Guide to Microsoft System Center Management Pack for SQL Server Reporting Services (Native Mode)

Published in December 2020 by Microsoft Corporation.

This guide is based on version 7.0.29.0 RTM of the Management Pack for Microsoft SQL Server Reporting Services (Native Mode).

The Operations Manager team encourages you to provide feedback on the management pack by sending it to sqlmpsfeedback@microsoft.com.

Copyright

Table of Contents

Changes History

December 2020 - version 7.0.29.0 RTM

- **What's New**

  - Updated and improved SCOM 2019 HTML Dashboards to display SSRS health and alerts
  - Updated alert description for Report Manager Accessible & Web Service Accessible monitors

June 2020 - version 7.0.22.0 RTM

- **What's New**

  - Added tasks Start/Stop Reporting Services Windows Service
  - Updated display name of SSRS Deployment object to display AG name as part of it instead of GUID
  - Updated logic of installation detection for local Reporting Services instance to query Windows Registry instead of WMI
  - Improved error handling for cases when error "Process with an Id is not running" is returned

- Updated display strings

- **Bug Fixes**

    - Fixed false alerting with status code 400 in monitor Report Manager Accessible for SSRS 2016 and PBIRS
    - Fixed alert parameter replacement failure in monitor Report Manager Accessible
    - Fixed issue with accessing DBConnectionString property of PBIRS

December 2019 - version 7.0.19.0 CTP

- **What's New**

    - Added support for monitoring SQL Server Reporting Services 2012, 2014, and 2016 in addition to 2017 and up
    - Updated Event Log Collection Target Management Server Discovery to make it use default SCOM action profile instead of SQL MP Discovery run as profile
    - Updated display strings

- **Bug Fixes**

    - Fixed NullReferenceException error when Report Server portal being configured to have several ports

April 2019 - version 7.0.15.0 RTM

- Supported last changes in version-agnostic management pack for SQL Server

January 2019 - version 7.0.12.0 RTM

- MP was prepared for GA release

November 2018 - version 7.0.11.0 CTP

- Added support for Power BI Report Server

October 2018 - version 7.0.10.0 RTM

- Replaced the Core Library in the delivery with the version 7.0.7.0, that version which is delivered with the most recent RTM version of the management pack for SQL Server 2017+.
- Updated the monitoring of Memory Consumption and CPU Usage in order to collect performance data for all subprocesses in addition to the main SSRS service process.
- Updated Summary dashboards.
- Fixed minor issues found in the CTP version.
- Updated display strings.

June 2018 - version 7.0.8.0 CTP. The original release of this management pack.

Management Pack Scope and Supported Configurations

Management Pack for SQL Server Reporting Services is version-agnostic and supports discovery and monitoring of SQL Server Reporting Services 2012 through 2019 and higher as well as Power BI Report Server.

This section explains what SQL Server Reporting Services features are covered by this management pack, what configurations are supported, what monitoring features the management pack offers and what prerequisites should be met to begin with this management pack.

Reporting Services Configurations and Features

Operating Systems and Platforms

The list of supported operating systems/platforms is as follows:

- Windows Server 2012
- Windows Server 2016
- Windows Server 2019

SQL Server Reporting Services Features

The following is a list of features and configurations supported by Management Pack for SQL Server Reporting Services. Unsupported features and configurations are also on this list marked as "Not supported":

- SQL Server Reporting Services Instance (Native Mode)

- SQL Server Reporting Services Scale-out deployment

- Power BI Report Server - Verified with build 15.0.1104.239

  The management pack considers PBIRS as a special kind of SSRS and provides the same monitoring for PBIRS instances as it does for SSRS instances. In this guide, we will use "SSRS" or "Reporting Services" but each term is intended for both SQL Server Reporting Services and Power BI Report Server.

- Clustered installation of SSRS - **Not supported by SSRS**

SCOM Configurations

Management Pack for SQL Server Reporting Services is designed for the following versions of System Center Operations Manager:

- System Center Operations Manager 2012 R2
- System Center Operations Manager 2016
- System Center Operations Manager 1801
- System Center Operations Manager 1807
- System Center Operations Manager 2019

A dedicated Operations Manager management group is not required for this management pack.

Prerequisites

- **.NET Framework 4.5+**

  Installation of .NET Framework 4.5 or newer is required

- **Management Pack for Windows Server Operating System**

  As a best practice, you should import Windows Server Management Pack for the operating system you are using. The management packs monitor aspects of the operating system that influence the performance of computers running SQL Server Reporting Services such as disk capacity, disk performance, memory utilization, network adapter utilization and processor performance.

- **Each agent has the Agent Proxy option enabled**

  Enable the Agent Proxy option on all agents installed on servers that host either an instance of SQL Server Reporting Services or SQL Server instance with respective SSRS Catalog Database.

  The Agent Proxy setting allows an agent to forward data to the management server on behalf of another entity and it should be enabled if the agent workflow scenarios discover any non-hosted objects (the management pack creates a non-hosted object for every SQL Server instance).

  To enable Agent Proxy option, perform following steps:

  1. Open the Operations Console and click **Administration**.
  2. In the **Administrator** pane, click **Agent Managed**.
  3. Double-click an agent.
  4. On the **Security** tab, select *Allow this agent to act as a proxy* and discover managed objects on other computers.

- **TCP/IP protocol is enabled for SQL Server Instance**

  Enable the TCP/IP protocol for SQL Server instances that host the report server database.

- **SQL Server Browser service is enabled for SQL Server Instance**

  The SQL Server Browser service is required for Reporting Services discovery and monitoring and must be installed and running on both the computers with Reporting Services and on the computers with SQL Server instances that host the report server database.

- **Installation of management pack "Microsoft SQL Server on Windows (Discovery)" is required for Deployment monitoring**

  This management pack does not discover database objects for SSRS Catalog Database or SSRS Temporary Database.

  Import the management pack for SQL Server to enable discovery, monitoring and health rollup for SSRS databases and Deployment performance collection. For more information, see Discovery of SQL Server Reporting Services Deployment.

- **Monitoring with a domain account is a recommended option**

  Monitoring with a domain account is highly recommended. For more information, see Run As Profiles.

  You can use the Local System account or HealthService SSID as an action account, but with some restrictions.

Management Pack Delivery

You can download Management Pack for SQL Server Reporting Services from the Microsoft portal or find it in the System Center Operations Manager Online Catalog.

The package includes the following files:

- **Microsoft.SQLServer.ReportingServices.ManagementPack.msi**

    A set of .MP and .MPB files to start monitoring of Reporting Services on Windows.

- **SQLServerReportingServicesGuide.md**

    An SSRS operations guide.

- **SQLServerDashboardsGuide.pdf**

    An operations guide for SQL MP Dashboards.

- **SSRSMPWorkflowList.pdf**

    A complete list of SSRS MP workflows with descriptions and parameters.

Management Pack for SQL Server Reporting Services consists of the following files:

File	Description
Microsoft.SQLServer.ReportingServices.Discovery.mpb	Microsoft SQL Server Reporting Services (Discovery). This Management Pack discovers Microsoft SQL Server Reporting Services (Native Mode) and related objects. The management pack contains the discovery logic only and requires a separate monitoring management pack to be imported to monitor the discovered objects.
Microsoft.SQLServer.ReportingServices.Monitoring.mpb	Microsoft SQL Server Reporting Services (Monitoring). This management pack enables the monitoring of Microsoft SQL Server Reporting Services (Monitoring, Native Mode).
Microsoft.SQLServer.ReportingServices.Core.Library.mpb	Microsoft SQL Server Reporting Services Core Library. This library contains the basic components required for the monitoring of Microsoft SQL Server Reporting Services (Monitoring, Native Mode).
Microsoft.SQLServer.ReportingServices.Core.Views.mp	Microsoft SQL Server Reporting Services Core Library (Views). This management pack defines views for Microsoft SQL Server Reporting Services (Native Mode).

File	Description
Microsoft.SQLServer.Visualization.Library.mpb	Microsoft SQL Server Visualization Library. This library contains basic visual components required for SQL Server dashboards.
Microsoft.SQLServer.Core.Library.mpb	Microsoft SQL Server Core Library. This Management Pack is the core library for all versions of SQL Server. It defines all SQL Server base classes and relationships.

Management Pack Purpose

Monitoring Scenarios

Discovery of SQL Server Reporting Services Instance and Power BI Report Server

Management Pack for SQL Server Reporting Services automatically discovers instances of both SQL Server Reporting Services and Power BI Report Server by implementing the following workflows:

- Reading the registry to detect if the installation of SSRS or PBIRS exists on the server. If so, the management pack creates a "Seed" object.

- If the "Seed" object has been discovered, the management pack reads various data sources such the registry, WMI, SSRS configuration file, and so on to discover instance properties and the *Deployment Seed* object.

  The "Deployment Seed" object is an unhosted object that is managed by SCOM Management Server. Appropriate permissions are required to access all the necessary data sources. For more information, see Security Configuration.

Discovery of SQL Server Reporting Services Deployment

SSRS Deployment includes the following components:

- One or more instances of SQL Server Reporting Services.
- SSRS Database and SQL Server that hosts it.

SSRS Database is a term that describes two databases used by SSRS:

- SSRS Catalog Database
- SSRS Temporary Database

Pre-installation of *Microsoft SQL Server on Windows (Discovery)* version 7.0.20.0 or later is required to discover SSRS Deployment. This MP file is part of the *Microsoft System Center Management Pack for SQL Server on Windows* delivery. In the case of absence of *Microsoft SQL Server on Windows (Discovery)*, Management Pack will not discover and monitor availability and performance of SSRS Deployment. It does not affect monitoring of SSRS Instance.

The management pack supports different kinds of installations of SQL Server. The SSRS Database can be deployed to:

- Stand-alone instance (either named or the default one)
- Cluster instance
- Availability group

For Availability Group failover, it will take about 8 hours to rediscover Deployment. As a workaround you can temporary change running intervals of the following discoveries:

- MSSQL Reporting Services: Native Mode Deployment Discovery
- MSSQL Reporting Services: Deployment Seed Discovery

To find a SQL Server instance that hosts SSRS Database, the management pack uses the connection string utilized by SSRS Instance to connect to the database. The following connection string formats are supported:

- MachineName
- MachineName\InstanceName
- IPAddress
- IPAddress,PortNumber
- (local)

Deployment discovery runs on a SCOM Management Server and queries SCOM API to get a list of SSRS Instances and databases discovered on different SQL Servers.

Not only Deployment discovery creates a new *Deployment* object, but it also create a new *Deployment Watcher* object, both of which are unhosted.

SSRS Scale-out Deployment is a distributed application. Therefore, the Deployment object is managed by Management Server. The main purpose of the Deployment object is to combine the health state of various SSRS components and group respective SCOM objects.

Deployment Watcher is an auxiliary object and managed either by an agent installed on the server that hosts SSRS Database, or an agent that hosts one of SSRS Instances from the given deployment. This object is used to collect information about SQL Server Reporting Services deployments.

Availability of SQL Server Reporting Services Components

This management pack introduces a set of monitors that enable monitoring of both SSRS Deployments and SSRS Instances. The monitors verify availability of these components from the following perspectives:

- SSRS Scale-out Deployment:
    - SSRS catalog database is accessible;
    - SSRS temporary database is accessible;
    - There are no broken references to shared data sources;
    - Number of failed report executions (expressed as a percentage of total report executions) is below the threshold;
    - All instances within deployment are discovered.
- SSRS Instance:
    - SSRS catalog database is accessible;
    - SSRS temporary database is accessible;
    - SSRS windows service is started;
    - SSRS web service is accessible;

- SSRS report manager is accessible;
- SSRS Instance is not using too much CPU resources;
- SSRS Instance is not using too much memory resources;
- There is no memory configuration conflict between SSRS Instance and SQL Server Database Engine (if both components are running on the same server);
- Other processes allow enough memory resources for the SSRS Instance;
- A number of failed report executions per minute is below the threshold for the given SSRS Instance.

Some monitors are disabled by default. Please review the **Appendix: Management Pack Objects and Workflows** in the **SSRSMPWorkflowsList.pdf** file for more details about monitoring workflows implemented in this management pack.

Performance of SQL Server Reporting Services Installation

This management pack collects the following performance metrics:

- SSRS Scale-out Deployment:
    - Failed report executions per minute
    - Report executions per minute
    - Number of reports
    - Number of shared data sources
    - Number of subscriptions
    - On-demand execution failures per minute
    - On-demand executions per minute
    - Scheduled execution failures per minute
    - Scheduled executions per minute
- SSRS Instance:
    - CPU utilization (%)
    - WorkingSetMaximum (GB)
    - WorkingSetMinimum (GB)
    - Memory consumed by other processes (%)
    - Memory consumed by SSRS (GB)
    - Total memory on the Server (GB)
    - Total memory consumed on the server (GB)
    - Failed report executions per minute
    - Report executions per minute

Please review the **Appendix: Management Pack Objects and Workflows** in the **SSRSMPWorkflowsList.pdf** file for more details about monitoring workflows implemented in this management pack.

How Health Rolls Up

The following diagram shows how health states of objects roll up for the SQL Server on Windows management pack.
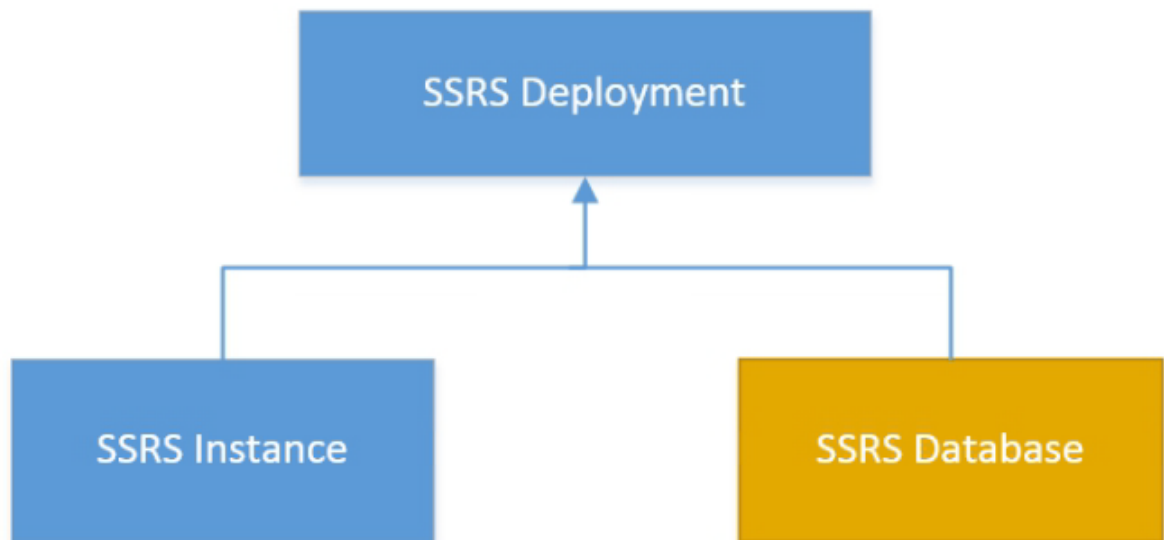
Legend

 Defined in the SQL Server Reporting Services Discovery Management pack

 Defined in the SQL Server on Windows Discovery Management pack

Configure the Management Pack

This section explains how to configure Management Pack for SQL Server Reporting Services.

Best Practice: Create a Management Pack for Customizations

The management pack is sealed so that you cannot change any of the original settings. However, you can create customizations such as overrides or new monitoring objects and save them to a different management pack.

By default, Operations Manager saves all customizations to the default management pack. As a best practice, you should create a separate management pack for each sealed management pack that you want to customize.

Creating a new management pack for storing overrides has the following advantages:

- When you create a management pack for the purpose of storing customized settings for a sealed management pack, it is helpful to base the name of the new management pack on the name of the management pack you are customizing such as, for example, *Microsoft SQL Server Reporting Services Overrides*.

- Creating a new management pack for storing customizations of each sealed management pack makes it easier to export the customizations from a test environment to a production environment. It also makes it easier to delete a management pack, because you must delete any dependencies before you can delete a management pack. If customizations for all management packs are saved in the Default Management Pack and you need to delete a single management pack, you must delete the Default Management Pack first which also deletes customizations for other management packs.

For more information about management pack customizations and the default management pack, see Using Management Packs.

How to Configure a Run As Profile

To configure a Run As profile, perfrom the following steps:

1. Identify the names of the target computers where the default action account has insufficient rights to monitor SQL Server Reporting Services.
2. For each system, create or use an existing set of credentials that have at least the set of privileges described in Security Configuration.
3. For each set of credentials identified in Step 2, make sure a corresponding **Run As Account** exists in the management group. Create a **Run As Account** if necessary.
4. Configure mapping between targets and Run As accounts on the **Run As Accounts** tab of each of the Run As profiles.

Refer to the Run As Profiles section for the detailed explanation of what Run As profiles are defined in the management pack.

For more information about discoveries, rules and monitors, see Appendix: Run As Profiles.

Security Configuration

This section explains how to configure security for Management Pack for SQL Server Reporting Services.

Run As Profiles

When the management pack is imported for the first time, it creates the following Run As profiles:

- Microsoft SQL Server Discovery Run As Profile
- Microsoft SQL Server Monitoring Run As Profile
- Microsoft SQL Server SCOM SDK Run As Profile

By default, all discoveries, monitors and rules defined in the SQL Server Reporting Services management pack use accounts defined in the **Default Action Account** Run As profile.

If the default action account for the given system does not have necessary permissions to discover or monitor the instance of SQL Server Reporting Services, those systems can be bound to more specific credentials in Microsoft SQL Server Run As profiles.

> ⚠ It is not recommended to use Local System account or HealthService SSID as its special case to monitor SSRS, as some workflows run from the server hosting an SSRS instance and try to reach the SSRS Database usually installed on another server. You will need to provide computer accounts of all servers hosting SSRS instances with the required permissions to access the SSRS Database. A domain account is a more preferable option.

Least-Privilege Monitoring Configuration

Configure Permissions in Active Directory

1. In Active Directory, create three domain users that will be commonly used for low-privilege access to all target SSRS instances and SQL Server DBE instances hosting report database:

   o **SSRSMonitoring**
   o **SSRSDiscovery**
   o **SSRSSDK**

2. Create a domain group named **SSRSMPLowPriv** and add the following domain users:

   o **SSRSMonitoring**
   o **SSRSDiscovery**

Configure Permissions on the Agent Machine

1. Grant Local Administrator permissions to the **SSRSMPLowPriv** group.

Configure Permissions on the Instance of SQL Server Reporting Services

1. Open Internet Explorer and connect to SSRS Report Manager.
2. Click the **Site Settings** link in the upper right corner of the page to navigate to the **Site Settings** page.
3. Click the **Security** menu item on the left side of the **Site Settings** page.
4. Click "New Role Assignment" button.
5. On **New Role Assignment** enter the group name (<Your Domain>\ **SSRSMPLowPriv**) and select the **System Administrator** checkbox.
6. Click **OK** to apply changes.

Configure Permissions on SQL Server Reporting Services Catalog Database

1. In SQL Server Management Studio, for the instance of SQL Server Database Engine that hosts SSRS Catalog Database, create a login for **SSRSMPLowPriv**.
2. Create a **SSRSMPLowPriv** user in both SSRS Catalog and Temporary databases.
3. Assign the 'db_datareader' role for **SSRSMPLowPriv** on both SSRS Catalog and Temporary databases.

Configure Permissions on the System Center Operations Manager Management Server

1. Grant Local Administrator permissions to the **SSRSSDK** account.

Configure Permissions on the System Center Operations Manager

1. Open the SCOM console and navigate to the **Administration** pane.
2. Open the **User Roles** view (located under the **Security** folder).
3. Right-click the **Operations Manager Operators** role and select **Properties**.
4. On the **General Properties** tab, click **Add**.
5. Find the **SSRSSDK** user and click **OK**.
6. Click **OK** button to apply changes and close the **User Role Properties** dialog.

Configure System Center Operations Manager

1. Import SQL Server Management Pack if it has not been imported.

2. Create **SSRSMonitoring**, **SSRSDiscovery** and **SSRSSDK** Run As accounts with the "Windows" account type. For more information about how to create a Run As account, see How to Create Run As Account in Operations Manager 2012. For more information about various Run As Account types, see Managing Run As Accounts and Profiles in Operations Manager 2012.

3. On System Center Operations Manager console, configure the Run As profiles as follows:

   a. Set "Microsoft SQL Server Discovery Run As Profile" Run As profile to use **SSRSDiscovery** Run As account.

   b. Set "Microsoft SQL Server Monitoring Run As Profile" Run As profile to use **SSRSMonitoring** Run As account.

   c. Set "Microsoft SQL Server SCOM SDK Run As Profile" Run As profile to use **SSRSSDK** Run As account.

> ⚠ If you have imported the management pack for SQL Server (e.g. Microsoft SQL Server on Windows (Discovery)), please note that they use the same Run As Profiles. Therefore, you should adjust the Low Privilege configuration for the SQL Server management packs as well.

View Information in the Operations Manager Console

Version-Independent (Generic) Views and Dashboards

This management pack introduces a common folder structure that will be used by future releases of management packs for different components of SQL Server.

The following views and dashboards are version-independent and show information about all versions of SQL Server:
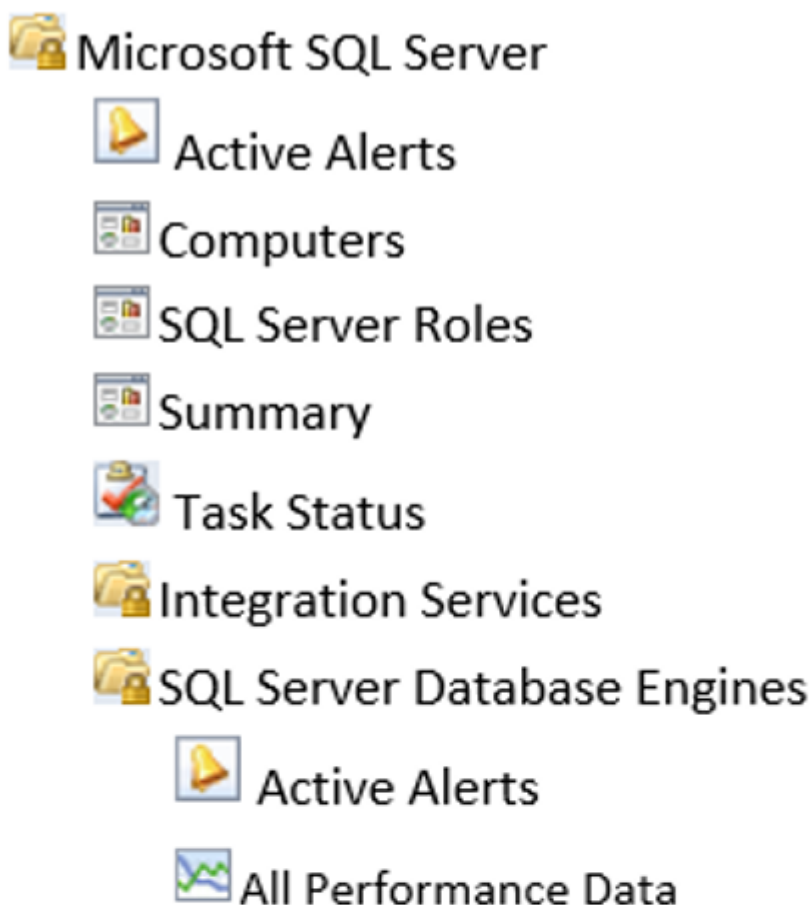
Summary

Task Status

Always on High Availability

Database Engines

    Database Engines

    Databases

    Filegroups

    Database Engines on Linux

        Database Engines

        Databases

        Filegroups

        Summary

    Database Engines on Windows

        Database Engines

        Databases

        Filegroups

        Summary

    Memory-Optimized Data

SQL Agent

SQL Server Reporting Services

    Active Alerts

    Deployments

    Instances

The **Computers** view displays computers on which agents are installed and the management pack discovery is running. Note that this view does not display computers configured for agentless monitoring.

The **SQL Server Roles** dashboard provides information about all instances of SQL Server Database Engine, SQL Server Reporting Services, SQL Server Analysis Services and SQL Server Integration Services:

For more information, see the guide for Microsoft SQL Server dashboards.

Reporting Services Views and Dashboards

Management Pack for SQL Server Reporting Services introduces a comprehensive set of state, performance and alert view that can be found in the dedicated folder.

Some views may contain a very long list of objects and metrics. To find a specific object or group of objects, you can use **Scope** , **Search** , and **Find** buttons on the Operations Manager toolbar. For more information, see the "Finding Data and Objects in the Operations Manager Consoles" article in the Operations Manager Help.

This management pack includes a set of rich dashboards, which provide detailed information about SQL Server Reporting Services Instances and Deployments. The structure of the management pack views and folders is as follows:

RSMP views

Links

The following links provide information about common tasks associated with System Center Management packs:

- [Management Pack Life Cycle](#)
- [How to Import an Operations Manager Management Pack](#)
- [Creating a Management Pack for Overrides](#)
- [Managing Run As Accounts and Profiles](#)
- [How to Export an Operations Manager Management Pack](#)
- [How to Remove an Operations Manager Management Pack](#)

If you have any questions about the Operations Manager and management packs, refer to System Center Operations Manager community forum (http://go.microsoft.com/fwlink/?LinkID=179635).

All information and content on non-Microsoft sites is provided by the owner or users of the website. Microsoft makes no warranties, express, implied, or statutory as to the information at this website.

Appendix: Run As Profiles

Microsoft SQL Server Discovery Run As Profile

- Workflow Type: *Discovery*
    - MSSQL Reporting Services: Deployment Seed Discovery
    - Microsoft SQL Server Reporting Services (Discovery)

Microsoft SQL Server SCOM SDK Run As Profile

- Workflow Type: *Discovery*
    - MSSQL Reporting Services: Native Mode Deployment Discovery
- Workflow Type: *Monitor*
    - All deployment instances are discovered

Microsoft SQL Server Monitoring Run As Profile

- Workflow Type: *Monitor*

    - Configuration conflict with SQL Server
    - CPU utilization (%)
    - Database accessible
    - Instance configuration state
    - Memory consumed by others
    - Memory consumed by SSRS Instance
    - Misconfigured data sources
    - Number of failed report executions

- Report manager accessible
- Temporary database accessible
- Web service accessible
- Windows service state

- Workflow Type: *Rule*

  - MSSQL Reporting Services: CPU utilization (%)
  - MSSQL Reporting Services: Failed report executions per minute
  - MSSQL Reporting Services: Failed report executions per minute (Deployment)
  - MSSQL Reporting Services: Memory consumed by other processes (%)
  - MSSQL Reporting Services: Memory consumed by SSRS (GB)
  - MSSQL Reporting Services: Number of reports
  - MSSQL Reporting Services: Number of shared data sources
  - MSSQL Reporting Services: Number of subscriptions
  - MSSQL Reporting Services: On-demand execution failures per minute
  - MSSQL Reporting Services: On-demand executions per minute
  - MSSQL Reporting Services: Report executions per minute
  - MSSQL Reporting Services: Report executions per minute (Deployment)
  - MSSQL Reporting Services: Scheduled execution failures per minute
  - MSSQL Reporting Services: Scheduled executions per minute
  - MSSQL Reporting Services: Total memory consumed on the server (GB)
  - MSSQL Reporting Services: Total memory on the Server (GB)
  - MSSQL Reporting Services: WorkingSetMaximum (GB)
  - MSSQL Reporting Services: WorkingSetMinimum (GB)

Appendix: Known Issues and Release Notes

The DeploymentSeedDiscovery module fails if the instance is stopped or paused

Issue: When the instance is stopped or paused, Microsoft.SQLServer.ReportingServices.Windows.Module.Discovery.DeploymentSeedDiscovery module fails with the following error: "An error occurred during discovery."

Resolution: No resolution required, as this is a by-design behavior. Start or resume the instance to eliminate the issue.