service Manager using the SQL widget. The following table lists the types of information retrieved and the visualization format used for each.

Table 20: Service Manager Work Items

Information	Description
Incident Status	Displays incidents within the last 30 days that have a status of active, resolved, or closed. It is configured to display as a pie chart by default, but users have the option to view it in grid format.
Incidents	Displays incidents within the last 30 days that are open or active. It is configured to display as a heat map by default, and also makes a grid view available.
Change Request Status	Displays the status of change requests within the last 30 days: in progress or on hold. It is configured to display as a pie chart by default, but users have the option to view it in grid format.
Change Requests	Displays a list of the change requests within the past 30 days and indicates the status of each. It is configured to display as a heat map by default, but users have the option to view it in grid format.
Problem Status	Displays problems within the last 30 days that have a status of closed, resolved, or active. It is configured to display as a pie chart by default, but users have the option to view it in grid format.
Problems	Displays open or active problems as a heat map.
Number of Work Items per Day	Shows the number of change requests, incidents, and problems per day in bar graph format.

To import this pre-configured dashboard into Savision Live Maps and customize it for your environment, complete the following tasks:

- "Import a Dashboard" on page 58
- "Customize a Dashboard" on page 59

Import a Dashboard

Savision Live Maps has a catalog of pre-configured items that you can import, such as themes and dashboards.

1. From the Administration menu, select **Catalog**.
2. To filter the list of catalog items, enter `dashboards` in the **Category** field.

3. Select the dashboard that you want to import. To select multiple dashboards, hold down the SHIFT key and select the items you want to add.
   A dialog box asks you to confirm the action.
4. Click **OK**.

Customize a Dashboard

After you import a pre-configured dashboard, use the following procedure to edit it for your environment.

Before you Begin

Ensure that you meet the prerequisites listed in the following table.

Dashboard Type	Requirements
SCOM Data Warehouse SQL Health	Give the app pool user the "View Server State" permissions in the master database.
SCOM Environmental Health	
Hyper-V	Ensure that you have installed the Hyper-V management packs for 2012, 2012 R2, and 2016.

1. Put the dashboard in edit mode by clicking on the lock icon to unlock it.
2. Click the ⚙ icon to edit the widget configuration.
3. On the **Data** tab, edit the settings:
   - For widgets that use scripts to retrieve information, edit the database and server settings. You may also need to change the authentication type, which is set to Windows authentication by default. If you use SQL authentication, you need to change the selection and enter the SQL credentials.
   - For other widgets, edit the settings to scope the search to a specific class or group if needed.
   - For widgets that include an Advanced field, you have the option of using it to add a SQL query that searches for additional object properties. For information about using the Advanced field, see "Using the Advanced Field" on page 13.
4. Optional. On the **General** tab, edit the name of the widget.
5. Optional. If the widget uses a script to retrieve information, there may be variables in the script that you can customize. Use the links below to see a list of the variables that you can customize.

Script Variables

- "Variables for Service Manager Work Items Widgets" on page 60
- "Variables for Configuration Manager Software Widgets" on page 60

Variables for Service Manager Work Items Widgets

The table below lists the variables that you can set in the Service Manager Work Items dashboard. The following widgets have scripts that contain variables:

- Incident Status
- Incidents
- Change Request Status
- Change Requests
- Problem Status
- Work Items per Day

Commands And Variables	Description
SET	
start	Specify the number of days of data to display, using the format `-30`.

Variables for Configuration Manager Software Widgets

The following table lists the variables that you can set in the Configuration Manager Software dashboard.

Widget	Commands And Variables	Description
Software Update Deployment by Type	**SET**	
	deployment	Type name of your chosen software update deployment.
	SELECT	
	title	Optional. Enter a title to simplify the name on the piechart.
	Name	Order alphabetically by name.

Widget	Commands And Variables	Description
Chrome Installation on Clients widget	**DECLARE**	
	software	Enter the name of the software in single quotes.
	requiredVersion	Enter the required version number in single quotes.